

Proceedings of the Ninth International Conference on

Epigenetic Robotics

Modeling Cognitive Development in Robotic Systems

November 12-14, 2009

Venice, Italy

Editors

Lola Cañamero

Pierre-Yves Oudeyer

Christian Balkenius

Organizing Committee

General and Program Chair: Lola Cañamero (University of Hertfordshire, UK)
Program Co-chair: Pierre-Yves Oudeyer (INRA Bordeaux, France)
Publicity Co-chair: Hideki Kozima (Miyagi University, Japan)
Publicity Co-chair: Nadia Bianchi-Berthouze (University College London, UK)
Publicity Co-chair: Aude Billard (EPFL, Switzerland)
Publications Chair: Christian Balkenius (Lund University, Sweden)

Program Committee

Pierre Andry (University of Cergy Pontoise, France)
Minoru Asada (Osaka University, Japan)
Gianluca Baldassarre (ISTC-CNR, Italy)
Christian Balkenius (Lund University, Sweden)
Kim Bard (University of Portsmouth, UK)
Tony Belpaeme (University of Plymouth, UK)
Luc Berthouze (University of Sussex, UK)
Nadia Bianchi-Berthouze (University College London, UK)
Mark H. Bickhard (Lehigh University, USA)
Cynthia Breazeal (MIT Media Laboratory, USA)
Angelo Cangelosi (University of Plymouth, UK)
Lola Cañamero (University of Hertfordshire, UK)
Kerstin Dautenhahn (University of Hertfordshire, UK)
Yiannis Demiris (Imperial College London, UK)
Philippe Gaussier (University of Cergy Pontoise, France)
Lakshmi Gogate (Florida Gulf Coast University, USA)
Verena V. Hafner (Humboldt-Universität zu Berlin, Germany)
Antoine Hiolle (University of Hertfordshire, UK)
Ian Horswill (Northwestern University, USA)
Frederic Kaplan (EPFL, Switzerland)
Hideki Kozima (Miyagi University, Japan)
Benjamin Kuipers (University of Michigan, USA)
Giorgio Metta (University of Genova, Italy)
Jacqueline Nadel (University Pierre & Marie Curie, France)
Yukie Nagai (University of Bielefeld, Germany)
Chrystopher Nehaniv (University of Hertfordshire, UK)
Pierre-Yves Oudeyer (INRIA Bordeaux, France)
Giulio Sandini (University of Genova, Italy)
Brian Scassellati (Yale University, USA)
Matthew Schlesinger (Southern Illinois University, USA)
Georgi Stojanov (American University of Paris, France)
Tom Ziemke (University of Skovde, Sweden)

Ad-Hoc Reviewers

Justin W. Hart (Yale University, USA)
Marc Kammer (University of Bielefeld, Germany)
Elizabeth Kim (Yale University, USA)
Dan Leyzberg (Yale University, USA)
Katrin Solveig Lohan (University of Bielefeld, Germany)
Masaki Ogino (Osaka University, Japan)
Lars Schillingmann (University of Bielefeld, Germany)
Anna-Lisa Vollmer (University of Bielefeld, Germany)
Yuichiro Yoshikawa (Osaka University, Japan)

Cañamero, L., Oudeyer, P.-Y., and Balkenius, C. (Eds)
Proceedings of the Ninth International Conference on Epigenetic Robotics:
Modeling Cognitive Development in Robotic Systems
Lund University Cognitive Studies, 146.

ISSN 1101-8453

ISBN 978-91-977-380-7-1

ISRN LUHFDA/HFKO--5070--SE

Copyright © 2009 The Authors

Preface

Since 2001, the Epigenetic Robotics annual International Conference (initially a workshop) has established itself as a unique forum to present and discuss original interdisciplinary research from developmental sciences, neuroscience, biology, cognitive robotics, artificial intelligence, and other disciplines relevant to the study of cognitive development in natural and robotics systems.

Epigenetic systems, whether natural or artificial, share a prolonged developmental process through which varied and complex cognitive and perceptual structures emerge as a result of the interaction of an embodied system with a physical and social environment.

Epigenetic robotics includes the twofold goal of understanding biological systems by the interdisciplinary integration between social/life and engineering sciences and, simultaneously, that of enabling robots and other artificial systems to autonomously develop skills for any particular environment (instead of programming them to solve particular goals for a specific environment). Interdisciplinary theory and empirical evidence are used to inform epigenetic robotic models, and these models can be used as theoretical tools to make experimental predictions in developmental psychology and other disciplines studying cognitive development in living systems.

To promote interdisciplinary discussion, this year's edition of Epigenetic Robotics will include ample time for discussions and organized brainstorming, in addition to keynote talks and oral and poster presentations. Although the conference is open to all aspects of cognitive development, this year we have a special focus on emotional and social development, particularly addressed by keynote speakers and special working groups. Submissions were however welcome regarding all aspects of the study of cognitive development, including (but not limited to):

- The roles of and interactions among motivation, emotion, and value systems in development.
- The development of emotional competencies and systems.
- The development of "social skills", such as imitation, synchrony processing, intersubjectivity, joint attention, intentionality, non-verbal and verbal communication, sensorimotor schemata, shared meaning and symbolic reference, social learning, social relationships, social cognition ("mind reading", "theory of mind").
- The role of play in emotional, social, and cognitive development.
- The development of verbal and non-verbal communication.
- Links between (the development of) expression and communication.
- Architectures for autonomous development.
- Dynamical systems models of emotional, social, and cognitive development.
- The scope and limits of maturation, the mechanisms of open-ended development.
- The mechanisms of stage formation and stage transitions.
- Interaction between innate structure, ongoing developing structure, and experience.
- The interplay between embodiment, learning biases and environment.
- Algorithms for self-supervision, autonomous exploration, representation making, and methods for evolving new representations during ontogeny.

- Philosophical and social issues of development.
- The epistemological foundations of using robots to study development.
- The use of robots as theoretical tools (e.g., to make predictions) in the study of development in biological systems.
- The use of robots in applied settings (e.g., autism therapy) to study development in biological systems.
- Robots that can undergo morphological changes and how they can be used to study interplays among social, emotional, cognitive and morphological development.

Submissions were solicited in two categories: long papers presenting more mature research ideas and results, and short abstracts presenting more preliminary and ongoing work. We received a large number of high quality submissions, and we are very grateful to the members of the Program Committee and other additional reviewers for helping us with the selection process and the care they put to provide constructive feedback to authors).

We are also grateful to the rest of the EpiRob'09 Organizing Committee: Nadia Bianchi-Berthouze (University College London, UK), Aude Billard (EPFL, Switzerland), and Hideki Kozima (Miyagi University, Japan), as Publicity Co-chairs for the help advertising the conference, and to Christian Balkenius (Lund University, Sweden), our Publications Chair, for his help with the publication of the proceedings. Thanks go also to Kostas Karpouzis, Amaryllis Raouzaïou (both of NTUA, Greece) and Giles Thomas for their help in the preparation of various dissemination materials.

We kindly acknowledge the financial support provided by the EU projects FEELIX GROWING (FP6 IST-045169, www.feelix-growing.org), EUCogII (requested, to be confirmed, www.eucognition.org) and of the School of Computer Science at the University of Hertfordshire.

Last, but not least, very special thanks to the students of the Adaptive Systems Research Group at the University of Hertfordshire who eagerly helped with many organizational issues: Luisa Damiano, Ester Ferrari, Yossi Borenstein, Antoine Hiolle, and particularly Sven Magg, who doubled as conference secretariat and webmaster always with extreme efficiency and kindness.

We hope you enjoy the EpiRob'09 proceedings!

Lola Cañamero
EpiRob'09 General and Program Chair

and

Pierre-Yves Oudeyer
EpiRob'09 Program Co-Chair

Table of Contents

Invited talks

<i>An epigenetic approach aids the study of primate social cognition</i>	3
Kim Bard	
<i>Robots as social learners</i>	5
Cynthia Breazeal	
<i>Relationship formation: the culture of attachment</i>	7
Heidi Keller	
<i>Joint attention in apes and humans</i>	9
David Leavens	
<i>The missing link between emotion and motivation: insights from developmental research</i>	11
Jacqueline Nadel	
<i>Why language acquisition and intrinsic motivation should go hand in hand</i>	13
Pierre-Yves Oudeyer	

Papers

<i>The emergence of words: Modelling early language acquisition with a dynamic systems perspective</i> ..	17
Guillaume Aimetti, Louis ten Bosch, Roger K. Moore	
<i>Interactions between motivation, emotion and attention: From biology to robotics</i>	25
Christian Balkenius, Jan Morén, Stefan Winberg	
<i>Adults structure object demonstrations to support infant attention and learning</i>	33
Rebecca J. Brand	
<i>Epigenetic embodiment</i>	41
Luisa Damiano, Paul Dumouchel	
<i>Two examples of active categorisation processes distributed over time</i>	49
Tomassino Ferrauto, Elio Tuci, Marco Mirolli, Gianluca Massera, Stefano Nolfi	
<i>Applying the schema mechanism in continuous domains</i>	57
Franck Guerin, Andrew Starckey	
<i>Caregiver's auto-mirroring and infant's articulatory development enable vowel sharing</i>	65
Hisashi Ishihara, Yuichiro Yoshikawa, Minoru Asada	
<i>Self-regulation mechanism for continual autonomous learning in open-ended environments</i>	73
Kenta Kawamoto, Yukiko Hoshino, Kuniaki Noda, Kohtaro Sabe	

<i>Category-based intrinsic motivation</i>	81
Rachel Lee, Ryan Walker, Lisa Meeden, James Marshall	
<i>A cognitive robotic model of grasping</i>	89
Zoran Macura, Angelo Cangelosi, Rob Ellis, Davi Bugmann, Martin H. Fisher, Andriy Myachykov	
<i>Navigation via Pavlovian conditioning: a robotic bio-constrained model of autoshaping in rats</i>	97
Francesco Mannella, Ansgar Koene, Gianluca Baldassare	
<i>Evaluating intrinsically motivated robots using affordances and point-cloud matrices</i>	105
Kathryn Merrick	
<i>An unsupervised model of infant acoustic speech segmentation</i>	113
Matthew Miller, Alexander Stoychev	
<i>A comparison of strategies for developmental action acquisition in QLAP</i>	121
Jonathan Mugan, Benjamin Kuipers	
<i>Can imprecise internal motor models explain the ataxic hand trajectories during reaching in young infants?</i>	129
Francesco Nori, Giulio Sandini, Jürgen Konczak	
<i>Learning of situation dependent prediction toward acquiring physical causality</i>	137
Masaki Ogino, Tetsuya Fujita, Sawa Fuke, Minoru Asada	
<i>Reward-free learning using sparsely-connected hidden Markov models and local controllers</i>	145
Kohtaro Sabe, Kenta Kawamoto, Hirotaka Suzuki, Katsuki Minamino, Kenichi Hidai	
<i>Formalization of different learning strategies in a continuous domain framework</i>	153
Danijel Skocaj, Matej Kristan, Ales Leonardis	
<i>Learning the sensorimotor structure of the foveated retina</i>	161
Jeremy Stober, Lewis Fishgold, Benjamin Kuipers	
<i>Bottom-up social development through reproducing contingency with sensorimotor clustering</i>	169
Hidenobu Sumioka, Yuji Takeuchi, Yuichiro Yoshikawa, Minoru Asada	
<i>Affordance learning from range data for multi-step planning</i>	177
Emre Ugur, Erol Sahin, Erhan Oztop	

Posters

<i>Using the interaction rhythm to build an internal reinforcement signal: a tool for intuitive HRI</i>	187
Pierre Andry, Nicolas Garnault, Philippe Gaussier	

<i>The IM-Clever project: Intrinsically motivated cumulative learning versatile robots</i>	189
Gianluca Baldassare et al.	
<i>Emotion non-verbal behaviour modelling: Low and high exhibitors</i>	191
Stefania Balzarotti, Rita Ciceri	
<i>Proximo-distal competence based curiosity driven exploration</i>	193
Adrien Baranes, Pierre-Yves Oudeyer	
<i>The role of internal value systems for a memory-based robotic architecture</i>	195
Paul Baxter, Will Browne	
<i>Gesture recognition as a prerequisite of imitation learning in human-humanoid experiments</i>	197
Florian A. Bertsch, Verena V. Hafner	
<i>Designing a turn-taking mechanism as a balance between familiarity and novelty</i>	199
Arnaud J. Blachard, Jacqueline Nadel	
<i>Towards a new social referencing paradigm</i>	201
S. Boucenna, P. Gaussier, L. Hafemeister, K. Bard	
<i>Should I worry about my stressed pregnant robot?</i>	203
David Bowes, Lola Cañamero, Rod Adams, Volker Steuber, Neil Davey	
<i>Retro-projected faces effectiveness on gaze reading</i>	205
Frédéric Delaunay, Joachim de Greef, Tony Belpaeme	
<i>How internal modelling arises when “the world is not enough”: an evolutionary robotics study</i>	207
Onofrio Gigliotta, Giovanni Pezzulo, Stefano Nolfi	
<i>Experimental setup for studying the development of tool-use on the example of object throwing</i>	209
Verena V. Hafner, Werner Sommer	
<i>Learning affective landmarks</i>	211
Antoine Hiolle and Lola Cañamero	
<i>Implementing inhibition of return: embodied visual memory for robotic systems</i>	213
Martin Hülse, Sebastian McBride, Mark Lee	
<i>Distal place recognition based navigation control inspired by hippocampus – amygdala interaction</i>	215
Ansgar Koene, Gianluca Baldassare, Francesco Mannella, Tony J. Prescott	
<i>Learning paths as a sequence of sensorimotor associations</i>	217
Matthieu Lagarde, Pierre Andry, Philippe Gaussier	
<i>Learning to collaborate by observation</i>	219
Stéphane Lallée, Felix Warneken, Peter Ford Dominey	

<i>Integrating a need module into a task-independent framework for modelling emotion: a theoretical approach</i>	221
S.L. Lufti, C. Sanz-Moreno, R. Barra-Chicote, J.M. Montero	
<i>Investigating the basis for conversation between human and robot</i>	223
Carolyne Lyon, Joe Saunders	
<i>The use of emotions in an autonomous agent's decision making process</i>	225
Maria Malfaz, Miguel A. Salichs	
<i>Multimodal representation of hand grasping based on deep belief nets</i>	227
Masaki Ogino, Takanori Nagura, Minoru Asada	
<i>How are representations affected by scene statistics in an adaptive active vision system?</i>	229
Dimitri Ognibene, Giovanni Pezzulo, Gianluca Baldassare	
<i>Self-motivated learning robot</i>	231
Mohamed Oubbati, Günther Palm	
<i>Emerging attention: Reward based model</i>	233
Vitaly Pimenov	
<i>Long short-term memory for affordance learning</i>	235
Sergio Roa, Geert-Jan Kruijff	
<i>Modelling emotional development via finite topological spaces and stratified manifolds</i>	237
Lee Rudolph, Li Han, Eric Charles	
<i>Selective integration based on subjective consistency facilitates simultaneous development of vocal imitation and lexicon acquisition</i>	239
Yuki Sasamoto, Yuichiro Yokishawa, Minoru Asada	
<i>Facing the homunculus: on innate structures for vision of assistive robots</i>	241
Matthias J. Schlemmer, Markus Vincze	
<i>History of usage of Piaget's theory of cognitive development in AI and robotics: a look backwards for a step forwards</i>	243
Georgi Stojanov	
<i>A minimum relative entropy principle for the brain</i>	245
Antoine Van de Ven	
<i>CASA MILA: cross-cultural and social aspects of multimodal interactions in language acquisition</i> ..	247
Paul Vogt	

Invited Talks

An epigenetic approach aids the study of primate social cognition

Kim Bard

University of Portsmouth, UK

Abstract

In this talk, I will discuss my developmental studies of emotion, socialization, and social cognition in chimpanzees. I've found that the behavior of newborn chimpanzees, within the first 30 days of life, changes in response to the social environment, in predictable ways. For example, the emotional expression of joy, the playface, is seen more often in chimpanzees raised in a nursery in which human faces are visible than in one where human faces are masked. Patterns of eye gaze, within the first 3 months of life, are determined by socialization practices. For example, more (or less) mutual gaze is encouraged by mother chimpanzees as a function of less (or more) physical contact with their infants. Nine to twelve months of experience of emotional engagement with social partners and with objects, provides the foundation for joint attention. For example, in nursery-raised chimpanzees, I've found that emotion and sociability account for a significant 50% of the variance in joint attention outcomes. I will speak about the value of applying comparative perspectives to the study of development and of applying developmental perspectives to the study of other species. This comparative developmental approach is, in a general sense, an epigenetic approach that aids in the study of the evolution of social cognition.

Short bio

Kim A. Bard is Professor of Comparative Developmental Psychology and Director of the Centre for the Study of Emotion at the University of Portsmouth, UK. Prior to arriving at Portsmouth, she was Research Scientist at Yerkes National Primate Research Center of Emory University, where she investigated the roles of emotion and socialization in early development, and designed a Responsive Care Nursery for chimpanzees to enhance their species-typical development. Kim Bard has a distinctive perspective, which concerns understanding the process of development in evolution. She conducts empirical studies with an eye to clarifying universal and species-specific characteristics of humans and great apes. Her studies of social cognition suggest that humans and great apes share a large degree of plasticity, especially in early socio-emotional communicative abilities. These social cognitive abilities include intentional and referential communication, and social referencing (i.e., the ability to seek information from a caregiver about novel objects and use that emotional information to regulate behavior). The study of these abilities across species leads to better understanding of the precursors, contexts, and sequelae of social cognition in human development.

Robots as social learners

Cynthia Breazeal

Massachusetts Institute of Technology, Media Laboratory, USA

Abstract

As personal robots enter the social environments of our workplaces and homes, it will be important for them to be able to learn from a wide demographic of people. Our research seeks to identify simple, natural, and prevalent human teaching cues as well as social-cognitive mechanisms that are useful for directing the attention of robot learners so they can learn efficiently and effectively from these interactions.

This research goal is significant for several reasons. First, most people do not have expertise in robotics or machine learning techniques and therefore are not willing or able to tune parameters, label data sets, specify evaluation functions, or otherwise structure the learning task for the robot learner via technical means. Second, personal robots will have to learn new tasks and skills within the bounds of human attention and patience. Third, people bring a lifetime of experience in learning from and teaching others. Through social interaction, they naturally structure appropriate learning environments and interactions for each other to learn efficiently and effectively. Personal robots should be equipped with social cognitive skills to leverage these social interactions to learn efficiently and effectively from people.

In this talk, we present our research in human-robot interaction that concerns the structure of social behavior, embodied interaction, and social-cognitive skills that we term “social filters.” Namely, the myriad of ways in which external social interaction and internal social-cognitive skills mediate the interaction of attention with learning. Social filters can be social-cognitive capabilities such as perspective taking that focuses the robot’s attention on the subset of the problem space that is important to the teacher. This constrained attention allows the robot to overcome ambiguity and incompleteness that can often be present in human demonstrations and thus learn what the teacher intends to teach. Other social filters can be external, dynamic, embodied cues through which the teacher uses his or her body to spatially structure the learning environment to direct the attention of the learner. Our challenge is to identify what cues people use, how they employ them, and how they might be leveraged by the robot’s social-cognitive mechanisms to efficiently guide the robot’s internal attention and learning processes. We report on a series of empirical investigations of human teaching and learning behavior to identify such cues and their use. We then present a set of “social filters” that we have implemented within the cognitive architecture of the robot to demonstrate and evaluate the robot’s ability to learn tasks from human demonstration and guidance.

Short bio

Cynthia Breazeal is an Associate Professor of Media Arts and Sciences at MIT, where she founded and directs the Personal Robots Group (formerly Robotic Life Group) at the Media Lab and also co-directs the Center for Future Storytelling. She is a pioneer of Social Robotics and Human Robot Interaction (HRI). Her research program focuses on developing the principles, techniques, and technologies for personal robots. She has developed numerous robotic creatures ranging from small hexapod robots, to embedding robotic technologies into familiar everyday artifacts, to creating highly expressive humanoids, including the well-known Kismet. Ongoing research includes the development of socially intelligent robot partners that interact with humans in human-centric terms, and how HRI can be applied to enhance human behavior as applied to motor learning and cognitive performance.

Relationship formation: the culture of attachment

Heidi Keller

University of Osnabrück, Germany

Abstract

Relationship formation is a universal developmental task for which humans are equipped with universal predispositions. The claim of attachment theory, that the emergence, the nature and the consequences of attachment are equally universal has been challenged by cultural and cross cultural research. In this presentation, the prevailing attachment conception is discussed as an adaptation to Western middle class psychology, where psychological autonomy is the motor of development. An alternative is presented with the case of the rural Cameroonian Nso who have a physical conception of attachment, i.e. reliability of physical care and body contact, integrated in a multiple caregiving system. The necessity to recognize culture specific solutions of universal developmental tasks is discussed.

Short bio

Heidi Keller holds a chair for developmental psychology at the University of Osnabrück in Germany and is head of the Culture and Development Unit there. She also holds a position at NIH Section of Social and Emotional Development in Bethesda and is a fellow at the Center for Advanced Studies in Berkeley, USA. Currently, Dr. Keller is president of the International Association of Cross-Cultural Psychology. Her longstanding research interests and publications have focused on cross-cultural similarities and differences in childrearing in societies such as Costa Rica, Cameroon, India and the United States.

Joint attention in apes and humans

David Leavens

University of Sussex, UK

Abstract

Joint attention is foundational to the acquisition of language in humans. Numerous theorists have concluded that joint attention is therefore a human species-specific biological adaptation for establishing co-reference. However, the well-documented emergence of humanlike joint attentional skills in our nearest living relatives, the great apes, without any explicit training, poses a challenge for this theoretical perspective. One reaction to these emerging findings from great apes has been the claim that although there is surface similarity in joint attention, there are, nevertheless, deep psychological differences between humans and apes in the display of joint attention. An alternative account emphasises psychological continuity between humans and apes. I will argue for the latter view, in a review of the empirical data on joint attention in humans and great apes.

Short bio

Dr. Leavens earned a B.S. in anthropology (with honours, Phi Beta Kappa) from the University of California at Riverside, in 1990, an M.A. in anthropology from Southern Illinois University at Carbondale, in 1993, and a Ph.D. in psychology from the University of Georgia at Athens, in 2001. He is a senior lecturer in the School of Psychology at the University of Sussex, near Brighton, United Kingdom. Since 1994, he has studied communication in chimpanzees, in collaboration with Dr. William D. Hopkins at the Yerkes National Primate Center, in Atlanta, Georgia, and Prof. Kim A. Bard, at Portsmouth University, United Kingdom.

The missing link between emotion and motivation: insights from developmental research

Jacqueline Nadel

CNRS USR3246, Emotion Centre, Paris, France

Abstract

This talk will deal with a central misunderstanding in the field of cognitive and neurocognitive sciences: the misunderstanding of the dynamics of emotions. Emotions are seen as consequences, although they are causes of experiences. Infant experiences start with the synchronic sharing of postures and gestures that are enacted as a common property by the partners via imitative exchanges. Purposes guide the infants toward the generation of events that can be shared and thus that gain emotional meaning. This leads our human brain to spot emotion everywhere, to attribute mental states of emotion to expressive patterns devoid of humanity, as the report on neuroimaging experiments will show. We will invite you to follow Trevarthen (2005)'s stance when he said: "a valid psychology of emotions is concerned with motives"

Short bio

Jacqueline Nadel is a CNRS Research Director (grade A), where she leads the team "Development and Psychopathology", co-director of a master program at the University Pierre & Marie Curie, and responsible of Autism-Science, an interdisciplinary network of research on autism. She has a wide experience of working with low-functioning children with autism and healthy infants. She has more than 120 publications, and has given numerous international invited keynote talks and invited lectures. She is a specialist of imitation and emotion in normal and impaired development and has edited the books *Emotional Development* (with Darwin Muir, Oxford University Press, 2005) and *Imitation in Infancy* (with George Butterworth, Cambridge University Press, 1999). She has designed innovative set-ups allowing to study emotional interaction of very young infants and children with autism in embedded situations. She has been and is involved in a number of interdisciplinary contracts on various aspects of social and emotional development, including EU-funded projects ADAPT, HUMAINE, MATHESIS and FEELIX GROWING.

Why language acquisition and intrinsic motivation should go hand in hand

Pierre-Yves Oudeyer
INRIA, France

Abstract

Language acquisition and intrinsic motivation are two topics which have mainly been studied separately both in developmental robotics and psychology. In this talk, I will show that they should in fact be studied together, especially if one wants to build developmental robots that may learn language in real complex environments. I will begin by outlining the big challenges of language acquisition in human and robots, especially those related to the acquisition of meaning. In this context, I will explain that many essential meanings learnt at the onset of language are rooted in sensorimotor representations, and affordances in particular. Thus, learning linguistic meanings implies the ability to learn motor affordances. While social learning mechanisms are essential in this process, I will explain why they are not sufficient in real complex sensorimotor spaces in which it is essential that the robot/human infant learns affordances by self-experimentation. Besides, self-experimentation through motor babbling can only be efficient if exploration is guided and organized, which is one of the main functions of intrinsic motivation. I will illustrate this point by describing several experiments in which a robot learns efficiently low-level motor skills and affordances driven by a computational model of intrinsic motivation used as an active learning heuristics. Furthermore, I will argue that intrinsic motivation conceptualized as active learning can also be essential to allow true interactive social language learning, where it allows both the teacher and the learner to control the growth of complexity in linguistic interactions. I will conclude by outlining a number of challenges implied by this joint study of language and intrinsic motivation.

Short bio

Since January 2008, Pierre-Yves Oudeyer is a research scientist in INRIA Bordeaux - Sud-Ouest, heading the FLOWERS team, in developmental and social robotics. Before that, he was a permanent researcher in Sony Computer Science Lab in Paris for 8 years (2000-2007). He studied computer science at Ecole Normale Supérieure de Lyon, and obtained his PhD in artificial intelligence from University Paris VI. He is interested in the mechanisms that allow humans and robots to develop perceptual, motivational, behavioral and social capabilities to become capable of sharing cultural representations and of natural embodied interaction.

Pierre-Yves Oudeyer's recent work in developmental and social robotics focuses on sensorimotor development: how can we build robots that can learn a variety of novel reusable skills in initially unknown environments, either by themselves or through interaction with social peers? In this research, concepts from developmental psychology are imported, formalized and implemented in robots. In particular, he is developing systems capable of intrinsically motivated exploration and learning, aka artificial curiosity, as well as biologically inspired methods of human-robot interaction.

In previous years, he also used robots to study how new linguistic conventions can be established in a society of individuals, as well as the mechanisms of language acquisition. This had a double objective: 1) contributing to the understanding of the acquisition and evolution of language(s), 2) developing new technological approaches for building intelligent sociable robots.

Papers

The emergence of words: Modelling early language acquisition with a dynamic systems perspective

Guillaume Aimetti*

Louis ten Bosch**

Roger K. Moore*

*Speech & Hearing Group

University of Sheffield

Sheffield, UK

{g.aimetti|r.k.moore}@dcs.shef.ac.uk

**Department of Linguistics

Radboud University

Nijmegen, NL

l.tenbosch@let.ru.nl

Abstract

This paper introduces a computational model of early language acquisition that is able to build word-like units from cross-modal stimuli (acoustic and pseudo-visual). The architecture, data processing and internal representations of the model strives for ecological plausibility, and is therefore inspired by current cognitive views of preverbal infant language learning behaviour. In this paper, we attempt to visualise the emergence and development of the models internal representations as an epigenetic landscape, which is a popular method for depicting the evolution of behaviour through the dynamic systems theory. We show that our computational model, through a general statistical learning mechanism, displays similar properties to the dynamic systems theory and supports the empiricist view of human development.

1. Introduction

An increasingly popular view, of developmental researchers, is that the brain is a complex dynamic system and behaviour is emergent through self-organization, known as the dynamic systems theory (DST) (Kelso, 1995, Muchisky et al., 1996, Newell et al., 2003, Smith and Thelen, 2003, Evans, 2007). This perspective takes an empiricist view of development, stating that the acquisition of behaviour is based on a general statistical learning mechanism which is dependent upon experience and initial control parameters. The set of behavioural states of the brain defines a landscape: “Development, then, can be envisioned as a changing landscape of preferred, but not obligatory, behavioural states with varying degrees of stability” (Thelen and Smith, 1995). This view of development, as a constantly evolving landscape, challenges the nativist view that infants are ‘hard-wired’ with skills that are at their disposal from birth or appear at discrete, arbitrary time-steps. As an example,

nativists suggest that young language learners are born with an innate language acquisition device, a universal grammar, which allows them to derive the structure of their native language during a critical period of infancy (Chomsky, 1975, Pinker, 1994).

In the DST framework, attractor states emerge and strengthen as a result of the repeating patterns of the co-operative actions of the systems components. Learning can thus be seen as a shift or bifurcation into a new attractor state by the destabilisation of older stable states (Thelen and Smith, 1995). Behaviour is classed into more or less stable attractor states and changes between these states have a non-linear relationship with environmental input. The behaviour of the system becomes more complex with age, with the formation of multiple attractor states. The wider areas encompass certain categories of actions such as walking, jogging and sprinting.

The timing of developmental changes is controlled by variation in the control parameters, body or environmental changes, rather than some kind of internal clock. Thelen strengthened this theory, overturning the previously held belief that developmental changes were due to cortical inhibition, by proving that the stepping reflex in newborns disappears due to an increase in non-muscular body mass and then reappears when the legs are, once again, strong enough (Thelen and Fisher, 1982). This sparked further research into the application of DST to other motor skills, such as the development of motor skills required to reach for an object (Savelsbergh and Van der Kamp, 1993). DST can thus be used to predict the behaviour of a system with varying control parameters. Thelen argues that the view of development as an evolving landscape is not supposed to prescribe behaviour, but represent a probability of behaviour of a system with varying control parameters.

The epigenetic landscape is currently a popular method for visualising behavioural evolution within developmental science, and was originally drawn in 1957 to display the developmental stability of phenotype over time (Waddington, 1957). Figure 1 is

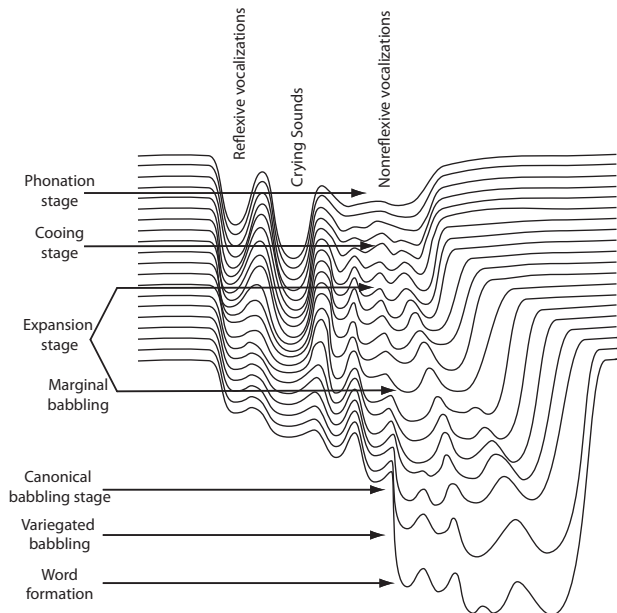


Figure 1: *Diagram of the evolving speech production attractor landscape as illustrated by (Muchisky et al., 1996).*

a diagram of the attractor landscape for the acquisition of speech production skills of an infant as envisioned by current developmental theorists (Muchisky et al., 1996). The three dimensions represent a) time, b) emergent behaviour, and c) the relative stability of the system at any point in time. Each attractor well is a state of behaviour. The deeper the well of an attractor, the more stable the system is when in that state.

It is becoming commonplace to analyse connectionist models, particularly recurrent neural networks, as dynamic systems. We use DST to analyse our computational model in an attempt to gain a deeper understanding of the dynamically evolving internal representations.

The paper is organised as follows. The next section introduces the main components of our computational model, followed by a keyword detection experiment and results. The penultimate section analyses the internal representations through the DST theory. The final section concludes the work and discusses future work being carried out.

2. The computational model

This section describes the Acoustic DP-ngram algorithm (Aimetti, 2009), which is one of three alternative implementations of a comprehensive model of early language acquisition under development in the FP6 FET project ACORNS¹. The other two methods are Non-negative Matrix Factorisation (NMF)

¹<http://www.acorns-project.org>

(Stouten et al., 2007) and Concept Matrices (CM) (Räsänen et al., 2009). CM is the most symbolic approach, detecting recurrent patterns of discrete framed-based codebook labels. DP-ngrams is the most episodic, finding repeating patterns from the raw acoustic signal. NMF sits between the two. Another difference is that CM and DP-ngrams take into account the dynamics of the speech signal over time, whereas NMF does not. Instead, NMF processes the whole utterance to form a representation in memory and at a later stage decomposes it to discover structure in the signal.

Figure 2 displays the interactive framework between the caregiver (carer) and learning agent (LA), along with LA’s learning processes within a cognitively motivated memory architecture (Jones et al., 2006).

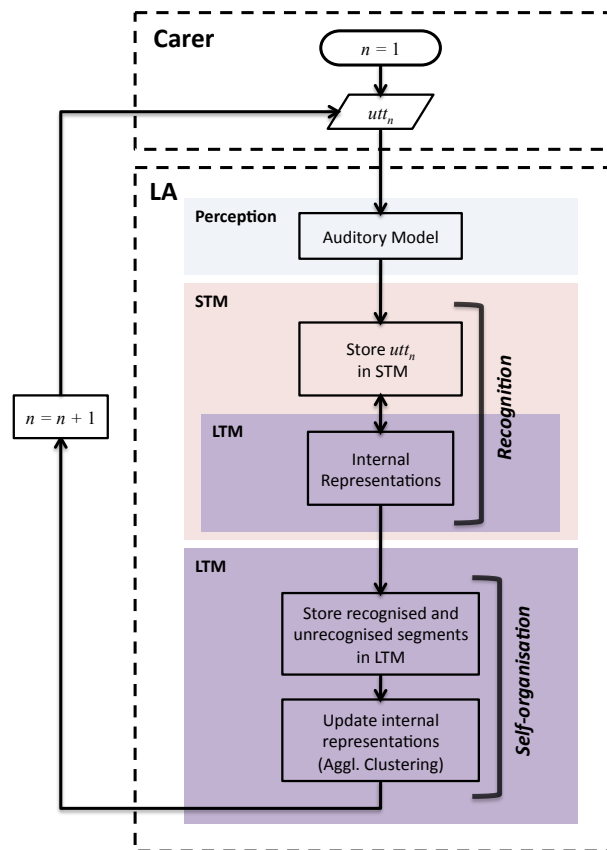


Figure 2: *Flowchart of the carer-learner interactive framework and learning process within a cognitively motivated memory architecture (Jones et al., 2006).*

LA is incrementally presented cross-modal utterances ($utt_{1:\infty}$) by the carer, which contain the raw acoustic signal and a pseudo-visual representation of a keyword (represented as a canonical binary feature indicating its presence within the utterance). Each utterance contains one of ten keywords (bath, telephone, mummy, daddy, car, bottle, nappy, shoe, book and Angus) and has been constructed using a

simple syntax, such as ‘Have you seen the W ?’, where ‘ W ’ is a keyword. LA carries out *recognition* using its internal representations. By this procedure, each utterance (utt_n) is segmented into recognised and unrecognised acoustic segments which are appended to long-term memory (LTM). The segment list in LTM is denoted by $X = \{x_1, \dots, x_m\}$.

Internal representations of keywords and non-keywords emerge through *self-organisation* as a result of clustering the elements of X , on the basis of acoustic similarity, and accumulating their associated pseudo-visual features. The learning processes are discussed in more detail in the following sections.

2.1 Automatic acoustic segmentation

Automatic acoustic segmentation is carried out using the Acoustic DP-ngram algorithm (Aimetti, 2009). This algorithm is a modification of two previous DP-ngram implementations, the first of which was used to find sub-repetitions within a gene sequence (Sankoff and Kruskal, 1983), and the second was used to find sub-repetitions of the output of a phonetic transcription (Nowell and Moore, 1995). The two previous implementations are limited to sequences of discrete symbols, whereas the new implementation can handle multi-dimensional feature vectors. When carrying out experiments directly on the raw acoustic signal we parameterise it to a series of 39-dimensional mel-frequency cepstral Coefficients (MFCC’s), which reflect the frequency sensitivity of the human auditory system.

This Acoustic DP-ngram method uses a popular dynamic programming technique, dynamic time warping (DTW), in order to accommodate temporal distortion present in the acoustic speech signal (similar approaches include (ten Bosch and Cranen, 2007, Park and Glass, 2008)). Through an accumulative scoring mechanism, this method is able to detect similar portions of speech that commonly re-occur within utterances (such as phones, words and sentences) whilst being robust against noise, speech rate and pronunciation variation. The discovered sub-sequence portions are termed *local alignments*. An additional property of the accumulative quality score is that longer, more meaningful local alignments produce a higher final quality score, thus allowing the system to list them in order of importance. The three steps of the segmentation process are outlined below.

Step 1: The carer presents LA with the n^{th} utterance (utt_n), which is stored in short-term memory (STM) as a set of MFCC feature vectors (A). LA then carries out template based recognition by comparing this input representation with each internal representation (B). Both A and B are represented as sequences of feature vectors. By

applying the Euclidean Squared Distance between each pair of feature vectors (v_A, v_B) we obtain a distance matrix $D = (d(v_A, v_B)v_a, v_b)$.

Step 2: D is then used to calculate the accumulative quality scores for successive frame steps within A and B using the recurrence defined by (1) to give the global quality score matrix Q . Higher local quality scores $q_{i,j}$ are obtained by accumulating successive local-matches, thus the score for a local-match must be positive, and scores for non-matches (insertions and deletions) must be negative to penalise temporal distortion (2).

$$q_{i,j} = \max \begin{cases} q_{i-1,j-1} + (s(a_i, b_j) \cdot d(v_i, v_j)), \\ q_{i,j-1} + (s(\phi, b_j) \cdot |d(v_i, v_{-j}) - 1| \cdot q_{i,j-1}), \\ q_{i-1,j} + (s(a_i, \phi) \cdot |d(v_{-i}, v_j) - 1| \cdot q_{i-1,j}), \\ 0 \end{cases} \quad (1)$$

where,

$$\begin{aligned} s(a_i, b_j) &= +1 && \text{(local-match score)} \\ s(\phi, b_j) &= -1 && \text{(insertion score)} \\ s(a_i, \phi) &= -1 && \text{(deletion score)} \\ q_{i,j} &&& \text{(local quality score)} \end{aligned} \quad (2)$$

Backtracking pointers p are maintained at each step of the recursion (3).

$$p_{i,j} = \begin{cases} (i-1, j-1), & \text{(local-match)} \\ (i, j-1), & \text{(insertion)} \\ (i-1, j), & \text{(deletion)} \\ (0, 0) & \text{(initial pointer)} \end{cases} \quad (3)$$

Step 3: Finally, the optimal local alignment is discovered within Q by backtracking from the highest quality score $\max(q_{i,j})$ until $q_{i,j}$ equals 0. Multiple local alignments are discovered by repeating this process while $\max(q_{i,j})$ is greater than the quality threshold (q_{thresh}).

2.2 The emergence of meaning

The incoming utterance is presented to the system in two modalities in parallel, acoustic and pseudo-visual. The pseudo-visual stream contains keyword information as a canonical representation, each keyword is assigned a binary value indicating whether it’s present or not present within the current utterance. It is important to note that there is *no* lexical or phonetic information attached to the pseudo-visual feature and no a priori knowledge is assumed. As the incoming utterance is segmented into recognised and unrecognised portions, LA is also associating co-occurring pseudo-visual features to them.

The next section shows an example of the associative learning process for the first two utterances; for the sake of clarity we are using orthographic and not acoustic data for these examples:

1. Begin life

<i>utt</i> ₁	
Acoustic	Visual
'the_bottle_is_on_the_seat'	0 0 0 1 0 0

LA does not recognise any of the utterance as there are no internal representations yet, so *utt*₁ is stored in LTM as a token in cluster *C*₁.

LTM		
<i>C</i>	Segments	Visual
1	the_bottle_is_on_the_seat	0 0 0 1 0 0

2. Next utterance

<i>utt</i> ₂	
Acoustic	Visual
'have_you_seen_the_bottle'	0 0 0 1 0 0

LA compares *utt*₂ with the internal representation *C*₁ and recognises the acoustic segment 'the_bottle' and associates it with the co-occurring visual feature. The recognised segment is stored as a token in *C*₂.

LTM		
<i>C</i>	Segments	Visual
1	the_bottle_is_on_the_seat	0 0 0 1 0 0
2	the_bottle	0 0 0 1 0 0
3	have_you_seen_	0 0 0 0 0 0

The unrecognised portion of *utt*₂ is stored in *C*₃ with no associated visual features as it has already been recognised and associated with *C*₂.

The associative learning mechanism implemented within this algorithm has been cognitively motivated by current developmental theories and experimental data (Morrongiello et al., 1998, Smith and Yu, 2008), which shows that infants exploit cross-situational statistics to aid the word learning process. In this way, form-referent pairs emerge by grouping the internal acoustic tokens into clusters of the same underlying unit and accumulating the associated visual features. A hierarchical agglomerative clustering (HAC) method is used for the grouping process. The HAC method initialises each element of *X* as separate clusters $\{C_1, \dots, C_k\}$ of size 1, and then merges the two clusters *C*_{*i*} and *C*_{*j*} with the shortest distance, as defined by (4), to create *k* - 1 clusters.

$$d(C_i, C_j) = \min_{v_i \in C_i, v_j \in C_j} [d(v_i, v_j)] \quad (4)$$

This process is repeated until $d(C_i, C_j)$ is greater than the distance threshold *T*, leaving clusters of similar word-like segments. Table (1) displays an example of the kind of clusters that would be created by the system. The segments in bold are the cluster centroids, which is the segment with the shortest total intra-centroid distance as defined by (5).

$$\operatorname{argmin}_{v_a \in C_i} \left[\sum_j d(v_a, v_b) \right] \quad v_b \in C_i \quad (5)$$

LTM			
<i>C</i>	Segments	Visual	Accum.
1	the_bottle_is_on_the_seat	0 0 0 1 0 0	0 0 0 1 0 0
2	the_bottle the_bottle the_bottle	0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0	0 0 0 3 0 0
3	have_you_seen_	0 0 0 0 0 0	0 0 0 0 0 0
4	_the_b _the_ the _	0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0	0 1 0 1 1 0
5	a_bath _bath _bath	1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0	3 0 0 0 0 0

Table 1: Clusters of similar word-like units are obtained with HAC clustering. The item in bold is the *golden* representation and is semantically represented by the accumulative visual features of each token within the cluster.

With experience LA acquires a larger vocabulary of *C* with a greater number of representative acoustic tokens. With a greater number of exemplar acoustic tokens the system is able to handle more variation within the speech signal. The accumulation of the visual features for each cluster also allows LA to build an increasing semantic confidence for keywords.

Table 1 shows how the word-like clusters begin to evolve. The addition of the pseudo-visual modality allows the system to derive meaning for the specific task at hand - discovering keyword units. However, it is not limited to this task as it is a general purpose pattern discovery mechanism which derives meaning through cross-situational association, which concurs with current cognitive theories of human development (Morrongiello et al., 1998, Kuhl, 2004, Smith and Yu, 2008).

2.3 Internal representations: a dynamic systems theory perspective

Describing human development as a dynamic system has become very popular within the cognitive science, where it is visualised as a continuously evolving epigenetic landscape. Current literature depicts these theoretical landscapes as hand drawn examples, such as the diagram of the evolving speech production attractor landscape displayed in figure 1.

As observed above, the landscape shows the emergence of behaviour, with varying stability, as a function of time. Each behaviour is represented as an attractor well and its stability is displayed by its depth

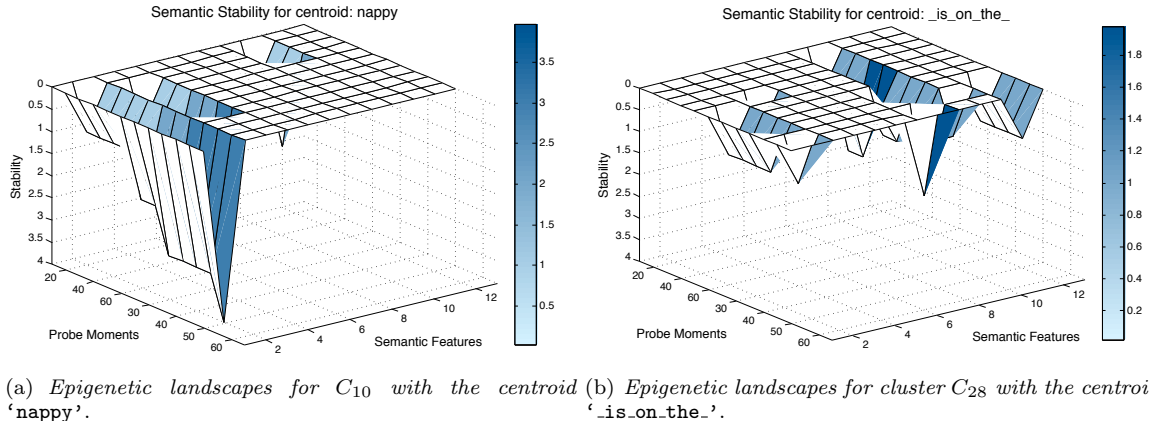


Figure 3: These figures show the epigenetic landscapes for two of LA’s internal representations. a) shows how meaning emerges with experience for clusters representing keywords, whereas b) shows that non-keywords will be semantically noisy and stay relatively flat.

and width. As yet, there does not seem to be anyone who has attempted to visualise the emergence and evolution of the internal representations of a computational model of early language acquisition in a similar fashion.

Figure 3 displays the epigenetic landscapes for two of LA’s internal representations, which are made up of a cluster of similar word-like exemplar segments. The epigenetic landscape in figure 3(a) displays a cluster with an underlying keyword representation, and the epigenetic landscape in figure 3(b) displays a cluster with a non-keyword representation. The x-axis refers to the pseudo-visual label for each of the 10 keywords, the y-axis refers to the number of utterances observed (referred to as probe moments) and the z-axis refers to the semantic stability of the cluster. Stability is simply the accumulation of each visual feature as demonstrated in table 1. Comparing the two epigenetic landscapes in figure 3 it is clear to see that clusters *not* representing a keyword are semantically noisy (fig. 3(b)). Because of this noisiness the system is not able to derive any meaning for this cluster, however, this does not mean that this cluster is not important for the language acquisition process, it just means that it hasn’t been given any meaning for this particular task.

Figure 4 shows the epigenetic landscape for all internal representations in LTM, displayed as wells. The x-axis refers to the cluster space, thus, the width of each well represents the amount of acoustic variation from the median within each cluster. Each cluster is positioned in chronological order along the x-axis, with the newest being appended to the right-hand side. The y-axis refers to the probe moment, which shows the emergence and continuous evolution of each cluster after every utterance observation (only the first 12 utterances have been drawn to preserve clarity). The z-axis refers to the semantic

stability (S), which is defined as the semantic cleanliness of the cluster C_i calculated using (6)

$$S = \left(\frac{\max A}{\sum A} \right) \times \max A \quad (6)$$

where A is the accumulative visual feature vector $\{a_1, \dots, a_n\}$ for C_i .

After observing the first utterance we can see that LA stores it as an internal representation, which can then be used for recognition. It is also clear to see that the most common repetition is ‘the’, as represented by the cluster with the median token ‘_the_’. It is interesting to note that although there are a lot of occurrences of this item it does not gain semantic stability. Whereas the two clusters with the median representations ‘book’ and ‘a_shoe’ gradually gain semantic stability, and represent keywords.

3. Experiments

3.1 Data

The training and test sets have been designed using a selection of utterances recorded within the ACORNS project. The database consists of 4000 utterances spoken by two male (M1 and M2) and two female (F1 and F2) speakers (1000 utterances per speaker). The training set consists of 450 single-speaker utterances from F1, containing both acoustic and pseudo-visual information. The test set consists of 280 single-speaker utterances from F1 that are held-out during training, and *only* contain acoustic information. The accuracy of the systems internal representations is measured with a keyword detection task, LA only observes the acoustic portion of the test utterance and must reply with the correct visual feature. Learning is incremental, therefore LA

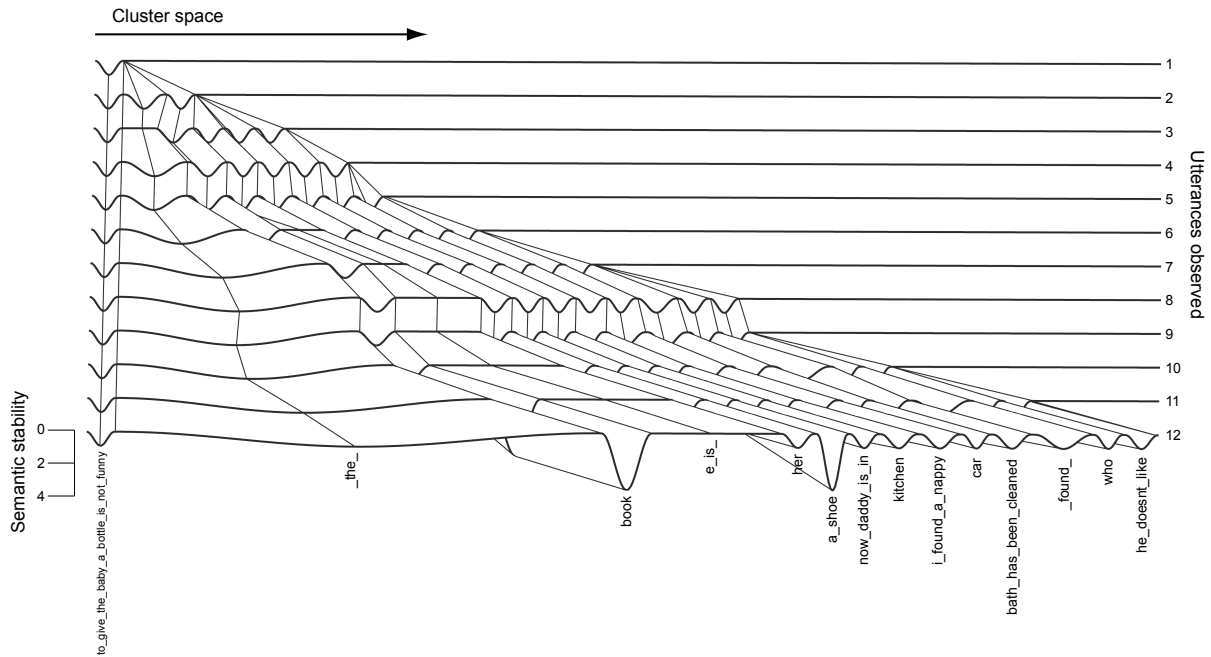


Figure 4: *Epigenetic landscape of all the internal representations during the first 12 training utterances. Each cluster is displayed as an attractor well where the acoustic variation is plotted as the width within the cluster space and semantic stability is plotted as the depth. Two clusters representing an underlying keyword have already begun to emerge from the noisy clusters - ‘book’ and ‘a_shoe’.*

is probed after each successive training utterance is observed with the complete test set, giving us a percentage of correct keyword detections at each stage of development.

3.2 Results

Keyword detection is carried out with the acoustic DP-ngram algorithm. The test utterance is compared with each internal representation, and the visual features associated with the cluster achieving the highest quality score is replied. The system does not know a priori that each keyword is represented by only one visual feature and is penalised when replying with multiple, thus making the problem a lot more difficult but more ecologically plausible.

Figure 3.2 displays the keyword detection accuracy (y-axis) as a function of the number of utterances observed (x-axis). The green plot with circles displays the keyword detection accuracy for LA and the red dotted plot displays chance at 10%.

It can be seen from the figure that keyword representations are discovered extremely quickly but that accuracy never quite reaches 100%. This is because LA has built an internal representation of an infrequently occurring acoustic unit with an associated visual feature. This means that it will be semantically very clean and thus weighted with higher importance. A solution to this problem would be to add a forgetting mechanism in order to prune internal

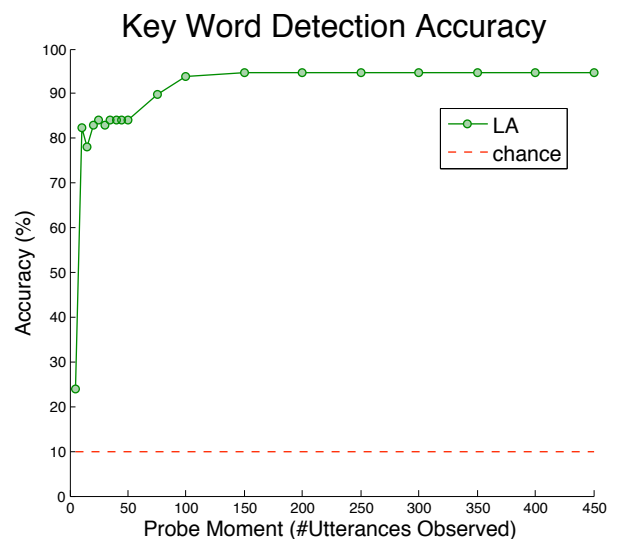


Figure 5: *Keyword detection accuracy as a function of the number of utterances observed. The green plot with circles displays LA’s detection accuracy and the red dotted plot displays chance (10%).*

representations that are not useful, where usefulness is classed as how often it recognises segments.

Table 2 displays the top and bottom twelve internal representations (tables 2(a) and 2(b) respectively) out of a total of 168 that have been built within LA’s LTM after observing all training utterances. The internal representations are ranked (R) in order of semantic stability (Stab), which was calculated using Eq. 6. The centroid token (Cent) of each cluster is displayed along with its cluster index (C). Observing the top twelve we can see that all of the ten keywords have emerged as the strongest clusters. It is also interesting to see the other structures that have emerged, for example multiple word units such as ‘s_on_the’, single word units such as ‘_reading’ and sub-word units such as ‘ing_’.

(a) Top twelve reps				(b) Bottom twelve reps			
LTM				LTM			
R	C	Cent	Stab	R	C	Cent	Stab
1	27	angus	112	157	11	s_on.the	0
2	38	daddy	89	158	24	_has_	0
3	45	_a.bath	89	159	28	s.a.b	0
4	97	_car	84	160	37	_co	0
5	34	_sho	80	161	50	ook_	0
6	22	_bottle	67	162	56	s_on	0
7	13	nappy	48	163	15	_reading	0
8	18	telephon	48	164	84	ing_	0
9	20	mummy	48	165	85	_you_	0
10	108	book	42	166	86	are_	0
11	32	the_	23	167	108	_to_	0
12	7	_is_	21	168	111	_sits_	0

Table 2: Top (a) and bottom (b) twelve ranked clusters of 168, in order of semantic stability after observing all training utterances. It can be seen that the top 10 clusters are the keywords

4. Conclusion and discussion

In this paper, we have introduced a novel computational model of early language acquisition. Our model automatically segments speech into word-like units and derives meaning through cross-modal association. We have also presented an innovative method for comparing theoretical ideas on human development with a computational learning algorithm that is cognitively motivated. The results show that the system displays similar emergent behaviour as the DST theory of human development. Comparing the models behaviour with DST it successfully discovers keywords through self-organisation, gains knowledge without any pre-specified linguistic rules and builds internal representations which are continuously evolving with varying stability.

The results show that LA successfully builds internal representations of keywords and is able to distinguish non-keyword representations by their semantic noisiness and flat epigenetic landscape. This information would allow us to make the system more computationally efficient by reducing the size of in-

ternal representations by getting rid of or forgetting *unimportant* clusters (for this task).

Some developmental theorists believe that the DST perspective is useful for solving general problems but argue that the range of different cognitive behaviours is too great (Aslin, 1993, Port, 2000), and that it is difficult to incorporate non-observable influences such as motivation. However, for this task, the epigenetic landscape is a useful and novel tool for intuitively visualising the emergence and evolution of internal representations of a cognitively motivated computational model.

5. Further work

Experimental data shows that young language learners become faster at recognising words with experience (Swingley et al., 1999). This could be due to the development of abstract models of word representations, allowing the system to generalise. Currently, the system is using the median token of each cluster for recognition. This means that the system is building an ever increasing list of exemplar tokens, but is not taking advantage of the acoustic variation within the cluster. In order to do so it would either need to use all the tokens stored in the cluster or use a mean representation. The former is not computationally viable as the token list increases to infinity, and the latter is at the expense of accuracy. However, using a mean representation would concur with developmental data showing that infants lose the ability for finer phonetic discrimination with age.

Further work will also include the discovery of the fundamental units of speech. Theories suggest that language learners try to encode information from their environment in the most efficient way i.e. through compression (Wolff, 1982). It is hypothesised that the learner begins life discovering exemplar representations of commonly re-occurring units of speech (e.g. sentences, words, syllables etc.) and then builds prototypic models of them (i.e. an average of the units in memory). LA attempts to learn in the most efficient way, therefore, patterns are discovered from a large to small scale. This means that during the early stages of language development, the infant will predominantly use internal representations of sentences and words before it has an optimised lexicon for its native language. We believe that the word-spurt phenomena would be replicated in our model with this learning mechanism in place. When the system has a robust lexicon of the fundamental units then new words can be composed by concatenating these models rather than starting with an ever-increasing list of exemplar units.

Acknowledgements

This research was funded by the European Commission, under contract number FP6-034362, in the ACORNS project (www.acorns-project.org).

References

- Aimetti, G. (2009). Modelling early language acquisition skills: Towards a general statistical learning mechanism. In *Proceedings of the Student Research Workshop at EACL 2009*, pages 1–9. Association for Computational Linguistics.
- Aslin, R. N. (1993). *Dynamic Systems in Development: Applications*, chapter The strange attractiveness of dynamic systems to development, pages 385–399. Cambridge, MIT Press.
- Chomsky, N. (1975). *Reflections on Language*. New York: Pantheon Books.
- Evans, J. L. (2007). *Blackwell Handbook of Language Development*, chapter 7. The Emergence of Language: A Dynamical Systems Account, pages 128–147. Blackwell Publishing.
- Jones, D. M., Hughes, R. W., and Macken, W. J. (2006). Perceptual organization masquerading as phonological storage: Further support for a perceptual-gestural view of short-term memory. *Journal of Memory and Language*, 54(2):265–281.
- Kelso, J. A. S. (1995). The self-organization of brain and behavior. In *Dynamic Patterns*, chapter 2, pages 46–53. MIT Press.
- Kuhl, P. K. (2004). Early language acquisition: cracking the speech code. *Nature*, 5:831–843.
- Morrongiello, B. A., Fenwick, K. D., and Chance, G. (1998). Crossmodal learning in newborn infants: Inferences about properties of auditory-visual events. *Infant Behavior and Development*, 21(Index to Volume 21, Issue 4):543 – 553.
- Muchisky, M., Gerschkoff-Stowe, L., Cole, E., and Thelen, E. (1996). The epigenetic landscape revisited: A dynamic interpretation. *Advances in Infancy Research*, 10:121–159.
- Newell, M. K., Liu, Y.-T., and Mayer-Kress, G. (2003). A dynamical systems interpretation of epigenetic landscapes for infant motor development. *Infant Behavior and Development*, 26(4):449–472.
- Nowell, P. and Moore, R. K. (1995). The application of dynamic programming techniques to mon-word based topic spotting. *EuroSpeech '95*, pages 1355–1358.
- Park, A. and Glass, J. (2008). Unsupervised pattern discovery in speech. In *Trans. ALSP*, volume 16, pages 186–197.
- Pinker, S. (1994). *The Language Instinct*. New York: Morrow.
- Port, R. F. (2000). Dynamical systems hypothesis in cognitive science. In *Encyclopedia of Cognitive Science*.
- Räsänen, O. J., Laine, U. K., and Altosaar, T. (2009). A noise robust method for pattern discovery in quantized time-series: the concept matrix approach. In *Interspeech 2009*.
- Sankoff, D. and Kruskal, J. B. (1983). *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, chapter Finding similar portions of two sequences, pages 293–296. Addison-Wesley Publishing Company, Inc.
- Savelsbergh, G. J. P. and Van der Kamp, J. (1993). The development of coordination in infancy. *Advances in Psychology*, 97:289–317.
- Smith, L. and Yu, C. (2008). Infants rapidly learn word-referent mappings via cross-situational statistics. *Cognition*, 106:1558–1568.
- Smith, L. B. and Thelen, E. (2003). Development as a dynamic system. *Trends in Cognitive Sciences*, 7(8):343 – 348.
- Stouten, V., Demuyne, K., and Van hamme, H. (2007). Automatically learning the units of speech by non-negative matrix factorisation. In *Interspeech 2007*.
- Swingle, D., Pinto, J. P., and Fernald, A. (1999). Continuous processing in word recognition at 24 months. *Cognition*, 71:73–108.
- ten Bosch, L. and Cranen, B. (2007). A computational model for unsupervised word discovery. In *INTERSPEECH 2007*.
- Thelen, E. and Fisher, D. M. (1982). Newborn stepping: An explanation for a “disappearing reflex”. *Developmental Psychology*, 18:760–775.
- Thelen, E. and Smith, L. B. (1995). A dynamic systems approach to development of cognition and action. *Journal of Cognitive Neuroscience*, 7(4):512–514.
- Waddington, C. H. (1957). *The strategy of the genes*. London: Allen & Unwin.
- Wolff, J. G. (1982). Language acquisition, data compression and generalization. *Language and Communication*, (2):57–89.

Interactions between Motivation, Emotion and Attention: From Biology to Robotics

Christian Balkenius¹

Jan Morén²

Stefan Winberg¹

¹Lund University Cognitive Science
Kungshuset, Lundagård
S-222 22 Lund, Sweden

²Graduate School of Informatics
Kyoto University, Japan

Abstract

A model of emotional conditioning is extended with a cortical model where stimulus codes compete for activation. This system is combined with motivational inputs that modulate both sensory and emotional processing. The extended model is able to reproduce the attentional blocking effect. It can also learn to switch between different sensory targets when the motivational state changes. The relation between motivation, emotion, and attention control is learned through the presentation of different stimuli in combination with reward. The model has also been used to control saccades in a stereo vision head that learns what object are compatible with what motivations.

1. Introduction

Although the importance of emotions (or value systems) have been stressed for autonomous systems (Huang and Weng, 2002; Canamero, 2003), it is seldom discussed in relation to stimulus selection. However, a robot with multiple goals and motives must be able to learn what objects are useful for each of its activities. This is even more important for a developing system where object representations and more complex motivations need not be present initially.

Stimulus selection can be carried out by assigning an emotional value to each stimulus depending on how well it satisfies each motivation. The development of this ability in children has only recently come into focus (Hajcak and Dennis, 2009), but it has been studied extensively in animals. The assigning of an emotional value to a stimulus is most likely controlled by classical conditioning and is believed to take place in the amygdala and related structures in the brain (Rolls, 1995; LeDoux, 1995). As a result of this learning, processing of motivationally significant stimuli is enhanced (Morris and Dolan, 2001).

Classical conditioning is often assumed to play a secondary role in the control of action, it can be sufficient on its own in situations where only a single action is needed (Balleine and Dickinson, 1998). For example, once the target stimulus has been selected, an innate appetitive system could be responsible for approaching the target and eventual consummation (Balkenius, 1995). In this case, all that is needed of the organism is that the correct target stimulus has been selected. It is also necessary to decide whether the target should be approach or not.

We have extended a computational model of the amygdala with two mechanisms that makes this possible. The first is a selection mechanisms that determines what stimuli should be allowed to remain active and the second is an approach system that is activates by the output from the amygdala. It is assumed that a stimulus that has been paired with reward is worth approaching. We have also added a mechanism that relates this choice as well as the emotional reactions to the current motivational state. Although a lot about the brain system for stimulus evaluation is not known, a number of computational models have been proposed and these can be used as a basis for a robot implementation of an emotional system.

The system level model is centered around the function of the amygdala which is a system of interrelated nuclei within the temporal lobe that is responsible for the conditioning of emotional reactions to previously neutral stimuli (Rolls, 1995; LeDoux, 1995). The extended amygdala is involved both appetitive (Waraczynski, 2006) and aversive (LeDoux, 1995) learning. The amygdala is not only involved in the generation of emotional reactions, it also plays a role in the modulation of different processes in other parts of the brain.

One effect of an emotional reaction in the amygdala may be to modify the cortical coding of emotion-

ally charged stimuli Vuilleumier and Huang (2009). The size of the cortical code for a stimulus increases with repeated presentation to allow a larger set of cells in cortex to be tuned to the specific properties of the stimulus. This effect is enhanced if the presentation is combined with an emotional reaction. Weinberger (1995) has shown that the cortical area representing a stimulus increases in size when it takes part in emotional conditioning. This process is thought to be controlled by the back-projection from the amygdala to cortex through the nucleus basalis of Meynart (nbM) (Weinberger, 1995). Through these connections, the amygdala could modulate learning in the sensory system based on how emotional the current situation is (LeDoux, 1996) by non-specifically controlling the level of acetylcholin in cortex.

Another effect of amygdala activity could be to bias processing in cortex towards stimuli that have an emotional significance. In contrast to the projections from cortex to the amygdala, the direct back-projections from the amygdala to cortex influence the whole of sensory cortex (LeDoux, 1996). These projections could take part in emotional priming of sensory processing by enhancing processing (or attention) to stimuli that have emotional significance (Wilson and Rolls, 1990; Wilson and Ma, 2004). This type of emotional influence on cortical processing could lead to a form of biased competition (Desimone and Duncan, 1995) where emotionally relevant stimuli are able to suppress stimuli that are not of emotional significance. This mechanism could possibly explain how emotional reactions can influence attention (Jolkkonen et al., 2002). Unlike the modulation of learning, this type of feedback to cortex must be specific to particular cells in cortex that code for the relevant aspects of the stimulus.

If emotional evaluation of stimuli influence activity in cortex, this modulation will subsequently also influence learning. A neutral stimulus that is presented together with an emotional stimulus may be suppressed and would not be able to form association with reward or punishment. This is the essence behind attentional theories of blocking (Kamin, 1968; Mackintosh, 1974; Grossberg, 1975).

The goal of the present system is to investigate how a computational architecture motivated by the interactions of different emotional and motivational structures in the brain can be used as a basis for a control system for a robot.

2. A System-Level Model

The emotion/motivation system described here is an extension of the model of the amygdala proposed by Balkenius and Morén (2000) and further developed by Morén (2002). This model is described at a system-level and is not intended to model the details within each component. Instead it aims at under-

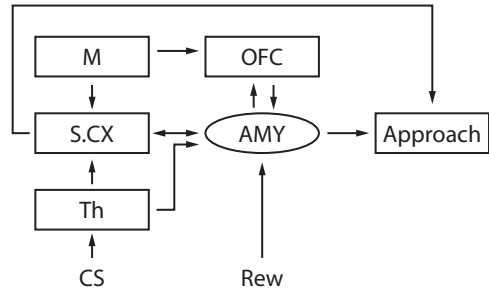


FIGURE 1: *Outline of the model. Th: thalamus, S.CX: sensory cortex, M: motivational state, AMY: amygdala, OFC: orbitofrontal cortex, Approach: approach system parameterized by the selected activity in S.CX. See text for explanation.*

standing the overall function of the system of interacting components. Here we add two important new components that drastically increase the scope of the model (Fig. 1). The first addition is the inclusion of a cortical model that allows for biased competition between cortical codes and makes attentional processing possible. The second addition is a motivational system that modulates emotional as well as sensory processing.

The model consists of a number of components named after different brain regions. In the following subsections, these labels refer to parts of the model rather than their biological counterparts.

2.1 Cortex

The cortical subsystem must have two features that are critical to the operation of the system: First, it is necessary that multiple cortical codes can be simultaneously active. For example, if the robot is simultaneously looking at two different objects, it must be possible for them to activate two different set of codes in cortex. This is generally a feature of many neural network model of visual processing, but it excludes the direct use of models such as the standard self-organizing map that only allows a single activated region.

Second, the competition can be modulated by external signals. This idea was put forward by to explain attentional modulation of cortical codes (Duncan, 1998), but is also applicable to the emotional or motivational influence on cortex.

There are several ways to implement these ideas and here we have chosen the following. Let I_i be one input to cortex and let x_i be the corresponding cortical activity. To simplify the description, we assume here that one input component is directly responsible for the activation of one particular cortical code. The competition for activation is modeled in three steps. First x_i is set to the input I_i . Second, the current bias $B_i(t)$ is calculated as

$$B_i(t) = \beta + \sum_j m_j(t)b_{ji}(t) \quad (1)$$

where β is the bias when no motivational input is available. This makes sure that all inputs have a chance of activating cortex. Finally, the following recurrence relation is iterated until the values of x_i converge:

$$\xi_i(t) = [f(x_i(t)B_i(t) - \theta)]^+ \quad (2)$$

$$x_i(t+1) = \frac{\xi_i(t)}{\|\xi(t)\|} \quad (3)$$

where $f(x) = x^2$, θ is a threshold and $[x]^+$ indicates that the value of x must be zero or larger. As a result, the cortical activity will be normalized and activity levels that falls below the threshold will be removed.

2.2 Amygdala

The amygdala is modeled as an associator that connect its input vector to a single emotional output. This output can also be inhibited by the output from the orbitofrontal cortex. A complete description of the amygdala model can be found in (Morén, 2002). Here we only summarises the required equations without justification.

The output from the amygdala E is calculated as

$$E(t) = \left[\sum_i x_i(t)V_i(t) - E_O(t) \right]^+ \quad (4)$$

where I_i is the input from cortex and thalamus, E_O is the input from OFC, and V_i are connection weights that are updated according to

$$\delta V_i(t) = \alpha x_i(t - \tau) \left[R(t) - \sum_i I_i(t - \tau) \right]^+ \quad (5)$$

The reward (or US) is given by R and α is the learning rate. The optimal inter-stimulus interval is given by τ .

2.3 Orbitofrontal Cortex

The orbitofrontal model receives as input the current cortical state and the current motivational state M and learns to inhibit and emotional reaction when it is not appropriate. As for the amygdala model, the full explanation and justification for the equations can be found in Morén (2002), but we include them here for completeness.

The output from the orbitofrontal cortex is given by

$$E_O(t) = \sum_{ij} T_{ij}(t)W_{ij}(t) \quad (6)$$

where $T_{ij} = x_i(t)M_j(t)$ and W_{ij} are the connection weights that are updated according to

$$\delta W_{ij}(t) = \beta T_{ij}(t - \tau)R_O(t) \quad (7)$$

The learning rate is set by β and R_O is the reward function. When $R \neq 0$, the reward is set to

$$R_O = \left[\sum_i x_i(t)V_i(t) - R \right]^+ - \sum_i T_{ij}(t)W_{ij}(t) \quad (8)$$

and otherwise the following equation is used

$$R_O = \left[\sum_i x_i(t)V_i(t) - \sum T_{ij}(t)W_{ij}(t) \right]^+ \quad (9)$$

2.4 Motivational System

The motivational system is here modeled as a single vector M where the level of each component indicates the strength of the corresponding motivational state. We do not model the dynamics of the different motivations here and M can thus be seen as an input to the system.

2.5 Approach System

The approach system is based on the notion of attention as selection-for-action (Allport, 1990; Hannus et al., 2005). The approach system is assumed to lead the robot toward a stimulus either by locomotion or by reaching for it. Here, we leave it unspecified what exact approach mechanism is used as long as its actions can be parameterized based on the currently coded stimuli in the cortex.

It is possible that other learning mechanisms are used within the approach system to learn the sensory-motor transformations necessary to approach the target stimulus. The approach system can be seen as a part of a more general appetitive subsystem (See Balkenius 1995 for an overview).

The approach system has two inputs. The first is the cortical coding of the target stimulus that is used to direct the produced action. The second is the emotional output from the amygdala that activates the behavior itself. It is possible that the approach system produces a behavior directed at the stimulus currently in focus even without emotional activation, but it will be less vigorous and will habituate if it does not lead to a reward (Balkenius, 2000).

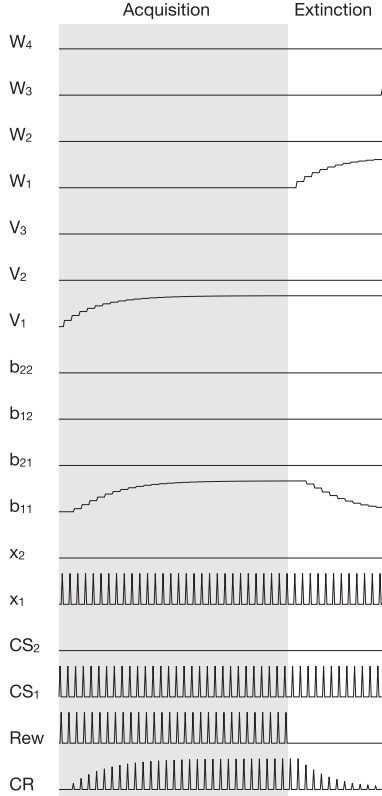


FIGURE 2: *Simulation of acquisition and extinction of a conditioned emotional response. The connections weights are shown in the lines marked with W_i and V_i , b_{ji} are the bias weights, x is the cortical activity, CS are the sensory inputs, Rew is the reward, and CR is the conditioned emotional response.*

3. Computer Simulations

In this section we show the results of a number of computer simulations of the complete system to illustrate its function in different situations. All simulations were run using the Ikaros system (Balkenius et al., 2009). Only two stimuli and two motivational states were used to simplify the results but all simulations can be extended with larger number of stimuli.

3.1 Stimulus Evaluation

This simulation shows how the model can learn which stimuli predict reward and how these predictions can change over time. This is an example of standard classical conditioning and extinction (Pavlov, 1927). The developments of the different values over time are shown in Fig. 2.

During the acquisition phase the stimulus CS_1 is paired with the reward (Rew) and as a result the associative weight V_1 increases and allows the stimulus to produce the conditioned emotional response CR . The motivational bias b_{11} also increases to enhance attention to the stimulus in cortex. During the extinction phase, the stimulus is presented on its

own and as a result the inhibitory modulation from OFC (W_1) will increase and suppress the emotional response. At the same time, the motivational bias b_{11} will decrease again.

Note that after extinction, the emotional response is only inhibited by the current motivational state through orbitofrontal cortex. If the motivation changes, the behavior can thus reappear quickly which is what happens in animals when the extinction context changes (Bouton and Nelson, 1998; Morén, 2002).

The emotional reaction that is produced can be seen as an evaluation of the stimulus. A larger reaction indicates a more valuable stimulus which should be attended as closely as coded by the bias and approached as vigorously as coded by the CR which in turn will activate the approach system.

3.2 Attentional Blocking

Blocking is the well known phenomenon that a stimulus that is presented together with an already conditioned stimulus will not acquire an association with the reward (Kamin, 1968). This result was originally explained as a result of attentional competition (Mackintosh, 1974), but has later mainly been explained as a competition for association with the reward (Rescorla and Wagner, 1972).

We simulated this phenomenon with the model as illustrated in Fig. 3. The system is assumed to be in motivational state 1. First CS_1 is presented together with the reward a number of times which leads to increases in V_1 and b_{11} . This is followed by the presentation of both CS_1 and CS_2 together with the reward. As V_1 has already reached its asymptotic value, no learning occurs in this phase. Finally, CS_1 and CS_2 are individually tested and as can be seen only CS_1 produces an emotional response.

In addition to showing that the system reproduces the blocking phenomenon, the simulation also illustrates that the two theories of blocking are not mutually exclusive since the system incorporates both mechanisms. Attentional blocking is used to select emotionally relevant stimuli in cortex and competition for associative strength is used within the amygdala to limit the range of the learned associations. Attentional selection of this type is essential in more complex learning situations as illustrated by the next simulation.

3.3 Switching

This simulation demonstrates that the model will select a stimulus that has been paired with reward in the current motivational state when several stimuli are simultaneously present (Fig. 4). First the system is conditioned with CS_1 in motivational context M_1 , then it is conditioned to CS_2 in motivation M_2 .

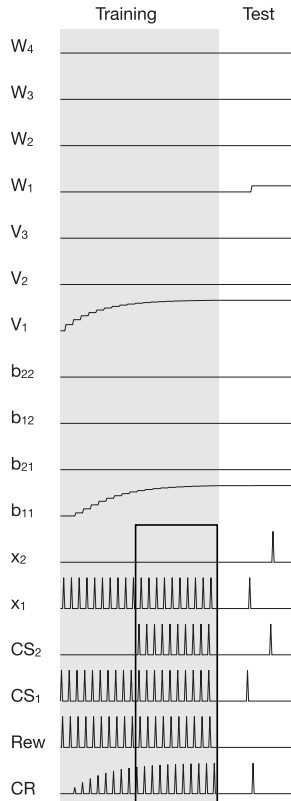


FIGURE 3: *Attentional blocking. The frame marks the compound conditioning trials that are followed by two tests with CS_1 and CS_2 respectively. Refer to Fig. 2 for an explanation of the different labels.*

Finally, the system is tested in motivational state M_1 with both stimuli. As can be seen in the figure, only the stimulus compatible with motivation 1, that is CS_1 , is active in cortex. When the motivational state is subsequently switched to M_2 , the cortical code also switches to activate stimulus CS_2 instead.

In this simulation, there is only a positive bias from the motivation on the cortical activity. The selection is the result of biased competition within cortex. The reason why W_1 and W_4 increases is that the associations undergo extinction during the final test phase.

We also tested a slightly more complicated situation where both stimuli were presented in both motivational contexts, but only rewarded in one (Fig. 5). There are four distinct training regimes that are repeated four times:

- $M_1 : CS_1 + Rew$
- $M_2 : CS_1$
- $M_1 : CS_2 + Rew$
- $M_2 : CS_2$

As a result, both stimuli undergo extinction in one motivational context. Finally, the system is tested in

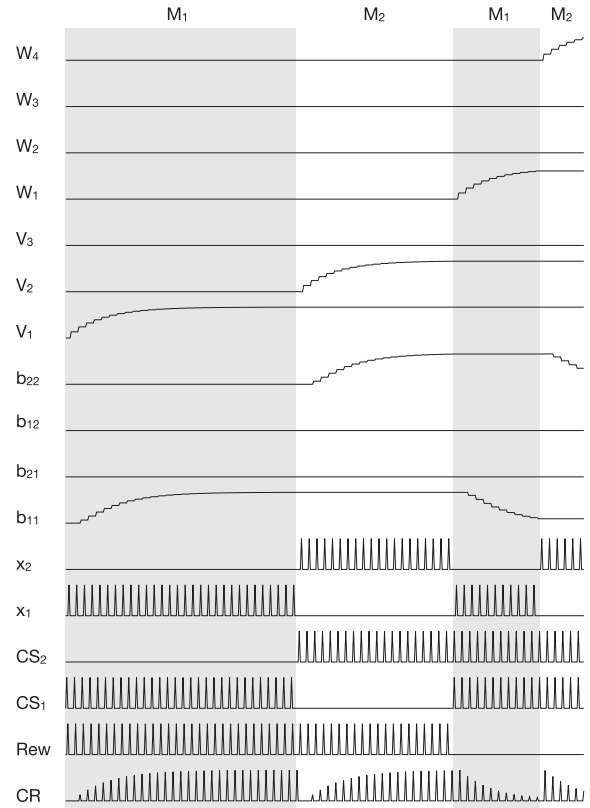


FIGURE 4: *Switching of attention as a function of motivational state. See Fig. 2 for an explanation of the different labels.*

motivational state M_1 with both stimuli present. Initially, the system selects stimulus CS_1 , but since it does not receive any reward, the emotional response CR gradually decreases. At the same time, the bias for CS_1 in M_1 decreases until the system is ready to try out CS_2 instead. At this point, the emotional response is very weak, but will be sufficient for the system to investigate CS_2 if there is no other stimulus present. Since CS_2 is not rewarded either, the system then alternates back to CS_1 a second time and then back to CS_2 , before the emotional reaction is completely extinguished.

This simulation shows that the model can learn what stimulus satisfies which motivation also in situations with inhibition. The cortical competition is essential in this case since the inhibiting stimulus would otherwise have shut off the emotional reaction for the stimulus that was conditioned in the current motivational state.

4. A Robot Implementation

Preliminary tests have been performed with a stereo head that is controlled by the model described above (Fig. 6). The same code was used as in the simulations above except that the input and outputs were connected through a number of extra Ikaros modules

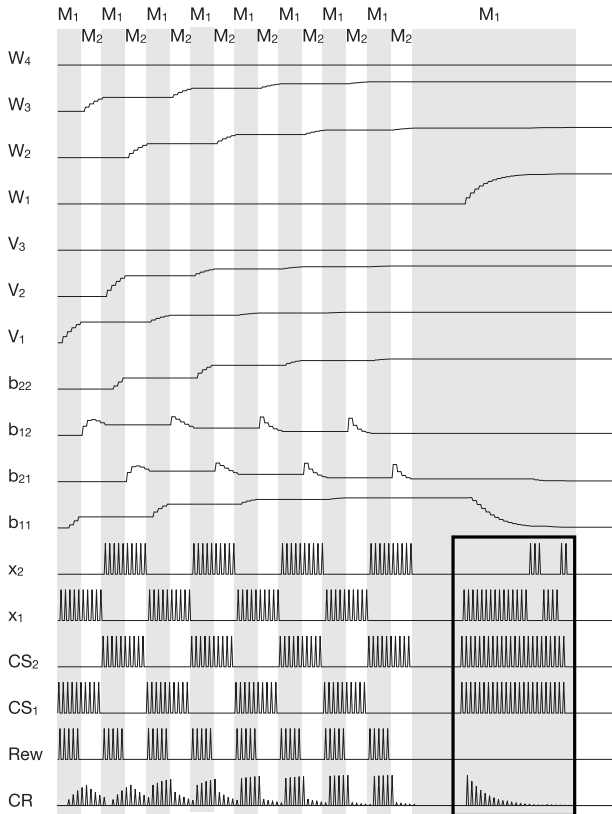


FIGURE 5: *Switching of attention after repeated acquisition and extinction in different motivational contexts. (See Fig. 2 for an explanation of the different labels.)*

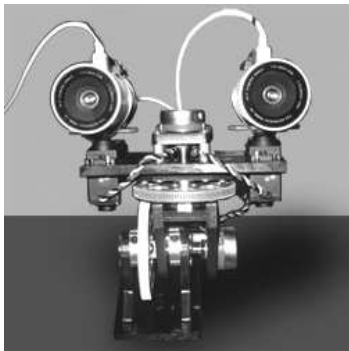


FIGURE 6: *A stereo head used to test the system with visual stimuli.*

to a robot head.

In addition, the two cortical nodes are replaced with a full saliency map where different features at different locations compete for activation. The implementation is based on the learning saliency map described by Balkenius et al. (2008), which in turn was inspired by the work of Itti and Koch (2001).

The saliency map is used to select the target stimulus which is then used as a parameter for the approach system which in this case is the saccade system that moves the gaze toward the selected stimulus. No other action is currently implemented.

In the tests, objects of two different colors were used as stimuli and the saliency map was trained exactly as in the switching simulation described above. The reward and motivational state was externally supplied and the reward was triggered by a saccade to the correct stimulus.

Apart from showing that the model can be used in a robot, this implementation also clearly illustrates the similarity between cortical competition, attentional blocking and saliency maps.

5. Discussion

The features of a motivational/emotional system that we have described above are essential in a robotic system that develops its cognitive abilities over time. As it encounters new objects in the environment or learn new skills, it needs to learn what actions and objects fit which motivations. We believe that the system-level model developed here contains several fundamental components of such a system. The model has a number of attractive properties. It is reasonably similar to the corresponding system in animals and it is computationally sound while incorporating a number of useful mechanisms. It bridges the gap between models of conditioning and models of attention control to allow classical learning mechanisms to control attention. It also suggests a way to incorporate modulation by the current motivational state on sensory and emotional processing.

The results are comparable to our earlier modeling of task-switching in instrumental learning (Balkenius and Winberg, 2004). The main difference is that there is only one action in this system and that cortical competition between different stimuli is necessary here while that is not the case in the instrumental situation. The model resembles the model proposed by Mannella et al. (2007) in that it includes interaction between motivational states and emotional responses, and learns to select behavior depending on the motivational state. The present model is different in that it attempts to explain the modulation of sensory processing rather than action selection.

There are a number of additional components that will need to be added for a more flexible and less specific learning ability. The components should work in close cooperation with the subsystems described here. The approach system needs to habituate when no reward is received. A mechanism that can handle this was described by Balkenius (2000). It is also necessary to include instrumental learning to allow learning of flexible behavior sequences. Instrumental learning could also in principle be under motivational control in the same way as emotional learning although the situation is probably more complex in humans and animals (Balleine and Dickinson, 1998). Both instrumental and classical conditioning could additionally interact with habit systems that learn

to produce behavior after repeated rehearsals.

There is also a need for adaptive sensory-motor mappings that learn goal-directed behavior that the other learning systems can activate or inhibit. One final feature that is missing in the present system is a mechanism that can handle incentive motivation (see Balkenius 1995). This is something that we want to add in the future.

We will also extend the robotic implementation of the system to include some form of manipulation and not only a head. The general structure of the control will be very similar to what we have now, except that the appetitive approach behavior will contain several segments such as visual fixation followed by reaching and possibly exploration or ingestion.

In summary, we have shown how a model of emotional conditioning can be extended with multiple motivations to learn what stimuli are rewarding in each motivational state. In addition, we have shown how the extended model can handle competition for attention in a cortical subsystem. Both these abilities are essential for a robot that is engaged in many different activities motivated by different goals or needs.

Acknowledgements

We gratefully acknowledge support from the Linnaeus grant Thinking in Time: Cognition, Communication and Learning that is financed by the Swedish Research Council.

References

- Allport, A. (1990). Visual attention. In Posner, M. I., editor, *Foundations of Cognitive Science*. MIT Press, Cambridge, MA.
- Balkenius, C. (1995). *Natural Intelligence in Artificial Creatures*. Number 37 in Lund University Cognitive Studies.
- Balkenius, C. (2000). Attention, habituation and conditioning: toward a computational model. *Cognitive Science Quarterly*, 1(2):171–214.
- Balkenius, C., Förster, A., Johansson, B., and Thorsteinsdottir, V. (2008). Anticipation in attention. In Pezzulo, G., Butz, M. V., Castelfranchi, C., and Falcone, R., editors, *The Challenge of Anticipation*, volume 5225 of *Lecture Notes in Artificial Intelligence*, pages 65–83. Springer, Berlin.
- Balkenius, C. and Morén, J. (2000). Emotional learning: A computational model of the amygdala. *Cybernetics and Systems*, 32(6):611–636.
- Balkenius, C., Morén, J., Johansson, B., and Johansson, M. (2009). Ikaros: Building cognitive models for robots. *Advanced Engineering Informatics*, doi: 10.1016/j.aei.2009.08.003.
- Balkenius, C. and Winberg, S. (2004). Cognitive modeling with context sensitive reinforcement learning. In *Proceedings of AILS '04*, Lund. Dept. of Computer Science.
- Balleine, B. W. and Dickinson, A. (1998). Goal-directed instrumental action: contingency and incentive learning and their cortical substrates. *Neuropharmacology*, 37:407–419.
- Bouton, M. E. and Nelson, J. B. (1998). Mechanisms of feature-positive and feature-negative discrimination learning in an appetitive conditioning paradigm. In Schmajuk, N. and Holland, P. C., editors, *Occasion setting: Associative learning and cognition in animals*, pages 69–112. American Psychological Association, Washington, DC.
- Canamero, D. (2003). Designing emotions for activity selection. In Trapp, R., Petta, P., and Payr, S., editors, *Emotions in Humans and Artifacts*, pages 115–148. MIT Press.
- Desimone, R. and Duncan, J. (1995). Neural mechanisms of selective visual attention. *Annu. Rev. Neurosci.*, 18:193–222.
- Duncan, J. (1998). Converging levels of analysis in the cognitive neuroscience of visual attention. *Phil. Trans. R. Soc. Lond. B*, 353(1373):1307–1317. 09628436.
- Grossberg, S. (1975). A neural model of attention, reinforcement, and discrimination learning. *International Review of Neurobiology*, 18:263–327.
- Hajcak, G. and Dennis, T. A. (2009). Brain potentials during affective picture processing in children. *Biol Psychol*, 80(3):333–8.
- Hannus, A., Cornelissen, F., Lindemann, O., and Bekkering, H. (2005). Selection-for-action in visual search. *Acta Psychologica*, 118(1-2):171–191.
- Huang, X. and Weng, J. (2002). Novelty and reinforcement learning in the value system of developmental robot. In Prince, C. G., Demiris, Y., Marom, Y., Kozima, H., and Balkenius, C., editors, *Proceedings of the Second International Workshop on Epigenetic Robotics*, volume 94. Lund University Cognitive Studies.
- Itti, L. and Koch, C. (2001). Feature combination strategies for saliency-based visual attention systems. *Journal of Electronic Imaging*, 10(1):161–169.

- Jolkkonen, E., Miettinen, R., Pikkarainen, M., and Pitkanen, A. (2002). Projections from the amygdaloid complex to the magnocellular cholinergic basal forebrain in rat. *Neuroscience*, 111(1):133–149.
- Kamin, L. J. (1968). Attention-like processes in classical conditioning. In Jones, M. R., editor, *Miami symposium on the prediction of behavior: aversive stimulation*, pages 9–31. University of Miami Press, Miami.
- LeDoux, J. E. (1995). In search of an emotional system in the brain: leaping from fear to emotion and consciousness. In Gazzaniga, M. S., editor, *The cognitive neurosciences*, pages 1049–1061. MIT Press, Cambridge, MA.
- LeDoux, J. E. (1996). *The emotional brain*. Simon and Schuster, New York.
- Mackintosh, N. J. (1974). *The psychology of animal learning*. Academic Press, New York.
- Mannella, F., Miroli, M., and Baldassarre, G. (2007). The role of amygdala in devaluation: A model tested with a simulated rat. In Berthouze, L., Prince, C. G., Littman, M., Kozima, H., and Balkenius, C., editors, *Proceedings of the Seventh International Conference on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*, volume 135. Lund University Cognitive Studies.
- Morén, J. (2002). *Emotion and Learning - A Computational Model of the Amygdala*. Lund University Cognitive Studies, 93.
- Morris, J. S. and Dolan, R. J. (2001). Involvement of human amygdala and orbitofrontal cortex in hunger-enhanced memory for food stimuli. *Journal of Neuroscience*, 21(14):5304–5310.
- Pavlov, I. P. (1927). *Conditioned Reflexes*. Oxford University Press, Oxford.
- Rescorla, R. A. and Wagner, A. R. (1972). A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. In Black, A. H. and Prokasy, W. F., editors, *Classical conditioning II: Current research and theory*, pages 64–99. New York: Appleton-Century-Crofts.
- Rolls, E. T. (1995). A theory of emotion and consciousness, and its application to understanding the neural basis of emotion. In Gazzaniga, M. S., editor, *The cognitive neurosciences*, pages 1091–1106. MIT Press, Cambridge, MA.
- Vuilleumier, P. and Huang, Y.-M. (2009). Emotional attention: Uncovering the mechanisms of affective biases in perception. *Current Directions in Psychological Science*, 18(3):148–152.
- Waraczynski, M. (2006). The central extended amygdala network as a proposed circuit underlying reward valuation. *Neuroscience and Biobehavioral Reviews*, 30(4):472–496.
- Weinberger, N. M. (1995). Retuning the brain by fear conditioning. In Gazzaniga, M. S., editor, *The cognitive neurosciences*, pages 1071–1089. MIT Press, Cambridge, MA.
- Wilson, F. and Ma, Y.-Y. (2004). Reinforcement-related neurons in the primate basal forebrain respond to the learned significance of task events rather than to the hedonic attributes of reward. *Cognitive Brain Research*, 19(1):74–81.
- Wilson, F. A. W. and Rolls, E. T. (1990). Neuronal responses related to reinforcement in the primate basal forebrain. *Brain Res*, 50:213 – 231.

Adults Structure Object Demonstrations to Support Infant Attention and Learning

Rebecca J. Brand

Villanova University

Rebecca.brand@villanova.edu

Abstract

Pedagogy theory (Csibra & Gergely, 2006; 2009) suggests that adults and infants comprise a co-evolved teaching-learning system. Adults spontaneously provide “ostensive cues” which function to engage infants’ attention and indicate a learning scenario. The current paper presents evidence that infant-directed action (“motionese”) and speech-action alignment (“acoustic packaging”) likely function as ostensive cues in the context of object-use demonstrations. Motionese has been documented in naturalistic learning scenarios and appears ubiquitous among adults, including non-parents. Further, infants attend more to motionese than to standard adult-directed action. On-going research is underway to evaluate the learning benefits (e.g., enhanced imitation) in the presence of these cues. Any information we can glean about the behavior of human adults (as natural teachers) and infants (as naive learners) supports attempts to model efficient learning in robots.

1. Introduction

Human infants treat other human agents as a special stimulus, paying more attention to stimuli with faces, hands, and particular kinds of human-like movements than to other kinds of stimuli (e.g., Frank, Vul, & Johnson, 2008; Legerstee, 2005; Simion, Regolin, & Bulf, 2008). Theories to explain this include Tomasello’s (Tomasello & Carpenter, 2007) “shared intentionality” theory, which argues that babies are fundamentally motivated by sharing attention and goals with others and Meltzoff’s (2007) “like me” theory, which argues that babies have an innate system that allows them to map their own movements to the movements of others and vice-versa. A third theory, the theory of “pedagogy,” (Csibra & Gergely, 2006) focuses on specific cues that garner infant attention and flag an interaction as a “learning” scenario. These so-called “ostensive cues,” such as eye contact and calling the infant’s own name, are not only the cues infants preferentially attend to but are also the cues that adults spontaneously emit when interacting with babies. According to this theory, adults and babies – or human teachers and learners more generally – comprise a co-evolved system. All three of these ideas (intrinsic motivation, like-me mapping, and humans’ natural teaching tendencies) have been explored recently in

epigenetic robotic systems (see Thomaz & Breazeal, 2008; Oudeyer, Kaplan, & Hafner, 2007).

The work described in this talk investigates a set of behaviors that we think function as ostensive cues when adults teach babies how to interact with new tools and objects. This research provides additional information about the human teaching-learning system, and can thus support the on-going attempts to model learning in robots, such as the socially-guided machine learning theory proposed by Thomaz & Breazeal (2008).

The set of behaviors explored here includes so-called “motionese” or “infant-directed (ID) action” (Brand, Baldwin & Ashburn, 2002) as well as “acoustic packaging” or “speech-action alignment” (Meyer, Hard, Brand, & Baldwin, 2008). Below, I will describe the cues themselves, as they are used by adults in various contexts, as well as evidence that these cues provide benefits to infants’ attention and learning.

2. Infant-Directed Action Modifications (“Motionese”)

The first studies of infant-directed (ID) object demonstrations attempted to establish a set of action modifications that were relatively consistent across adults. We first asked mothers of infants to demonstrate five novel objects to either their infant or to an adult partner. Mothers were told we were investigating how people demonstrate to one another, but were not told of the infant-adult comparison. We measured their behavior on a set of eight features that we thought might function to highlight action boundaries and enhance infant attention. We found significant modifications for six features. We found that ID demonstrations of novel objects tended to be enacted with greater proximity to the partner, a larger range of motion, and more enthusiasm, interactivity, repetitiveness, and simplification when compared to adult-directed (AD) demonstrations (Brand, et al., 2002). See Figure 1.

In a second study (Brand, Shallcross, Sabatos, & Massie, 2007; See Figure 2), we attempted to measure several features in a more precise way. As a way to quantify the variable of interactivity, we measured the number and length of gaze bouts, and the number of exchanges of the object between the demonstrator and partner. As a way to further quantify simplification, we measured the number of distinct action types mothers demonstrated during each turn.

Figure 1. Motionese features as found in Brand et al., (2002). All features but *punctuation* and *rate* show significant differences between ID and AD action.

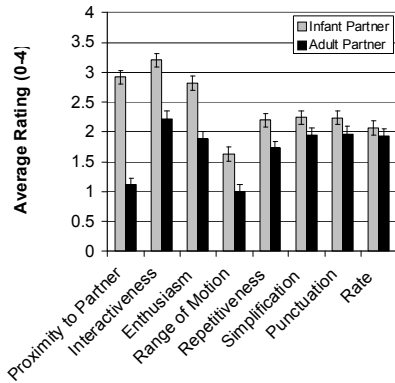
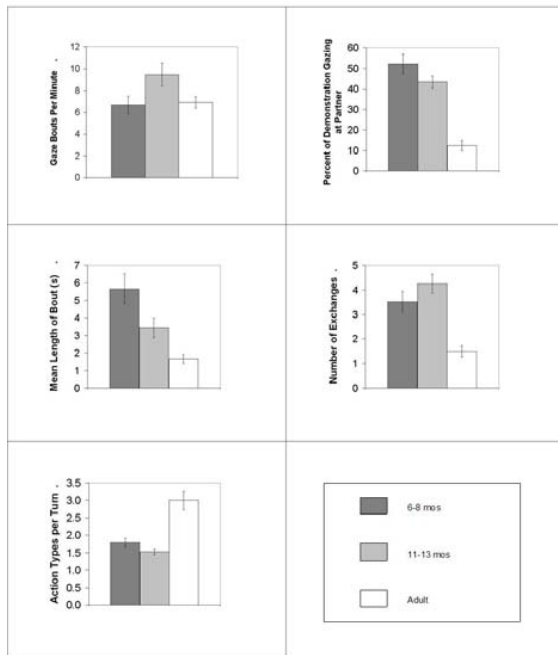


Figure 2. Modifications to eye gaze, object exchanges, and action types per turn for 6- to 8- and 11- to 13-month-olds as found in Brand et al., (2007).



These analyses revealed that ID demonstrations included more eye contact, more exchanges of the objects, and fewer action types per turn than AD demonstrations. They also provided the first evidence of mothers discriminating in their action demonstrations for infants of different age groups: 6- to 8-month-old infants, relative to 11- to 13-month-old

infants, received more eye gaze divided into fewer, longer bouts, and also received fewer turns with the object. We suspect that both of these modifications represent mothers' sensitivity to their infants' ability to control their own attention, which is developing rapidly across this period (Ruff & Rothbart, 1996).

Research from other labs confirms these findings and offers evidence of other ID action characteristics. For instance, Rohlfing, Fritsch, Wrede, & Jungmann (2006) found that demonstrations to infants were slower in pace (i.e., contained longer pauses) and used large, inefficient movements (similar to our "range of motion" variable). The work of Gogate and her colleagues (e.g., Matatyaho & Gogate, 2008) indicates that looming and shaking motions – in particular, in synchrony with utterances – characterize the actions of mothers when speaking to young infants.

As our first studies used only mothers, we wondered at the scope of these modifications. However, Rohlfing et al., (2006) also included fathers in their sample, and reported no differences between mothers and fathers. To test whether such modifications are limited to parents, we (Brand, Ragnarsson, & Casperson, 2009) asked a set of non-parent adults to demonstrate objects as if for an infant and an adult audience. We found that non-parents similarly discriminated among audiences in their demonstrations, particularly with regard to repetition, range of motion, and smiles. We also found that experience with or comfort with infants was not related to ID modifications, suggesting a minimal role for learning in the use of these modifications.

Additional recent work has focused on the structure and timing of specific cues within the set of motionese modifications. For instance, motionese comprises enhanced eye gaze and repetition; however, it is not clear precisely how these features are used or what their function is. Regarding eye gaze, we would expect mothers to make eye contact in between action units – perhaps as a way to check infant attention and/or to mark the unit onset or offset. Regarding repetition, our first hypothesis, expressed in Brand et al., (2002), was that when demonstrating a series of actions, mothers would most likely break actions into the smallest units and repeat at the unit level, rather than repeating sequences of action units. Using the global coding scheme, this is in line with what we found. However, recent insight suggested that mothers' repetitions of units versus sequences would likely depend on the nature of the object she was demonstrating: when the sequence is crucial to reach a salient goal (a so-called "enabling sequence" [Bauer, 1992]), we predicted mothers would repeat at the sequence level; when each unit could be considered a goal in itself and there was no hierarchical end goal, we predicted mothers would repeat at the unit level.

We recently collected a new sample of 40 mother-infant dyads to explore these issues. Regarding eye gaze, we found that, as predicted, onsets and offsets of

gaze and onsets and offsets of actions were more aligned than expected by chance (Brand, Hollenbeck, Kominsky, & Hard, in preparation). Regarding repetitions (Brand, McGee, Kominsky, Briggs, Grueneisen, & Orbach, in press), we tested our hypothesis by giving mothers objects of two distinct types: objects for which a sequence of three different actions was required to produce a salient end-goal, such as: push buttons, slide switch, open top of a key lockbox (see Figure 3a); and objects on which you could also perform three distinct actions, but the actions did not lead to an end-goal, such as shaking, twisting, and tilting a puzzle toy (see Figure 3b). This distinction was not mentioned to them, and the six objects were presented in random order.

Figure 3. Examples of end-goal (lockbox) and non-goal (puzzle) objects.



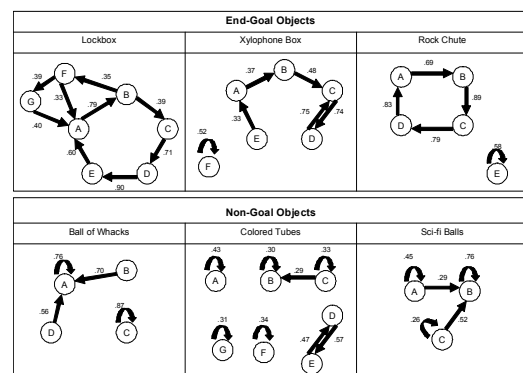
We assigned each distinct action (such as pushing buttons) a letter (e.g., A), and then transcribed mothers' demonstrations according to this coding scheme. A, B, and C always represented the three primary actions we suggested to mothers. Additional letters were added for other common actions. For instance, for the lockbox object, A was pressing the buttons, B was sliding the latch, and C was opening the top. For many mothers, they went on to take the key out, put the key back in, put the lid back on, and shake the object, so these were assigned codes D-G.

We then analyzed these transcriptions in three ways. First, we asked, of all two-unit series, what proportion were repetitions (e.g., AA) versus sequences (e.g., AB). We found that repetitions (AA) comprised a larger proportion of series on non-goal (arbitrary-sequence) objects (46%) than on end-goal (enabling-sequence) objects (13%). Next, we asked how often mothers completed the full sequence – in order and without interruption – of the three actions provided for them on the instructions (ABC). We found, across all subjects, they were more likely to repeat the full sequence for the end-goal objects than the non-goal objects. In fact, for end-goal objects, 81% of demonstrations went through the complete sequence at least one time, and 38% went through the whole sequence two or more times. For non-goal objects, only 18% of demonstrations went through the entire sequence at least once. Finally, we computed the

transitional probabilities (TPs) from any action on an object to any other action (see Figure 4). TPs represent the percentage of each action (A) followed by another action (B). Thus, high TPs from an action to itself (AA) represent repetitions of units, while sets of high TPs from one action to the next to the next in a cycle (AB, BC, CD, DA) represent repetitions of sequences.

In sum, we found that when demonstrating objects with no salient end-goal – similar to those used in previous studies – mothers tended to repeat the individual units (e.g., shake, shake, shake). However, when demonstrating objects whose actions form a coherent enabling sequence leading to a salient end-goal, we found that mothers were more likely to repeat the sequence, rather than the individual units.

Figure 4. Transitional probabilities for objects with and without a salient end-goal.



Finally, ongoing research in my lab is investigating the timing of mothers' behaviors in relation to infant attention as the interaction unfolds. To examine this, we are using sequential analysis in a sample of 11 mothers to determine the most common patterns of behavior across the demonstration (Kasparian & Brand, in progress). One common sequence involves mothers demonstrating an action, and infants subsequently beginning to attend and to manipulate the object. At this point, mothers often do nothing (merely watching) until infants lose attention, at which point they demonstrate again, often using the largest, noisiest action available. Thus mothers often act when – and only when – it is necessary to re-engage infants' attention. They seem to be sensitive to repeated disengagements by infants, however; after more than one cycle of this pattern with a given object, mothers tend to respond to loss of attention by switching to a new object.

3. Benefits of Motionese

Now that a set of action modifications has been identified, an important goal of the research is to

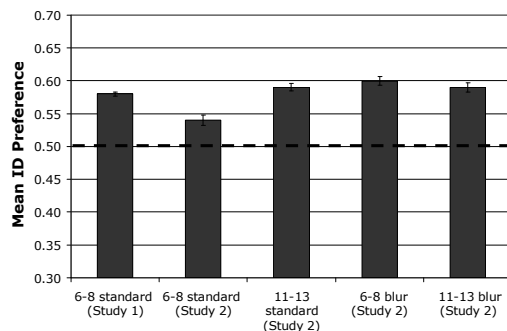
determine the degree to which these cues are relevant to infant learning. As a first step, we investigated whether these cues preferentially attract infant attention (Brand & Shallcross, 2008). Using a video preferential-looking paradigm, we showed 6- to 8- and 11- to 13-month-old infants side-by-side video clips of mothers demonstrating a single object in either an ID or an AD manner. These were videos of two different mothers – one who had been interacting with her infant, and one who had been interacting with her husband. Only the mothers and the objects (not the partners) are visible on the screen. We used still pictures to rule out the possibility that infants’ preference was based on the physical features (e.g., hair style) of the two mothers. See Figure 5.

Figure 5. Still frame of mothers demonstrating to an infant (left) and adult (right).



In Study 1, we found that 6- to 8-month-old infants preferred to look longer at the video of the ID demonstration. They did not have a corresponding preference for the still pictures. In order to determine whether infants’ preference for ID action was due solely to the mothers’ facial expressions and eye gaze, in Study 2 we tested a new set of infants after digitally blurring the faces of the mothers, leaving only the body movements visible. We tested both 6- to 8- and 11- to 13-month-olds in Study 2. We found that at both ages, and even with the facial features blurred, infants still had a preference for the ID actions. See Figure 6 for results from both studies.

Figure 6. Infants prefer motionese, even with mothers’ faces blurred (Brand & Shallcross, 2008). All conditions except “6-8 standard (Study 2)” show a preference for ID action that is significantly greater than chance (50%).



A computational model of attention (Nagai & Rohlfing, 2009) suggests that in addition to arousing greater infant attention than AD action, ID action may in fact guide infant attention to specific locations in the scene. Nagai and Rohlfing found that based on features such as motion, color, and intensity, their model “attended” most to parents’ hands and task-relevant objects (e.g., stacking cups) in both ID and AD action demonstrations. During the task, however ID action drew preferentially more attention to the face than AD action; just before and after the task, on the other hand, ID action drew preferentially more attention to the cups. The authors argue that by holding their faces still before and after the task, parents draw attention to the start and end state of the objects, thus helping infants to learn about the important change of state.

Having determined that ID action cues indeed attract infant attention, our next question is whether ID action benefits infants’ learning from the demonstrations. Several studies are underway to investigate whether children are able to imitate novel actions more faithfully if they are shown in ID action as opposed to AD action (Brand & Casperson, in progress; Brand & Jemmoua, in progress; Williamson & Brand, in progress). These imitation studies look for benefits across a wide age range (6 months to 3 years), with actions that range from a simple button-press to a complex causal sequence, and they make use of both experimentally-controlled action videos as well as mothers’ own demonstrations.

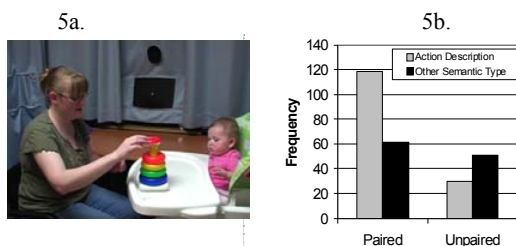
4. Speech-Action Alignment (“Acoustic Packaging”)

One additional cue that adults may use to support infants’ learning about actions is their speech. Even before infants understand the content of speech, there is reason to believe that the timing and prosody offers rich information to infants (e.g., Fernald, 1989). Further, the Intersensory Redundancy Hypothesis (Bahrick, Lickliter, & Flom, 2004) suggests that information across multiple modalities that occurs in synchrony attracts infants’ attention. The argument is that when making sense of the physical world, infants would do well to attend to events that provide redundant information across modalities; for instance, a ball bouncing on a surface will provide information about tempo and intensity in both visual and auditory modalities. Thus, this kind of co-occurrence helps infants find coherent events within the flow of information. By providing speech cues that align systematically with their actions, parents may be exploiting this tendency on infants’ part to attend to cross-modal synchrony. The use of this type of alignment has been referred to as “acoustic packaging” (Brand & Tapscott, 2007; Hirsh-Pasek & Golinkoff, 1996).

Evidence supports the claim that if parents provide such alignment, infants can make use of it. For instance, Gogate and her colleagues have shown that synchrony between words and movements of objects helps young infants learn the associations between word and object (Gogate & Barick, 1998). In our lab (Brand & Tapscott, 2007), we found that in an experimental setting, infants could learn which events were “packaged” by audio and which were not, after only a few repetitions.

To investigate parents’ use of acoustic packaging, we (Meyer, et al., 2008) offered mothers two sets of objects to demonstrate to their 6- to 14-month-old children (e.g., stacking rings). See Figure 7a. We then coded the onsets and offsets of each utterance and of each action. Based on the number of utterances and actions within each demonstration, we computed the degree to which one would expect them to be aligned just due to chance (following Zacks, Tversky, & Iyer, 2001). We found that onset and offset times were aligned more than expected by chance. Further, we divided utterances into attention-getting, goal-setting, description, and celebration, and we found that utterances that were aligned or paired with actions were more than twice as likely to be action descriptions than any other type of utterance. See Figure 7b. We suspect (and are currently testing to confirm) that the prosodic contours of these different utterance types are discriminable by infants; thus they may come to learn that a particular melodic contour tends to co-occur with salient actions. These contours may even directly affect infant attention such that it is focused on the actor just as she begins and carries out the relevant movements.

Figure 7. Mothers demonstrating a task such as stacking rings (a) tend to align (pair) their utterances with their actions, especially for utterances that are action descriptions, e.g., “And now the red one!” (b).



Recent computational work indicates remarkable alignment between speech and actions in both adult-infant and adult-adult interactions (Rolf, Hanheide, & Rohlfing, 2009; Schillingmann, Wrede, & Rohlfing, 2009; Wolf & Bungmann, 2006). To illustrate, Schillingmann et al. measured the overall amount of motion happening in each frame of a video recording and parsed it into “actions” at local motion minima. They parsed the speech at pauses. They found that

action and speech were more likely to be packaged in ID demonstrations than AD demonstrations. Rolf et al. (2009) provide a convincing model of infants’ attention being drawn by this packaging. They first determined the change in intensity in any pixel (typically representing movement across that spot) and the change in intensity in the auditory stimulus and noted the correspondences. Thus anything that moved in synchrony with a sound was highlighted in the resulting averaged map (or “mixelgram”) of synchrony. Using this technique, they again found more synchrony in ID than AD action, and in an exploratory sample of two participants, they found that synchrony tends to occur most often in the realm of the demonstrators’ face and hands, suggesting that low-level detection of synchrony by infants may indeed be effective at directing their attention to the relevant portions of an action scene.

5. Summary

Pedagogy theory argues that adult experts and infant novices are a co-evolved system for teaching and learning (Csibra & Gergely, 2006; 2009). According to this theory, adults spontaneously offer “ostensive cues” when interacting with infants, particularly in the case when they are providing information meant to be learned and generalized by infants. Here we have offered evidence that these cues extend to the object-manipulation domain. Adults – including mothers, fathers, and non-parents with a range of experience with babies – provide action that is larger, more repetitive, and simplified; that appears well-timed to capture and sustain infant attention; and that provides social-emotional cues such as eye gaze and smiles. These cues are exactly the sort predicted by pedagogy theory to trigger infants’ inherent attention and learning capacities. Evidence demonstrates that these features are sufficient for preferentially engaging infant attention; on-going work is testing whether they indeed facilitate infants’ learning from and re-enacting adult demonstrations.

Given that these features appear to arise naturally in adult teachers when interacting with infants, efforts to make teachable robots that behave in infant-like ways appears to be well-founded. For instance, both Nagai, Mull, & Rohlfing (2008) and Oudeyer et al. (2007) make use of a robot’s eye gaze to make its mental state more “transparent.” Particularly when the robot also makes eye contact, and is “cute” as in Nagai et al., this may trigger adults’ best teaching strategies. Thus, robotics and developmental researchers can continue to provide mutually beneficial insights about the optimum systems for teaching and learning.

References

Bahrick, L. E., Lickliter, R., & Flom, R. (2004). Intersensory redundancy guides the development of

- selective attention, perception, and cognition in infancy. *Current Directions in Psychological Science*, *13*, 99-102.
- Brand, R. J., Baldwin, D. A., & Ashburn, L. A. (2002). Evidence for 'motionese': modifications in mothers' infant-directed action. *Developmental Science*, *5*, 72-83.
- Brand, R.J., McGee, A., Kominsky, J., Briggs, K., Grueneisen, A., & Orbach, T. (in press). Repetition in infant-directed action depends on the goal structure of the object: Evidence for statistical regularities. *Gesture*.
- Brand, R.J., Hollenbeck, E., Kominsky, J., & Hard, B. (2009). *Mothers' speech- gaze alignment in demonstrations to infants*. Manuscript in preparation.
- Brand, R. J., Ragnarsson, K. A., & Casperson, C. (2009, April). *Non-parents use motionese when demonstrating objects for infants*. Poster presented at the Society for Research in Child Development, Denver, CO.
- Brand, R. J., & Shallcross, W. L. (2008). Infants prefer motionese to adult-directed action. *Developmental Science*, *11*, 853-861.
- Brand, R. J., Shallcross, W. L., Sabatos, M. G., & Massie, K., P. (2007). Fine-grained analysis of motionese: Eye gaze, object exchanges, and action units in infant- versus adult-directed action. *Infancy*, *11*, 203-214.
- Brand, R. J. & Tapscott, S. (2007). Acoustic packaging of action sequences by infants. *Infancy*, *11*, 321-332.
- Csibra, G., & Gergely, G. (2006). Social learning and social cognition: The case for pedagogy. In Y. Munakata & M. H. Johnson (Eds.), *Processes of Change in Brain and Cognitive Development, Attention and Performance, XXI*. Oxford: Oxford University Press, p. 249-274.
- Csibra, G., & Gergely, G. (2009). Natural pedagogy. *Trends in Cognitive Sciences*, *13*, 148-153.
- Fernald, A. (1985). Four-month-old infants prefer to listen to motherese. *Infant Behavior and Development*, *8*, 181-195.
- Frank, M. C., Vul, E., & Johnson, S. P. (2009). Development of infants' attention to faces during the first year. *Cognition*, *110*, 160-170.
- Hirsh-Pasek, K. & Golinkoff, R. (1996). *The origins of grammar*. Cambridge, MA: MIT Press.
- Legerstee, M. (2005). *Infants' sense of people*. New York: NY. Cambridge University Press.
- Matatyaho, D. J., & Gogate, L. J. (2008). Type of maternal motion during synchronous object naming predicts preverbal infants' learning of word-object relations. *Infancy*, *13*, 172-184.
- Meltzoff, A. N. (2007). "Like me.": A foundation for social cognition. *Developmental Science*, *10*, 126-134.
- Meyer, M., Hard, B., Brand, R. J., & Baldwin, D. (2008, March). *Naturalistic acoustic packaging: Temporal synchrony between maternal speech and action in mother-infant dyads*. Poster presented at the International Conference for Infant Studies, Vancouver, BC.
- Nagai, Y., Muhl, C., & Rohlfing, K. (2008). Toward designing a robot that learns actions from parental demonstrations. *Proceedings of the IEEE International Conference on Robotics and Animation*, 3545-3555.
- Nagai, Y. & Rohlfing, K. J. (2009). Computational analysis of motionese toward scaffolding robot action learning. *IEEE Transactions on Autonomous Mental Development*, *1*, 44-54.
- Oudeyer P-Y, Kaplan, F. and Hafner, V. (2007) Intrinsic motivation systems for autonomous mental development, *IEEE Transactions on Evolutionary Computation*, *11*, 265-286.
- Rohlfing, K. J., Fritsch, J., Wrede, B., & Jungmann, T. (2006). How can multimodal cues from child-directed interactions reduce learning complexity in robots? *Advanced Robotics*, *20*, 1183-1199.
- Rolf, M., Hanheide, M., & Rohlfing, K. J. (in press). Attention via synchrony: Making use of multimodal cues in social learning. *IEEE Transaction on Autonomous Mental Development*.
- Ruff, H. A., & Rothbart, M. K. (1996). *Attention in early development: Themes and variations*. New York: Oxford University Press.
- Schillingmann, L., Wrede, B., & Rohlfing, K. (2009). Towards a computational model of acoustic packaging. *IEEE 8th International Conference on Development and Learning*. 1-6.
- Simion, F., Regolin, L., & Bulf, H. (2008). A predisposition for biological motion in the newborn baby. *Proceedings of the National Academy of Science, USA*, *105*, 809-813.
- Thomaz, A.L., & Breazeal, C. (2008). Teachable robots: Understanding human teaching behavior to

build more effective robot learners. *Artificial Intelligence*, 172, 716-737.

Tomasello, M., & Carpenter, M. (2007). Shared intentionality. *Developmental Science*, 10, 121-125.

Wolf J.C., & Bugmann G. (2006). Linking speech and gesture in multimodal instruction systems, *Proceedings of IEEE International Symposium on Robot and Human Interactive Communication*, Hatfield, UK, 141-144.

Zacks, J. M., Tversky, B. & Iyer, G. (2001). Perceiving, remembering, and communicating structure in events. *Journal of Experimental Psychology: General*, 130, 29-58.

Epigenetic Embodiment

Luisa Damiano¹ and Paul Dumouchel²

¹ Adaptive Systems Research Group, School of Computer Science & STRI, University of Hertfordshire, College Lane, Hatfield, Herts AL10 9AB, UK, luisa.damiano@gmail.com °

² 先端総合学術研究科 - Graduate School of Core Ethics and Frontier Sciences, 56-1 Kitamachi, Toji-in, Kita-ku, Kyoto 603-8577 Japan, dumouchp@ce.ritsumeai.ac.jp

Abstract

In this short presentation we wish to formulate a definition of social embodiment grounded on an analysis of the relations between emotions, social interaction and development. Our hope is that this theoretical definition can be “tested” by social robotics, conceived in a “synthetic” (Pfeifer et al., 2008) or “constructivist” (Nehaniv and Dautenhahn, 2007) way – in short: building robotic models to comprehend human cognition.

One interest of this definition of social embodiment lies in that it expresses a “post-cognitivist” approach to cognitive science in its refusal of the “classic paradigm” that modelises the human cognitive system as a computer and cognition as computation of representational internal states (Nunez and Freeman, 1999; Calvo and Gomila, 2009). This different orientation is manifest in the very structure of our definition of social embodiment, which condenses four closely interrelated sub-theses. The first is that emotions should be conceived as salient moments in a process of social coordination (Dumouchel, 2008), rather than as internal states resulting from representational information-processing. In conformity with this first thesis, the second claims that “expression of emotions” should not be seen as a source of information concerning the internal states or intention of action of individuals, but as a social process through which the intentions of actions of agents are co-determined leading to coordination. The third thesis is that this process of emotional co-determination drives human epigenetic development – in particular cognitive development. The fourth is that this view of the role of emotional coordination in development requires a “radical embodiment” (Clark, 1999) conception of human “mind” (Damiano, 2009) that rests on a systemic definition of social embodiment, that can be expressed in the theoretical language of autopoietic biology and fits well with recent social neuroscientific insights (Gallese, 2005).

Social Emotions

What is a social species? There does not seem to be any universally agreed upon definition among biologists or ethologists. An animal for which interactions with other members of its own species are important is said to be a social animal. However this very intuitive definition is not extremely useful because it is not clear what “important” means. Furthermore there is a sense in which relations with other members of one’s species are always important at least in reproduction for organisms which reproduce sexually. We propose the following definition: a species is social to the extent that the biological advantages or disadvantages of its members depend on their interactions with other members of their own species.¹ Understood in this way, being social is a question of degree, and most animals are social, at least during some period in their life, i.e. early childhood, mating and when (and to the extent that) they raise their offspring. Strictly speaking, a social species is one where the majority of advantages that its members obtain and the majority of disadvantages they suffer result from interaction with other members of their own species. Being member of a social species is therefore to be *species dependent* in a particular way. All animals depend on other members of their species to reproduce and competition mostly takes place between members of the same species. However, the interests of individual members of a social species do not conflict and diverge only, they often also converge, and individuals can profit (and incur costs) as much from cooperation as from competition. In consequence these animals are faced with a particular problem of coordinating their behavior with other members of their species. The central claim of (Dumouchel, 2008; 1999) is that what we call emotions among humans, are aspects of an evolved mechanism that addresses this problem.²

¹ “Of their own species”: this allows for a principled distinction between being a social animal and being a parasite, as well as with symbiosis.

For the purpose of this paper it is not necessary to agree with this general thesis concerning the nature of emotions but simply to recognize that in a social context affective displays play a fundamental role in intra-specific coordination. In a social context emotions, before being motivations for action, are strategic signals (Ross and Dumouchel, 2004). Members of a social species can harm and benefit each other in very important ways; there results for each animal an uncertainty concerning the future behavior of others. Through affective expression humans try to reduce that uncertainty. Emotions are strategic signals through which we coordinate each others behavior. Affective expressions allow to steer shared relations towards cooperation or competition. Affective signals can be considered strategic because they help determine each participant's "payoff" in the "game of life". It does not however follow that they are the results of strategy on the part of individuals. However, affective displays should be considered as a form behavior in its own right, rather than as the expression of an internal state of the organism. Through such displays social animals influence each other. They properly act upon each other and transform each other's internal state. In consequence, the "emotion", considered as an internal state, comes after the expression rather than before.

Emotions should not be taken independently of the sequence of interactions that precede them and within which they intervene. The impression that emotions can be studied in isolation comes from the fact that we generally conceive emotions in the context of a single organism's relation to his environment. For example, the paradigmatic scenario for fear is usually something like a hiker who suddenly comes face to face with a snake on a mountain path. However in the context of social interactions emotions rarely happen all of a sudden, rather they take place in a history of interaction which gives them their meaning. Emotions are salient moments in a continuous process of social coordination. They are salient moments, in view of the fact that this coordination for the most part takes place at a sub-personal level. Agents are usually not aware of the way in which their affective expression both acts upon others and is

² Emotions are not the only way to resolve the problem of coordination among members of a social species. For example among social insects, as (Mead, 1934) had already noticed, the difficulty is to a large extent resolved through phenotypic differentiation of individuals.

a result of the others' affective expression. In social contexts strong emotions correspond to the moment when this ongoing process of coordination comes to the forefront of the relation.

These salient moments at times also correspond to fixed points in the process of affective coordination. Strong emotions do not only lead to strong motivations in those who experience them, they also create expectations in those who are exposed to them. These expectations should not be conceived in the form of explicit propositional contents. To say that your anger creates in me the expectation that you will become violent does not mean that I have any mental representation of your future behavior, but simply that I am afraid. My fear anticipates your violence and your anger anticipates my non-resistance. My fear satisfies your expectation and your anger sustains my expectation. In this way the system made of my fear and your anger finds an equilibrium point, which can become entrenched as a convention of coordination between us. Such equilibrium of affective coordination does not imply that the corresponding expectations are counter-factually true. That is to say, it does not imply that it is the case that if you do attack me I will not resist, nor does it imply that if I do resist you will attack me. However, once equilibrium is reached the expectations are in a sense "realized" and third parties who witness the interaction expect the agents involved to "live up", so to speak, to the expectations on which they settled.³

Emotions understood in this way are not internal states, nor are they, in themselves, strong motivations, though they can lead to them, but salient moments in a continuous process of social coordination. It is nonetheless true that changes of internal states and behavior are associated with these salient moments. It is therefore important to understand how these changes are brought about.

Against Communication

This process of social coordination through affective expression should not be conceived as a form of communication and it does not rest on the exchange of information concerning the internal states (emotions) of the organisms

³ As one of us tried to argue elsewhere, such expectations are "quasi- normative". See Dumouchel (2004).

involved. Even though adults, and robots, can infer conclusions, or “extract information” about the future actions of agents on the basis of their emotional displays, there are good empirical and theoretical reasons to believe that exchange of information plays a relatively minor role in affective coordination. Fernald (1993) and many others argue that during interaction between a caregiver and a very young infant the affective expression of the mother directly affects the nervous system of the child.⁴ Rather than carrying information about the internal state of the caregiver the affective expression of the adult directly causes the child’s affective state as a reaction.⁵ This, we claim, is not only the case with young children: even among adults the primary effect of affective displays is not to provide information, but to produce an affective reaction. We do not simply perceive the affective expressions of others, but also react affectively to them. This affective reaction is not only faster, but also much more flexible than the simple “recognition” of the expressed emotion. While the recognition of an emotion knows only one good answer and failure to correctly recognize the emotion constitutes a mistake, many different affective reactions can be appropriate when one is exposed to the same affective display. Fear, anger, surprise or even laughter can constitute appropriate responses to the other’s expression of anger.

In the case of the affective relation between a very small infant and its care-giver there are good empirical reasons to believe that affective coordination cannot rest on an exchange of information and that the process of direct influence is to some extent reciprocal.⁶

The main theoretical reason to believe that the process of social coordination does not rest on the exchange of information is because the relevant information does not actually exist before affective coordination is achieved. In its most general form social coordination can be described as that of coordinating one’s intention of action towards another agent with that agent’s intention of action towards us. As de Wall and Aureli (1999) pointed out, the main difficulty

⁴ Meany M.J. et al. (1996).

⁵ See also Desjardins and Fernald (2008) on the effect of social interactions on brain structure in various species.

⁶ Recent research on *mirror neurons* and other *mirroring mechanisms* suggest ways in which this reciprocal direct influence could be achieved among humans and other primates.

involved in doing this lays in the fact that an agent’s intention of action towards another is not independent of the other’s intention of action towards the first.⁷ Since this is true of both agents, it is clear that the goal of affective displays cannot be to exchange information which does not yet exist concerning one’s intention of action towards the other. Rather just as in the case of mother and young infant, adult agents mutually determine each other’s attitude towards each other through a continuous process of affective expression.

This process of mutual influence mainly takes place at a sub-conscious, sub-personal level. We are unaware of it most of the time and it usually only becomes conscious during those salient moments which we call “emotions”. Nonetheless it is clear that affective expression is continuous. Only androids and dead people can have a perfectly neutral expression,⁸ each and every one of us is always either happy, sad, busy, relaxed, “not to be bothered with”, engaging, serious, anxious, ridiculous, and so on. There is no affective silence. Even though all of the above terms refer to different states or attitudes which a person can have, inasmuch as they are expressed their effect is to act upon others. It is, for example, to “drive them up the wall”, to make them feel at ease, to chase or attract them. Unlike “emotions” affective coordination is not something which has a beginning and an end, it is an ongoing process. Through this process we mutually co-determine each other’s intention of action towards each other. This entails that human beings do not determine independently, autonomously, (at least some of) their most important intentions of action. What explains why this is so is that they are social animals who radically depend upon each other.

Development and social interaction

A central aspect of development in humans and many animals is the extent to which the organism changes. To mature, to become able to do and learn new things is inseparable from

⁷ It can be argued that in human societies this problem of uncertainty is to a large extent resolved through social codes of behavior. This is true, but affective coordination is a mechanism that functions in real time and that can always overrule the injunctions of social codes: we get angry or fall in love *when we should not!*

⁸ Hiroshi Ishiguro suggests that the reason of the “uncanny valley” may be that interacting with an android is a bit like interacting with a dead person (personal communication).

becoming physically different. Development entails ontogenesis and is equivalent to a transformation of embodiment. This change in embodiment among humans comes together with a profound revolution in the way a person participates in social relations and coordination. This corresponds to the fact that the individual becomes less dependent, and simultaneously a more dangerous competitor and a collaborator of greater value. In this section we wish to argue that social interactions drive development and that affective coordination is the main element in the transformation of a helpless organism into a relatively autonomous agent. The mechanism through which agent's intentions of actions towards each other are co-determined, and that bears witness to their radical interdependency, is also the means through which they become, to some extent, autonomous.

As mentioned earlier the relation between a very young infant and its caregiver is probably where social coordination through the co-determination of affective states can be most clearly seen. As development proceeds the influence of the affective displays of a person on others becomes more and more hidden from view. However it does not disappear. It continues together with other forms of coordination and is integrated in explicit communication between agents. For example, affective expression remains a fundamental aspect of spoken language that is easily recognizable in differences of pitch, speed and volume of speech. We say things to each other and the response we receive does not only depend on the semantic content of what is said but also on how it was said. The main difference between a very young infant and an older child or adult is the development of mind, which allows us, for example, to understand language. However a person's "mind" is also often moved, in ways it does not understand, by the affective expression of others.

A very young infant has a very limited repertoire of action. He or she can only do little, but from the early beginning of its life a child is able to either accept or refuse to interact. It is a fundamental social action. Small and helpless as the child is, this response has a strong influence on the caregiver's actions and attitude towards the child. As time goes by the child and caregiver's relation gets populated with objects of joint attention that can either be physical objects or situations. The infant learns to interact not only with another co-specific, but also to

interact together, but not necessarily in agreement, in relation to various objects. Hobson (2002) argues that a child develops a mind through discovering that the adult's attitude towards a given object can be different from the child's. In consequence the child can acquire the ability to simultaneously entertain multiple attitudes towards the same object. The underlying idea is that having a mind, in the sense of being able to think, requires, at least, to be able to adopt various stances towards the same object, rather than to be immediately determined by it. According to Hobson a child develops this ability through social interactions. In order for this to be possible, two conditions are necessary. First the child must have a specific attitude towards an object or situation. Two the infant needs to discover that it is possible to adopt a different attitude. Studies on imitation reveal children's (and adult's) tendency to reproduce the attitude of their partners in relation. That is to say the child will reach out for or reject what the adult rejects or reaches out for. These studies like many others reveal that a co-specific's behavior can have a direct effect on the attitude or behavior of the child. This however is a tendency rather than straight mimicry, so that the child (or adult) is often torn between two different attitudes. It is this phenomena resulting from social interaction which, according to Hobson (2002), allows the child to develop a variety of attitudes towards the same object.

Therefore the co-determination of the child's attitude by others gives him or her not only a series of different attitudes, but also the distance needed to allow for choice. Mind, understood in this way, is a fundamentally social attribute. It does not correspond to any specific characteristic of an isolated individual. Individual development is a result of social interaction.

Social Embodiment

The three hypotheses we proposed so far suggest a profound shift not only in relation to *classic*, but also to *embodied cognitive science*. Our proposal requires a conception of the embodiment that goes beyond "mainstream embodiment", what Clark (1999) calls "simple embodiment", which merely adds bodily and environmental constraints to a cognitive system and which in consequence remains essentially conforms to the classical computationalist paradigm. Our approach is closer to "radical

embodiment” (Clark, 1999; Thompson and Varela, 2001; Damiano, 2009), which, conceiving mind as emerging from the materiality of the body, abandons reference to the classical computer paradigm and proposes new concepts and parameters to define cognition. In this approach not only are traditional notions of internal representation and computation considered “inadequate and unnecessary”, but the “classical decomposition” of the cognitive domain in objects (primarily: brain, body, environment) is re-drawn. Radical embodiment tends to describe mind as a structures of coupling which interconnects brain(s), body(ies) and environment(s).

In line with this second view the description of socio-emotional interaction previously presented entails that the relevant unit for understanding development should not be conceived as an individual cognitive system that receives information from outside, and, after computation, produces a representation of the appropriate state of affairs. The cognitive unit relevant to our earlier description exceeds intra-individual space, and is not characterized by informational or representational processes. Our characterization of affective interaction entails that the agents involved are not independent. Unlike classical cognitive agents they are not entirely pre-defined systems, but evolve and change during the relation. It is through relation that agents acquire definite emotions and intentions of actions; through a dynamic process that recursively re-defines them in a coupled way. The process does not rest on the exchange of information, but on the direct action⁹ of each upon the other through ongoing mutual affective display. This dynamic could be described as *embodied co-determination*: an inter-subjective cognitive process in which perception of the affective display of the other modifies the state of the nervous system of the first and vice versa. In this type of cognitive interaction the entities that interact are not entirely pre-determined from the offset, rather as the relation develops coordinated emotions and intentions co-emerge. The reciprocity of influence between the two agents involved implies that neither entirely controls this mutual transformation. The real agent of socio-emotional coordination has to be found in the relations between self and other. In fact, it is better to conceive of affective exchange

⁹ The “direct action” of one system upon the other does not imply that it controls the other, but simply means to convey that this action takes place without the intermediary of either representation of information.

as a system in its own right. This is a hypothesis which we developed elsewhere. Dumouchel (1999; 2008) argues that the “body” of emotions is not individual but social and Damiano (2009) develops a model of self-organization that interprets inter-subjective cognitive interactions as the locus of emergence of transitory inter-individual unities. These theses express at the interindividual level the post-cognitivist thesis of the “extended mind” following which the proper unit for analyzing cognition is neither the brain nor the organism but the system made of the organism and environment (Clark and Chalmers, 1998; Chiel and Beer, 1997; Wilson, 2002; Pfeifer et al., 2007; Ziemke, 2003)

In consequence, our hypothesis is not entirely speculative. It is at least possible to suggest some biological mechanisms that can underlie this type of non representational, non informational coordination. For example, Dumouchel (2006) on the basis of a comparison between emotions and biological modules identified a class of mechanism that are implemented at the sub-personal level but whose consequences (effect) become visible at the level of entire populations. Rizzolatti et al. (2002) argued that mirror neurons transgress the distinction between what is intra-individual and what is inter-individual. According to Gallese, (2005) these mechanisms which “attune” the action intention, emotion, bodily sensations of interacting individuals at a sub-personal level and allow us to conceive interactions able to structure and define the agents they relate. As Damiano (2009) argues, neuro-physiological literature concerning *mirror neurons* and other *mirroring mechanisms* in monkeys and humans supports the idea of an inter-individual dynamics that coordinates brains, bodies and environment(s) of interacting individual and, in the process co-defines them.

It is on the basis of this view of mind and socio-emotional developmental interaction that we propose a definition of social embodiment – a “radical embodiment” definition of social embodiment. We will begin from the interesting definition of embodiment proposed by Dautenhahn, Ogden and Quick (2002) in the context of robotics. This definition tries to cash in on the idea that embodiment is structural coupling with environment (Ziemke, 2003). Dautenhahn et al. characterize embodiment as the minimal condition (necessary and sufficient?) of structural coupling as conceptualized in autopoietic cognitive biology

(Maturana and Varela, 1973) has been strongly criticized by Riegler (2002) and Ziemke (2003) who consider it “an insufficient characterization”, because “every system is in one sense or another structurally coupled to its environment”. Therefore “this definition of embodiment does not distinguish between cognitive and non cognitive systems” (Ziemke, 2003:1306). It seems to us that the real difficulty of this definition is that it fails to formulate correctly the idea of structural coupling as understood by Maturana and Varela.

According to our authors:

Df1: embodiment (Dautenhahn et al.)

*A system S is embodied in an environment E if perturbatory channels exist between the two. That is, S is embodied in E if for every time t at which both S and E exist, some subset of E's possible states with respect to S have the capacity to perturb S's state, all and some subset of S's possible states with respect to E have the capacity to perturb E's state.*¹⁰

However, according to Maturana and Varela, structural coupling is not limited to the fact that states of the environment can perturb states of the system and vice versa. Rather there is structural coupling when the following circumstances are satisfied.

*Two (or more) autopoietic unities can undergo coupled ontogenies when their interactions take on a recurrent or more stable nature.... The result will be a history of mutual congruent structural changes as long as the autopoietic unity and its containing environment do not disintegrate: there will be structural coupling.*¹¹

Structural coupling refers to dynamic interaction between two systems **S1** and **S2** that gives rise to a shared history of transformation, where the actions (reactions) of one trigger, but do not control, the reactions (actions) of the other. Each system's reactions are its endogenous compensations to the perturbations that the other system's reactions constitute. As clearly outlined by many authors (Varela, 1979; Thompson, 2007, Damiano and Luisi, 2009) structural coupling corresponds to a unit of co-transformation in which the organism and its environment (two systems) become co-dependent. The self-regulatory compensations of

each constitute for the other perturbations that, in turn, trigger transformations of the internal dynamics of the first. In order for structural coupling to exist, it is not enough for one system to receive perturbation from another and vice versa, it must also be the case that these perturbations give rise in one to self-regulatory compensations that become sources of self-regulatory compensation for the other. Furthermore this interaction must become recurrent, leading to a shared history or coupled ontogenies. Therefore we propose the following definition.

Df2: embodiment of a system in an environment

A system S is embodied in an environment E if (i) perturbatory channels exist between S and E through which each can trigger and modulate the self-regulation (autonomous activity of self-definition) of the other and if (ii) each has sufficient structural plasticity to respond to others's perturbation through its self-regulation activity

Df1 and **Df2** are actually quite different, as can be seen in relation, for example, to... a robot. According to **Df2** a robot is embodied in an environment, not only if its sensors allow it to receive perturbations from that environment, it must also have a repertoire of sensory-motor reactions to these perturbations *that reflects its own internal self-regulatory dynamics.*

Following Dautenhahn et al.'s lead we can transform this definition of embodiment in a definition of social embodiment.

Df3: social embodiment

Two systems S1 and S2 are socially embodied if (i) perturbatory channels exist between them and (ii) each of them has sufficient structural plasticity to generate a relation of co-dependence and co-specification of their self-regulating dynamics that (a) defines S1 and S2 as agents structurally coupled in their respective coupling with an environment and (b) defines an inter-individual unit (S1 and S2) that is coupled or embodied in the environment (Cf Df 2).

As this definition shows, there is no such thing as an individual isolated system that is socially embedded in a “social environment”. Social embodiment does not simply entail, but is structural coupling between two or more systems. This definition conveys that to be a social creature is to be dependent on other

¹⁰ Dautenhahn, Ogden and Quick, 2002, p. 400.

¹¹ Maturana and Varela (1987), p. 75.

similar system(s) in a very particular way. It can be understood as repeating in a different way what was said in the first section concerning the nature of social species and animals.

Social Robotics

If this analysis of emotions and social embodiment is correct, what does it mean to build a social robot (Cañamero, 2008)? What kind of creature, machine, agent, would such an artifact be? This paper argued that individuals are social to the extent that they are radically co-dependant. However this co-dependency, unlike symbiosis for example, does not exclude the autonomy or negate the individuality of the co-dependant agents. To the contrary, affective exchange was described as a mechanism of coordination among co-dependant systems that can lead to the emergence of autonomous agents. Furthermore we argued that embodiment is essentially social, to have a body is to be a partner in a relation.

To build a social robot then is to build a robot that is radically dependant on others (humans or other robots); a robot that also is, or can become, a pole of initiative in its relations with others. Such a robot would need to be an agent that can suffer damages if it fails to coordinate its action with those of others and who can gain advantages if it succeeds. Can this be done? The difficulty is to define these advantages and disadvantages and to determine a currency in which they can be measured or at least cashed out. One problem is that among us the accounting of gains and damages is finally made by natural selection. However, reproduction and survival do not constitute the *measure* with which individuals judge their success or failure. In fact, our co-dependency is directly located at the level of our motivational system. This means that the measure of individuals' gains and disadvantage cannot be given beforehand. Ultimately, it must emerge in and from the interaction of the robot with others, just as it does among us. Thus the goal cannot be to build a robot that "values" coordination with other, at least, not if it is going to be social in the way that we are. We humans do not particularly value coordination with others. It seems that some people never value it and it is certain that all people do not value it at least some of the time. Rather the point is that what we value is determined through our interaction with others and this is precisely what coordination means.

However, what we value however is not simply determined by others, but emerges through interaction with them. In consequence not only should a social robot's goals and objective be determined through its relations with other social agents, but it should also be case that the robot co-determines the goal and intentions of others.

References

- Breazeal C. L. (2002). *Designing Sociable Robots*. MIT, Cambridge MA-London.
- Calvo P. and Gomila T. (2008). Directions for an Embodied Cognitive Science. In Calvo P. and Gomila T. (Eds.). *Handbook of Cognitive Science*. Elsevier, Amsterdam, 1-25.
- Cañamero L. (2008). Animating affective robots for social interaction. In Cañamero L. and Aylett R. (Eds.). *Animating Expressive Characters for Social Interaction*. John Benjamins, Amsterdam-Philadelphia, 103-121.
- Clark A. (1999). An embodied cognitive science? *Trends in Cognitive Science*, 3, 9, 345-351.
- Clark A. and Chalmers D. J. (1998). The Extended Mind. *Analysis* 58, 7-19.
- Chiel H. J. and Beer R. D. (1997). The brain has a body: *Trends in Neurosciences*, 20, 553-557.
- Damiano L. (2009). *Unità in dialogo*. Bruno Mondadori, Milano.
- Damiano L. and Luisi P.L. (2009). Towards an Autopoietic Re-Definition of Life. *Origins of Life and Evolution of Biospheres*. Forthcoming.
- Dautenhahn K., Ogden B. And Quick T. (2002). From embodied to socially embedded agents. Implications for interaction-aware robots. *Cognitive Systems Research*, 3, 397-428.
- Desjardins J.K. and R.D. Fernald (2008). How do social dominance and social information influence reproduction and the brain? *Integrative & Comparative Biology* 48, 596-603.
- De Wall F. B. M. and Aureli F. (1999). Conflict Resolution and Distress Alleviation in Monkeys and Apes. In Carter C. S., Lederhendler I. I. and Kirkpatrick B. (Eds.). *The Integrative*

- Neurobiology of Affiliation*. MIT, Cambridge MA-London, 119-130.
- Dumouchel P. (1999). *Emotions. Essai sur le corps et le social*. Les Empêcheurs de Penser en Rond, Paris.
- Dumouchel P. (2004). Y a-t-il des sentiments moraux ? *Dialogue* 43, 471-489.
- Dumouchel P. (2006). Biological Modules and Emotions. *Canadian Journal of Philosophy supplementary volume* 32, 115-134.
- Dumouchel P. (2008). Social Emotions. In Cañamero L. and Aylett R. (Eds.). *Animating Expressive Characters for Social Interaction*. John Benjamins, Amsterdam-Philadelphia, 1-20.
- Fernald (1993). Approval and Disapproval. *Developmental Psychology*, 64, 657-674.
- Gallese V. (2005). Being Like Me. In Hurley S. and Chater N. (Eds.). *Perspectives on Imitation*. MIT, Cambridge MA-London, 101-118.
- Grosenick L., Clement T.S. and Fernald D.R. (2007). Fish can infer social rank by observation alone. *Nature* 445, 429-432.
- Hobson P. (2002). *The Cradle of Thought*. Macmillan, London.
- Maturana H. and Varela F. (1973). *De Máquinas y Seres Vivos*. Editorial Universitaria, Santiago.
- Mead G. H. (1934). *Mind, Self and Society from the Standpoint of a Social Behaviorist*. Chicago University Press.
- Meany M.J. et al. (1996). Early environmental regulation of forebrain glucocorticoid receptor gene expression: Implications for adrenocortical responses to stress. *Developmental Neuroscience*, 18, 49-72.
- Nehaniv C. and Dautenhahn K. (2007). The constructive interdisciplinary viewpoint for understanding mechanisms and model of imitation and social learning. In Nehaniv C. and Dautenhahn K. (Eds.). *Imitation and Social learning in Robots, Humans and Animals*. Cambridge University Press.
- Nunez R. and Freeman W. J. (Eds.) (1999). *Reclaiming cognition*, Imprint Academic, Thorverton.
- Peifer R., Lungarella M. and Iida F. (2007). Self-Organization, Embodiment, and Biologically Inspired Robotics. *Science* 318, 1088- 1093.
- Peifer R., Lungarella M. and Sporns (2008). The Synthetic Approach to Embodied Cognition. In In Calvo P. and Gomila T. (Eds.). *Handbook of Cognitive Science*. Elsevier, Amsterdam, 121-135.
- Riegler A. (2002). When is a Cognitive System Embodied? *Cognitive System Res.* 3(3), 339-348.
- Rizzolatti G., Fogassi L., Gallese V. (2002). Neurophysiological mechanisms underlying the understanding and imitation of action. *Nature Reviews Neuroscience*, 2, 661-670.
- Ross D. and Dumouchel P. (2004). Emotions as Strategic Signals. *Rationality and Society*, 3, 251-286.
- Thompson E. (2007). *Mind in Life*, Belknap (Harvard University Press), Cambridge MA.
- Thompson E. and Varela F.(2001). Radical embodiment. *Trends in Cognitive Science*, 5, 10, 418-425.
- Varela F. (1979). *Principles of Biological Autonomy*. North-Holland, New York.
- Wilson M. (2002). Six views of embodied cognition. *Psychological Bulletin and Review* 9 (4), 625-636.
- Ziemke T. (2003). What's that Thing Called Embodiment? In Akterman and Kirsh (Eds.). *Proceedings of the 25th Annual Meeting of the Cognitive Science Society*, 1134-1310.

°This article was written while L. Damiano was working at the Graduate School of Core Ethics and Frontier Sciences, Ritsumeikan University, Kyoto

Two Examples of Active Categorisation Processes Distributed Over Time

Tomassino Ferrauto Elio Tuci Marco Mirolli Gianluca Massera
Stefano Nolfi

ISTC-CNR, Via San Martino della Battaglia, 44, 00185 Rome, Italy
{tomassino.ferrauto, elio.tuci, marco.mirolli, gianluca.massera, stefano.nolfi}@istc.cnr.it

Abstract

Active perception refers to a theoretical approach grounded on the idea that perception is an active process in which the actions performed by the agent play a constitutive role. In this paper we present two different scenarios in which we test active perception principles using an evolutionary robotics approach. In the first experiment, a robotic arm equipped with coarse-grained tactile sensors is required to perceptually categorize spherical and ellipsoid objects. In the second experiment, an active vision system has to distinguish between five different kinds of images of different sizes. In both situations the best individuals develop a close to optimal ability to discriminate different objects/images as well as an excellent ability to generalize their skills in new circumstances. Analyses of evolved behaviours show that agents are able to solve their tasks by actively selecting relevant information and by integrating these information over time.

1 Introduction

Traditionally, Cognitive Science and Artificial Intelligence tended to view intelligence as the result of a chain of three information processing systems, constituted by perception, cognition, and action. According to this view, the perception system operates by transforming the information gathered from the external world (sensations) into internal representations of the environment itself. The cognitive system operates by transforming these internal representations into plans (i.e. strategies for achieving certain goals in certain contexts). Finally, the action system transforms plans into sequences of motor acts. This is what Susan Hurley has labelled the “Cognitive Sandwich” view of intelligence (Hurley, 1998), according to which perception and action are considered as peripheral processes separated from each other and from cognition, which represents the central core of intelligence.

The criticisms raised to this general view during the last two decades, however, led to the development of a new framework according to which perception, action, and cognition are deeply intermingled processes that cannot be studied in isolation (Clark, 1997; Pfeifer and Scheier, 1999). According to this view, behaviour and cognition should be conceptualised as dynamical processes that arise from the continuous interactions occurring between the agent and the environment (van Gelder, 1998; Beer, 2000).

This new view of cognition led also to a new approach to categorisation. Categorisation represents one of the most fundamental cognitive capacities displayed by natural organisms, being an important prerequisite for the exhibition of several other cognitive skills (Harnad, 1987): for example, it is involved in any task that calls for differential responding, from operant discrimination to pattern recognition to naming and describing objects and states-of-affairs. The “Cognitive Sandwich” view of intelligence tends to look at categorisation by focusing on processes that are passive (i.e., the agents can not influence their sensory states through their actions) and instantaneous (i.e., the agents are demanded to categorise their *current* sensory state). The new paradigm to the study of cognition mentioned above demands to look at categorisation processes that are “active” and possibly distributed over time.

Active perception can be studied by exploiting the properties of autonomous embodied and situated agents, in which perception is strongly influenced by the agent action (on this issue, see also Gibson, 1977; Noë, 2004). Nevertheless, our ability to build artificial systems that are able to exploit sensory-motor coordination is still very limited. This can be explained by considering that, from the point of view of the designer of the robot, identifying the way in which the robot should interact with the environment in order to sense sensory states that might facilitate perception is extremely difficult. One promising approach, in this respect, is constituted by adaptive methods in which the robots are left free to determine how they interact with environment (i.e. how they behave in order to solve

their task). There are several works that successfully employed such methods for the control of embodied agents in categorisation tasks. For example the works described in (Nolfi, 2002) and in (Beer, 2003) demonstrate how categorisation can emerge from the dynamical interaction between the agent and the environment. Other works have shown how an active perception system can act in order to perceive discriminating stimuli that greatly simplify the discrimination task (see, for example Scheier et al., 1998; Nolfi and Marocco, 2002). In some cases, however, sensory-motor coordination is not sufficient to experience well differentiated sensory patterns for different categories. Thus, in these circumstances the agents are required to integrate “ambiguous” sensory-motor states over time. So far, only a few studies have shown evolved agents that are able to cope with this kind of problems (e.g. Gigliotta and Nolfi, 2008; Tuci et al., 2004).

This paper presents two experiments that aim to extend the current state of the art to more complex scenarios. The rationale behind the decision to investigate more complex scenario is twofold. On one side we wanted to verify whether the adaptive techniques used in previous related works scale to more challenging problems. On the other side we wanted to ascertain whether more complex problems would lead to solutions that are qualitatively similar to those observed in previous research or not. The first experiment consists of a simulated anthropomorphic robotic arm with coarse grained tactile sensors that is asked to discriminate between spherical and ellipsoid objects. The high number of Degrees of Freedom (DoFs), the necessity to master the effects of gravity, inertia, and collisions, and the high similarity between the two objects make this problem rather challenging. The second experiment consists in an active vision system that has to correctly recognise five different letters of different sizes. In this case the difficulty lies in the number of categories (almost all previous works use only two classes) and in the variability *within* elements of the same category. Despite the two setups are quite different, we show that the principles that underlie the behaviour of successful agents in the two cases are the same. In particular, successful agents are able to obtain close to optimal performance by (a) *actively selecting* sensory stimuli so to reduce perceptual ambiguities as much as possible, and (b) *integrating perceived sensory-motor states over time*.

2 Experiment 1

2.1 Methods

The first experimental setup consists of a simulated anthropomorphic robotic arm and hand with tactile sensors which is asked to discriminate between spher-

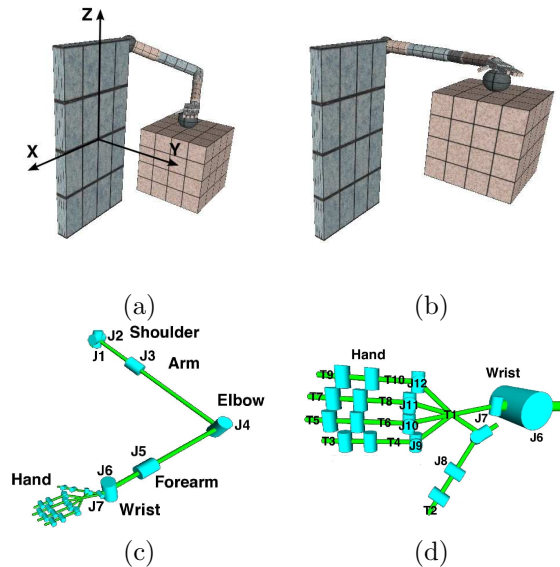


Figure 1: The simulated robotic arm (a) in position A, and (b) in position B. The kinematic chain (c) of the arm, and (d) of the hand. In (c) and (d), cylinders represent rotational DoFs; the axes of cylinders indicate the corresponding axis of rotation; the links among cylinders represents the rigid connections that make up the arm structure. T_i with $i = 1, \dots, 10$ are the tactile sensors.

ical and ellipsoid objects (see Fig. 1a and 1b). The experiment presented here is an extension of the work described in Tuci et al. (2009): please refer to that paper for additional information.

The robot and the robot/environment interactions are simulated using Newton Game Dynamics (NGD), a library for accurately simulating rigid body dynamics and collisions (www.newtondynamics.com). The arm has 7 actuated DoFs while the hand has 20 actuated DoFs. Fig. 1c shows the kinematic chain for the arm, the forearm and the wrist, with labels from J_1 to J_7 indicating rotational joints with the rotation axis along the axis of the corresponding cylinder. The robotic hand is composed of a palm and fourteen phalangeal segments that make up the digits (two for the thumb and three for each of the other four fingers) connected through 15 joints with 20 DoFs (see Fig. 1d). (See Massera et al., 2007, for a detailed description of the structural properties of the arm). Tactile sensors (indicated by the labels T_1 to T_{10} in Fig. 1d) return 1 if the corresponding part of the hand is in contact with any other body (e.g., the table, the sphere, the ellipsoid, or other parts of the arm), 0 otherwise.

The agent controller consists of a continuous time recurrent neural network (CTRNN, see Beer and Gallagher, 1992) with 22 sensory neurons, 8 internal neurons, 16 motor neurons, and 2 categorization neurons. The first 7 input neurons are updated on the basis of the state of the proprioceptive sensors on

joints J_1 to J_7 respectively (angles are linearly scaled on the range $[-1, 1]$), other 10 input neurons are updated accordingly to the state of tactile sensors T_1 to T_{10} respectively, and the remaining 5 input neurons are updated on the basis of the state of the hand proprioceptive sensors on joints J_8 to J_{12} respectively (angles are linearly scaled in the range $[0, 1]$, with 0 for a fully extended and 1 for a fully flexed finger). In order to take into account the fact that sensors are noisy, 5% uniform noise is added to proprioceptive sensors, while tactile sensors have a 5% probability of returning the wrong value. For all input neurons the activation value is computed by multiplying the corresponding sensory input by a gain factor g .

Internal neurons are fully connected to each other, and each receives one incoming synapse from each sensory neuron. Each motor and categorization neuron receives one incoming synapse from each internal neuron while there are no direct connections between sensory and motor neurons. The state of both internal, motor and categorization neurons is updated using the following equations:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

$$\tau_i \dot{y}_i = -y_i + \sum_{j \in N_i} \omega_{ji} \sigma(y_j + \beta_j) \quad (2)$$

where y_i is the state for neuron i , $\sigma(y_j + \beta_j)$ is the output of neuron j and N_i is the set of index of neurons with connection to neuron i . All time constants τ_i , biases β_i , network connection weights ω_{ij} , and all the input gains are genetically specified networks' parameters. There is one single bias for all the sensory neurons.

The activation values of motor neurons determine the state of the simulated muscles of the arm. Each joint in the arm is moved by an antagonist pair of muscles, so two neural outputs are associated with each joint (in total 14 neurons). For a complete description of the muscle model used in this work, see Massera et al. (2007). The joints of the hand are actuated by a limited number of independent variables through velocity-proportional controllers: the neural network has 2 output neurons for hand movements, one to set all desired thumb angles, the other to set the desired angles for all other fingers. The DoFs relative to joints J_9 to J_{12} are not actuated. Finally, the activation values of the two categorization neurons are used to categorize the shape of the object (see below).

A generational genetic algorithm is employed to set the parameters of the networks (see Goldberg, 1989; Nolfi and Floreano, 2000). The initial population contains 100 genotypes, represented as vectors of 420 parameters, each encoded with 16 bits. Generations following the first one are produced by a combination of selection with elitism and mutation:

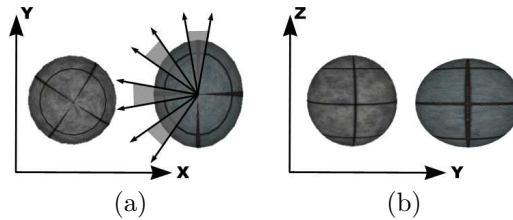


Figure 2: (a) The sphere and the ellipsoid of the first experiment viewed from above and (b) from west. The radius of the sphere is 2.5 cm. The radii of the ellipsoid are 2.5, 3.0 and 2.5 cm. In (a) the arrows indicate the intervals within which the initial rotation of the ellipsoid is set in different trials.

for each new generation, the 20 highest scoring individuals (“the elite”) from the previous generation are retained unchanged, while the remainder of the new population is generated by making 4 mutated copies of each of the 20 highest scoring individuals with 1.5% mutation probability per bit.

During evolution, each genotype is translated into an arm controller and evaluated 8 times in position A and 8 times in position B (see Fig. 1); for each position, the arm experiences 4 times the ellipsoid and 4 times the sphere. Moreover, the rotation of the ellipsoid with respect to the z-axis is randomly set in different ranges for each trial (see Fig. 2a). At the beginning of each trial, the arm is located in the corresponding initial position (i.e., A or B), and the state of the neural controller is reset. It is then left free to interact with the object (e.g. by sliding the hand above it so to make it slightly roll) for 4 simulated seconds (400 time steps) but the trial is terminated earlier if the object falls off the table.

In each trial, an agent is rewarded by an evaluation function that seeks to assess its ability to recognise and distinguish the ellipsoid from the sphere. Rather than imposing a representation scheme in which different categories are associated with *a priori* determined states of the categorization neurons, we leave the robot free to determine how to communicate the result of its decision, while requiring that objects' categories are well represented in the categorization-output space. More precisely, at each time step, the output of the two categorization neurons is a point in the bi-dimensional Cartesian space $C = [0, 1] \times [0, 1]$. Given a set of such points, one can build the AABB (Axis-Aligned Bounding Box), which is the minimum rectangle containing all points in the set such that its edges are parallel to the coordinate axes. The idea is that of scoring agents on the basis of the extent to which the AABBs associated to different categories are non-overlapping. During each trial, we collect the categorization output produced by the agent during the last 20 steps. We consider the sphere category (referred to as C^S) as the minimum bounding box

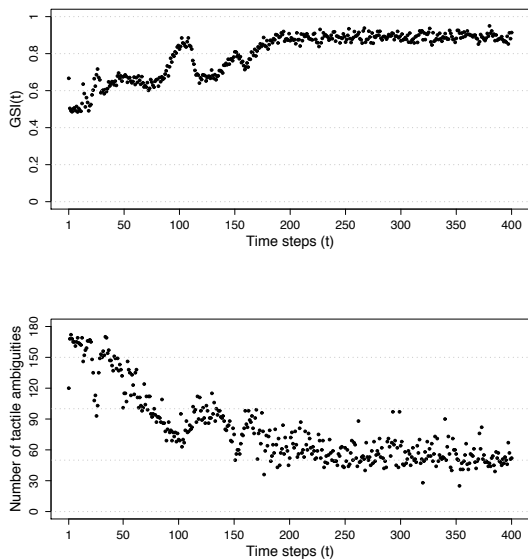


Figure 3: (a) The Geometric Separability Index (GSI). (b) Number of tactile ambiguities.

of all the categorization output collected while the agent was interacting with the sphere, and the ellipsoid category (referred to as C^E) as the minimum bounding box of all the categorization output collected while the agent was interacting with the ellipsoid.

The final fitness FF attributed to an agent is the sum of two fitness components: F_1 rewards the robots for touching the objects, and corresponds to the average distance over a set of 16 trials between the hand and the experienced object; F_2 rewards the robots for developing an unambiguous category representation scheme on the basis of the position in a two-dimensional space of C^S and C^E . F_1 and F_2 are computed as follows:

$$F_1 = \frac{1}{16} \sum_{k=1}^{16} \left(1 - \frac{d_k}{d_{max}} \right) \quad (3)$$

$$F_2 = \begin{cases} 0 & \text{if } F_1 \neq 1 \\ 1 - \frac{\text{area}(C^S \cap C^E)}{\min\{\text{area}(C^S), \text{area}(C^E)\}} & \text{if } F_1 = 1 \end{cases} \quad (4)$$

with d_k the euclidean distance between the object and the centre of the palm at the end of the trial k and d_{max} the maximum distance between the palm and the object when located on the table. $F_2 = 1$ if C^S and C^E do not overlap (i.e., if $C^S \cap C^E = \emptyset$).

2.2 Results

Eight evolutionary simulations, each using a different random initialisation, were run for 500 generations. Results of post-evaluation tests illustrated in (Tuci et al., 2009) shows that the best

evolved agent (hereafter, A_1) possesses a close to optimal ability to discriminate the shape of the objects as well as an excellent ability to generalize their skill in new circumstances. Moreover, in (Tuci et al., 2009) it is shown that A_1 , for one of the two positions experienced during evolution (i.e., position A, angle of joints J_1, \dots, J_7 are $\{-50^\circ, -20^\circ, -20^\circ, -100^\circ, -30^\circ, 0^\circ, -10^\circ\}$), exploits only tactile sensation to categorise the objects. In this Section, we take advantage of this latest result by running tests that further explore the dynamics of the decision of A_1 in position A, beyond the qualitative description illustrated in (Tuci et al., 2009). In particular, our interest is in finding out whether the discrimination process occur at a specific moment, as a response to a sensory pattern that encode the regularities which are necessary for discriminating, or if it occurs over time by integrating the information contained in several successive sensory states. Movies of the best evolved strategies can be found at http://laral.istc.cnr.it/esm/active_perception.

To answer this question we use a slightly modified version of the Geometric Separability Index (hereafter, referred to as GSI) originally proposed in (Thornton, 1997). GSI represents an estimate of the degree to which tactile sensor readings experienced during the interactions with the sphere or with the ellipsoid are separated in sensory space. We built four hundred data sets, one for each time step with the ellipsoid (i.e., $\{\tilde{I}_k^E(t)\}_{k=1}^{180}$), and four hundred data sets, one for each time step with the sphere (i.e., $\{\tilde{I}_k^S(t)\}_{k=1}^{180}$). Where, $\tilde{I}_k^E(t)$ is the tactile sensor readings experienced by A_1 while interacting with the ellipsoid at time step t of trial k ; and $\tilde{I}_k^S(t)$ is the tactile sensor readings experienced by A_1 while interacting with the sphere at time step t of trial k . Trial after trial, the initial rotation of the ellipsoid around the z-axis changes of 1° , from 0° in the first trial to 179° in the last trial. Each trial is differently seeded to guaranteed random variations in the noise added to sensors readings. At each time step t , the GSI is computed as follows:

$$GSI(t) = \frac{1}{180} \sum_{k=1}^{180} z_k(t)$$

$$z_k(t) = \begin{cases} 1 & \text{if } m_k^{EE}(t) < m_k^{ES}(t) \\ 0 & \text{if } m_k^{EE}(t) > m_k^{ES}(t) \\ \frac{u_k(t)}{u_k(t) + v_k(t)} & \text{otherwise} \end{cases}$$

$$m_k^{EE}(t) = \min_{\forall j \neq k} (H(\tilde{I}_k^E(t), \tilde{I}_j^E(t)))$$

$$m_k^{ES}(t) = \min_{\forall j} (H(\tilde{I}_k^E(t), \tilde{I}_j^S(t)))$$

$$u_k(t) = |\{\tilde{I}_j^E(t) : H(\tilde{I}_k^E(t), \tilde{I}_j^E(t)) = m_k^{EE}(t)\}_{\forall j \neq k}|$$

$$v_k(t) = |\{\tilde{I}_j^S(t) : H(\tilde{I}_k^E(t), \tilde{I}_j^S(t)) = m_k^{ES}(t)\}_{\forall j}| \quad (5)$$

where $H(x, y)$ is the Hamming distance between tactile sensor readings. $|x|$ means the cardinality of the set x . $GSI=1$ means that at time step t the closest neighbourhood of each $\tilde{I}_k^E(t)$ is one or more $\tilde{I}_k^E(t)$. $GSI=0$ means that at time step t the closest neighbourhood of each $\tilde{I}_k^E(t)$ is one or more $\tilde{I}_k^S(t)$.

As shown in Fig. 3a, the $GSI(t)$ tends to increase from about 0.5 at time step 1 to about 0.9 at time step 200, and to remain around 0.9 until time step 400. This trend suggests that during the first 200 time steps, the agent acts in a way to bring forth those tactile sensor readings which facilitate the object identification and classification task. In other words, the behaviour exhibited by the agent allows it to experience two classes of sensory states, rather well separated in the sensory space, which correspond to objects belonging to two different categories. However, the fact that the GSI does not reach the value of 1.0 indicates that the two groups of sensory patterns belonging to the two objects are not fully separated in the sensory space. In other words, some of the sensory patterns experienced during the interactions with an ellipsoid are very similar or identical to sensory patterns experienced during interactions with the sphere and vice versa. This is confirmed by the graph shown in Fig. 3b, which refers to the number of tactile ambiguities at each time step.

A tactile ambiguity is defined as a condition in which at least some of the patterns are experienced during interactions with both an ellipsoid and a sphere. If there are tactile ambiguities, then the agent cannot determine the category of the object solely on the basis of the single sensory stimuli. The fact that the number of tactile ambiguities never reaches zero while the agent gets an almost optimal performance implies that the agent’s categorization strategy involves an ability to integrate sequences of experienced sensory states over time.

3 Experiment 2

3.1 Methods

The second experimental scenario involves a simulated agent provided with a moving eye located in front of a screen that is used to display images to be categorized (one at a time). The eye includes a fovea constituted by 5×5 photoreceptors distributed uniformly over a square area located at the centre of the eye’s ‘retina’, and a periphery constituted by 5×5 photoreceptors distributed uniformly over a square area that covers the entire retina of the eye. Each photoreceptor detects the average grey level of an area corresponding to 1×1 pixel or to 10×10 pixels of the image displayed on the screen, for foveal and peripheral photoreceptors, respectively (see Fig. 4b). The activation of each photoreceptor ranges between

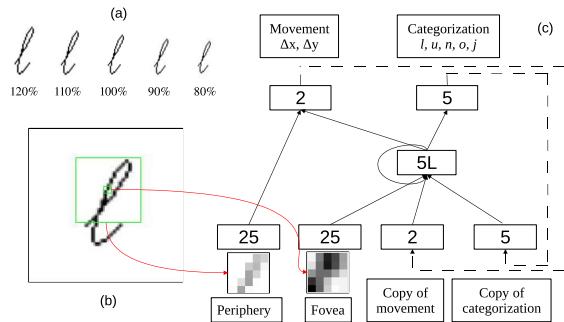


Figure 4: (a) Letter ‘l’ shown in the 5 different sizes used in the experiment. (b) The screen displaying the letter ‘l’ in its intermediate size and an exemplification of the field of view of the foveal and peripheral vision (smaller and larger squares, respectively). (c) The architecture of the neural controller. The number inside the each rectangle indicates the number of neurons, the letter L in a box indicates that these neurons are leaky integrators. Solid arrows between two boxes indicate all-to-all connections between neurons of those boxes, while dashed arrows indicate that the activation of the output units at time t is copied in the respective input units at time $t + 1$.

0 and 1 and is given by the average gray level of the pixels spanned by its receptive field (where 0 and 1 represent a fully white and a fully black visual field, respectively). The eye can explore the image by moving along the up-down and left-right axes up to a maximum distance corresponding to 25 pixels of the image. The screen, located in front of the agent’s eye, is used to display five types of italic letters (‘l’, ‘u’, ‘n’, ‘o’, ‘j’), each of which can be of 5 different sizes (with a variation of $\pm 10\%$ and $\pm 20\%$ with respect to the intermediate size: see Fig. 4a, for the letter ‘l’). The letters are displayed in black/gray over a white background. As shown in Fig. 4b, the eye can perceive only a tiny part of a letter with its foveal vision and a much larger but still incomplete part of the letter with its peripheral vision. It is important to clarify that this set-up is not intended to model how humans actually recognize letters; rather, the characteristics of the set-up have been chosen so to allow us to study how an active vision system can categorize stimuli through the exploitation of its eye movements and, possibly, to the integration of the perceived information over time.

Agents are provided with a neural network controller with 57 sensory neurons, 5 internal neurons, and 7 output neurons: see Fig. 4c for the network architecture. Notice that sensory neurons relative to the eye periphery are connected only to the two movement output neurons. This connection pattern represents a very crude abstraction of the functional organization of the human visual system, in which eye movements seem to be driven primarily by the periphery while recognition seems to be based pri-

marily on the information provided by fovea (Findlay and Gilchrist, 2003; Wong, 2008). To take into account the fact that sensors are noisy, a random value with a uniform distribution in the range $[-0.05; 0.05]$ is added to the activation state of each photoreceptor of the fovea in each time step.

The output of each of the 5 leaky internal neurons depends on the input received from the sensory and internal neurons through the weighted connections and by its own activation at the previous time step, and is calculated as follow:

$$O_i^t = \tau_i O_i^{t-1} + (1 - \tau_i) \sigma \left(\sum_{j \in N_i} O_j^{t-1} w_{ji} + b_i \right) \quad (6)$$

where O_i^t is the output of unit i at time t , τ_i is the time constant of unit i , in $[0; 1]$, w_{ji} is the weight of the connection from unit j to unit i , and b_i is the unit's bias, and $\sigma(x)$ is calculated as in equation 1. The output of the output units is calculated as in equation 6 but the time constant is fixed to 0 (i.e. output neurons do not depend on their previous state). The output of the motor units is then linearly normalized in the range $[-25; 25]$ and used to vary the position of the eye along the x and y axes of the image, respectively.

Free network parameters are learned using a genetic algorithm similar to the one described for the previous experiment. Agents are evaluated for 50 trials lasting 100 time steps each. At the beginning of each trial the screen is set so to display one of the five different letters in one of the five different sizes (each letter of each size is presented twice to each individual), the state of the internal neurons of the agent's neural controller is initialized to 0, and the eye is initialized in a random position within the central third of the screen (so that the agent can always perceive some part of the letter, at least with its peripheral vision). During the 100 time steps of each trial the agent is left free to visually explore the screen. Trials, however, are terminated earlier if the agent does not perceive any part of the letter through its peripheral vision for three consecutive time steps. The task of the agent consists in labelling the category of the current letter correctly during the second half of the trial. More specifically, the agents are evaluated on the basis of the following fitness function FF which comprises two components: the first one measures the agents' ability to activate the categorization unit corresponding to the current category more than the other units; the second one measures the ability to maximize the activation of the right unit while minimizing those of the other units:

$$F_1(t, c) = 2^{-rank(t, c)} \quad (7)$$

$$F_2(t, c) = \frac{1}{2} O_r^{t, c} + \sum_{O \in O_w^{t, c}} \frac{1}{8} (1 - O) \quad (8)$$

$$FF = \frac{\sum_{t=1}^{50} \sum_{c=50}^{100} \left(\frac{1}{2} F_1(t, c) + \frac{1}{2} F_2(t, c) \right)}{50 \cdot 50} \quad (9)$$

where $F_1(t, c)$ and $F_2(t, c)$ are the values of the two fitness components at step c of trial t , $rank(t, c)$ is the ranking of the activation of the categorization unit corresponding to the correct letter (from 0, meaning the most activated, to 4, meaning the least activated), $O_r^{t, c}$ is the activation of the output corresponding to the right letter at step c of trial t and $O_w^{t, c}$ is the set of activations corresponding to the wrong letters at step c of trial t . Notice that, as in the previous setup, individuals are not rewarded for moving their eyes or for producing a certain type of exploration behaviour but only for the ability to categorize (in this case the type of letter).

3.2 Results

Twenty evolutionary simulations were run, each lasting 3000 generations. The best agents of all simulations obtained on the average a good performance, with the best agent of the best replication reaching close to optimal performance. In order to better quantify the ability of the adapted agents to categorize the letters, we measured the percentage of times in which, during the second half of each trial, the categorization unit corresponding to the current letter is the most activated. We evaluated the best individuals of each of the 20 replications of the experiment for 10000 trials during which they are exposed to all possible combinations of the 5 letters with 50 sizes (uniformly distributed over the range $[-20\%, +20\%]$ of the intermediate size), 40 times each for each combination. As a result, we obtained that the average performance over all replications is 76.92% and the performance of the best individual of the best replication is 94.32%. In the remaining part of this section, we will focus our analysis on the best evolved agent, that is the best individual of replication 12.

By analysing the behaviour displayed by the best individual we can see how, after an initial phase lasting typically from 5 to 30 time steps (in which the behaviour varies significantly for different initial positions of the eye and for different letter sizes), the behaviour of the agent converges either on a fixed point attractor (i.e. the eye stops moving after having reached a particular position of the letter) or on a limit cycle attractor (i.e. the eye keeps moving by periodically foveating sequentially 2-6 different specific areas of the image). Interestingly, the agent displays the same type of behaviour in interaction with letters belonging to the same category even if they are of different sizes, and different behaviours for letters of different categories.

As for the previous experimental setup, we wanted to quantitatively ascertain the capacity of evolved

individuals to actively select discriminating stimuli. Apart from the efferent copies that provide as input the categorization output produced by the agent in the previous time step, the categorization answer of our system depends on two sources of information: the visual information provided by photoreceptors of the fovea and the motor information provided by the efferent copies of the motor neurons controlling the eye movements. Starting from the GSI index introduced in the previous experiment, we adapted it to the new setup and then we observed the evolution of the values of this index for both kinds of input (visual and motor) during the interaction of the agent with the images.

More precisely, in this case, the index takes into account all the stimuli experienced in interaction with an object of a given category. Hence, we devised what we call the Modified Geometric Separability Index (*MGSI*), which is defined as the average, over all patterns, of the proportion of the patterns belonging to the same category that are in the $|C_x|$ nearest patterns (using the euclidean distance), with $|C_x|$ representing the total number of patterns in the same category as pattern x . More formally, the *MGSI* is calculated as follows:

$$MGSI(P) = \frac{\sum_{x \in P} \frac{\sum_{n \in N_x} \mathbb{1}_{C_x}(n)}{|C_x|}}{|P|} \quad (10)$$

where $|S|$ indicates the cardinality of the set S , P is the set comprising all the patterns, C_x is the set of all patterns belonging to the same category as pattern x (x doesn't belong to C_x), N_x is the set of the $|C_x|$ patterns nearest to pattern x and $\mathbb{1}_{C_x}(n)$ is the indicator function of set C_x : it returns 1 if n is in the set C_x , 0 otherwise.

We calculated the *MGSI* of both the visual and motor-copy patterns experienced by the best evolved agent during 250 test trials, ten replications (with different initial positions) for each of the 5 by 5 letter-dimension pairings. More specifically, the two *MGSIs* were calculated for each of the 100 cycles composing trials, so that we could observe their evolution during the agent's interactions with the images. The results are shown in Fig. 5. They show three things. First, the separability of the input patterns in both sensory channels (visual and motor) significantly increase throughout trials, in particular during the first 20 cycles, meaning that the agent's sensory-motor behaviour has evolved so to facilitate the categorization process. Second, the geometric separability of the inputs in the two channels reaches very similar values (with the motor-copy channel being slightly better). Third, the geometric separability of neither of the two channels reaches very high values, meaning that, as in the previous experiment,

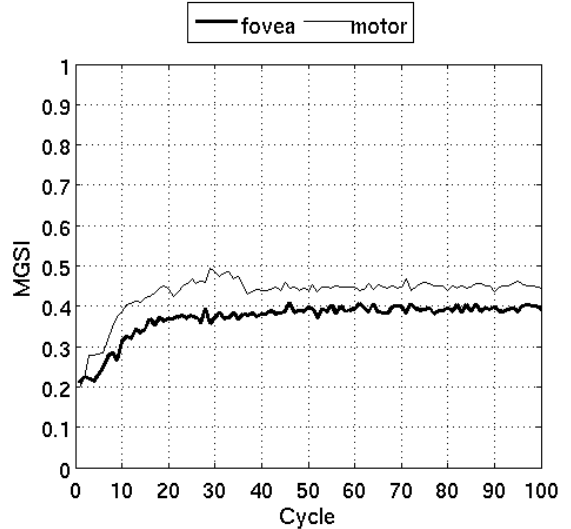


Figure 5: Evolution of the *MGSI* of the fovea and efferent copy of the eye movements inputs during the 100 cycles of the trials. Each point along the x axis represents the value of the *MGSI* calculated by taking all the inputs recorded in 250 trials (5 letters \times 5 dimensions \times 10 repetitions) during one of the 100 cycles of each trial.

to successfully solve the task the system has to integrate the information collected during different time steps, because each sensory pattern collected in a singular time step does not provide enough information for correct discrimination.

4 Conclusions

In this paper we presented two different experimental setups in which embodied agents are asked to categorize various objects by actively selecting their inputs. In the first scenario an anthropomorphic robotic arm equipped with coarse grained tactile sensors has been asked to distinguish between spherical and ellipsoidal objects. The setup is significantly more complex than those used in previous related works due to the high similarity between the objects to be discriminated, the difficulty of controlling a system with so many degrees of freedom, and the need to master the effects produced by gravity, inertia, collisions, etc. Nevertheless the evolved system is able to solve the task and reach close to optimal performance.

The second scenario involves an agent with a simulated moving eye that have to recognize different letters. Whereas work in related literature has mainly focused on experiments comprising only two categories, this setup is more challenging as there are significantly more categories with more variability (five letters of different dimensions). Also in this case the system is able to successfully solve the task with a close to optimal performance.

Both experiments show that active perception systems are indeed able to cope with complex scenarios. The ability to actively select one's own input is exploited by agents by selecting stimuli that provide regularities that can be used to categorize (i.e. stimuli that are often, although not necessarily always, experienced in interaction with objects of the corresponding category). Despite the effectiveness of their actions, however, agents often encounter input patterns associated with more than one category. Thus, evolved agents also show a complementary ability to integrate over time the partially conflicting information provided by the experienced stimuli.

Acknowledgements

This research work was supported by the *ITALK* project (EU, ICT, Cognitive Systems and Robotics Integrating Project, grant n° 214668).

References

- Beer, R. (2000). Dynamical approaches to cognitive science. *Trends in Cognitive Sciences*, 4:91–99.
- Beer, R. and Gallagher, J. (1992). Evolving dynamic neural networks for adaptive behavior. *Adaptive Behavior*, 1(1):91–122.
- Beer, R. D. (2003). The dynamics of active categorical perception in an evolved model agent. *Adaptive Behavior*, 11(4):209–243.
- Clark, A. (1997). *Being There: putting brain, body and world together again*. Oxford University Press, Oxford.
- Findlay, J. M. and Gilchrist, I. D. (2003). *Active Vision. The Psychology of Looking and Seeing*. Oxford University Press, Oxford.
- Gibson, J. J. (1977). The theory of affordances. In Shaw, R. and Bransford, J., (Eds.), *Perceiving, Acting and Knowing. Toward an Ecological Psychology*, chapter 3, pages 67–82. Lawrence Erlbaum Associates, Hillsdale, NJ.
- Gigliotta, O. and Nolfi, S. (2008). On the coupling between agent internal and agent/environmental dynamics: Development of spatial representations in evolving autonomous robots. *Adaptive Behavior*, 16:148–165.
- Goldberg, D. (1989). *Genetic algorithms in search, optimization and machine learning*. Reading, MA: Addison-Wesley.
- Harnad, S., (Ed.) (1987). *Categorical Perception: The Groundwork of Cognition*. Cambridge University Press.
- Hurley, S. (1998). *Consciousness in Action*. Harvard University Press, Cambridge, MA.
- Massera, G., Cangelosi, A., and Nolfi, S. (2007). Evolution of prehension ability in an anthropomorphic neurorobotic arm. *Front. Neurobot.*, 1.
- Noë, A. (2004). *Action in Perception*. MIT Press, Cambridge, MA.
- Nolfi, S. (2002). Power and limits of reactive agents. *Neurocomputing*, 49:119–145.
- Nolfi, S. and Floreano, D. (2000). *Evolutionary robotics. The biology, intelligence, and technology of self-organizing machines*. MIT Press, Cambridge, MA.
- Nolfi, S. and Marocco, D. (2002). Active perception: A sensorimotor account of object categorisation. In Hallam, B., Floreano, D., Hallam, J., Hayes, G., and Meyer, J.-A., (Eds.), *Proc. of the 7th International Conference on Simulation of Adaptive Behavior (SAB '02)*, pages 266–271. MIT Press, Cambridge, MA.
- Pfeifer, R. and Scheier, C. (1999). *Understanding intelligence*. MIT Press, Cambridge, MA.
- Scheier, C., Pfeifer, R., and Kuniyoshi, Y. (1998). Embedded neural networks: exploiting constraints. *Neural Networks*, 11(7-8):1551–1596.
- Thornton, C. (1997). Separability is a learner's best friend. In Bullinaria, J., Glasspool, D., and Houghton, G., (Eds.), *Proc. of the 4th Neural Computation and Psychology Workshop: Connectionist Representations*, pages 40–47. Springer Verlag, London, UK.
- Tuci, E., Massera, G., and Nolfi, S. (2009). Active categorical perception in an evolved anthropomorphic robotic arm. In *Proc. of the IEEE Conference on Evolutionary Computation (CEC '09), Special Session on Evolutionary Robotics*, ISBN: 978-1-4244-2959-2. Draft available at <http://laral.istc.cnr.it/elio.tuci/pagn/pubbb.html>.
- Tuci, E., Trianni, V., and Dorigo, M. (2004). Feeling the flow of time through sensory-motor coordination. *Connection Science*, 16(4):301–324.
- van Gelder, T. J. (1998). The dynamical hypothesis in cognitive science. *Behavioral and Brain Sciences*, 21:615–665.
- Wong, A. M. (2008). *Eye Movement Disorders*. Oxford University Press, Oxford.

Applying the Schema Mechanism in Continuous Domains

Frank Guerin

Department of Computing Science
University of Aberdeen
Aberdeen, AB24 3UE, Scotland
f.guerin@abdn.ac.uk

Andrew Starkey

School of Engineering
University of Aberdeen
Aberdeen, AB24 3UE, Scotland
a.starkey@abdn.ac.uk

Abstract

We are interested in developing a computational model of Piaget's theory of sensorimotor intelligence. Existing works in this area have demonstrated mechanisms which acquire Piagetian schemas, however the sensory inputs to these systems are typically constrained to a small number of discrete values. In order to model Piagetian developments such as the acquisition of skills it will be necessary to handle continuous (or real) domains of sensor values, and to learn skills which are guided by feedback from these real valued sensors. We extend existing Piagetian work by employing a neural network function approximator, to represent a reinforcement learning value function over a real valued sensor space. Using this combination of techniques allows our system to learn skilled actions which can then be treated as Piagetian schemas, and combined with other schemas. Our experiments in a simple simulated world show that this novel combination is feasible in principle; future work will need to test the approach in more challenging domains to determine its limitations, and to improve on it.

1. Introduction

We are interested in building AI systems which can learn their own world knowledge autonomously, and exhibit ongoing development (sometimes called "ongoing emergence" (Prince et al., 2005)). This idea comes from trying to copy the biological approach to the development of world knowledge, in particular human cognitive development during infancy. Piaget's theory of constructivism gives an account of how humans build up their world knowledge through their interactions with the environment (Piaget, 1936). Piaget's theory is a grand overview of the human learning mechanism, but unfortunately it does not give the level of detail which would be necessary to inform a computer implementation. To make progress in computational models of this theory

a number of AI works have carried out computational investigations on small parts of the theory (see Section 2). These works have demonstrated the possibility of a learning mechanism which acquires Piagetian schemas through trial and error, and can build on this knowledge through techniques such as chaining schemas.

Building on existing Piagetian AI work, we have set ourselves a long-term target to build a computational model of Piagetian means-end behaviours; this is the fourth of Piaget's six sensorimotor stages, and commences at about eight months of age. In order to model this fourth stage our AI system needs to be able to acquire *means* actions. These are skilled motor actions, such as grabbing a seen object, or hitting an object to make it swing, or scratching an object, etc. This gives us a short term target to be able to build a system which can acquire these skilled actions and use them within a Piagetian learning framework. We find that the existing Piagetian AI work is inadequate for skill acquisition, as existing work tends to use sensory inputs which are constrained to a small number of discrete values. For the acquisition of skills it will be necessary to handle continuous (or real) valued sensors, and to learn skills which are guided by feedback from these real valued sensors. To address this we use the technique of Neural Fitted Q Iteration (Riedmiller, 2005). This is a reinforcement learning method, which employs a neural network function approximator to represent a value function over a real valued sensor space. This technique makes use of a set of *transition experiences* (in our case these are small arm movements taken at various different positions), and generalises to find a value function over the space. A particular strength of the approach is that the same transition experiences can potentially be used in training for different goals. In order to identify the sensor variables which are relevant to the acquisition of a particular skill, we used a simple statistical analysis; this then allows the neural network to ignore irrelevant sensors, thus there are fewer inputs to the neural network.

Using this combination of techniques allows our system to learn skilled actions which can then be treated as

Piagetian schemas, and combined with other schemas. The overarching Piagetian framework we use is the Constructivist Learning Architecture (CLA) (Chaput, 2004). This architecture finds reliable schemas, and also allows higher layers of learning where composite actions can be treated as atomic in order to find further schemas.

Our experiments in a simple two-dimensional simulated world show that this novel combination is feasible in principle. Our system learnt skills to bring the hand to the mouth and to bring the hand to a seen object. Composite actions chained these to bring a seen object to the mouth.

In Section 2 we cover the background work which we are building on. Section 3 describes our simulated world. Section 4 details the learning techniques and how we have applied them. Section 5 gives results of our experimentation. Section 6 concludes with a discussion and some future directions.

2. Background

This section looks at some of the existing work in the computational modelling of Piagetian Schemas in the sensorimotor stage, and then gives the necessary background on the techniques which we will make use of.

2.1 *Piagetian Schemas in AI*

There exist a number of AI works which are inspired by Piagetian schemas. Drescher’s “schema mechanism” was the first of these (Drescher, 1991). Drescher simulated a baby, with a hand, eye, and mouth, in a 7x7 grid world. The world also contained some objects which the baby could grab. Drescher’s schemas were 3-part structures consisting of a context, action, and result. A schema is a prediction about the world: if its action is taken in the context specified, then the result is predicted. For example one schema which the program learnt is that if its current context was “HandInFrontOfMouth”, and it took the action “HandBackwards”, then it would expect to obtain the result “HandTouchingMouth”. The learning mechanism was also capable of chaining together a number of schemas as a *composite action* in order to achieve a goal. Drescher’s mechanism was criticised for its efficiency, and improved on by Chaput (2004), as described in the next subsection. Nevertheless, Drescher’s work has been influential, with many subsequent works following a similar pattern. The learning mechanism incorporates many of the elements we would want in a Piagetian framework: schemas are acquired when a context/action/result triple occurs reliably, schemas are then stored in a library, and can be activated, and furthermore schemas can be chained up to make composite actions to achieve goals. The shortcoming for our purposes is the lack of a method to generalise over real valued inputs, i.e. if the context consisted of real sensor

values.

In the Petitagé architecture of Stojanov (2001) the agent learns “expectancies” of the form $\langle \text{Sensor_state}, \text{action}, \text{Sensor_state} \rangle$, which are similar to Drescher’s context/action/result triple. This was applied to learning the structure of a maze with walls, and the agent built a partial map of its world. The architecture is not specific about the types of sensors required and allows for the possibility of real-valued sensors, however the issue of generalising over real values of sensors has not been tackled.

A further work by Perotto and Álvares (2006) has a tripartite schema structure consisting of: context, action and expectation (just like Drescher’s). The learner has the ability to generalise contexts that achieve the same result. Contexts are represented by binary strings, and the generalisation converts a ‘1’ and ‘0’ in the same position in two contexts to produce a ‘#’ value, which is a wildcard and could match ‘1’ or ‘0’. Although the work deals with binary strings, these strings could be used to represent real sensors (i.e. binary representation of the real number). This type of generalisation could learn tiled regions where contexts are similar, but would not be able to find regions of other shapes. We require a superior generalising ability for our purposes.

A completely different approach to schema learning appears in Hart et al. (2008) and is worth mentioning here. In this work closed-loop feedback control programs are used for basic sensorimotor actions (such as reaching and touching), and then these appear as discrete actions within a reinforcement learning framework which can learn higher level behaviours. This work handles continuous domains very well and is a feasible approach to robotic manipulation problems where there are many degrees of freedom. In such domains it might not be feasible to have a pure reinforcement learning approach exploring the whole space, so an a priori model of controller performance may be essential.

2.2 *Constructivist Learning Architecture*

The first three works above reach some sort of consensus on the idea of schemas being context/action/result triples. The work of Chaput (2004) goes further by incorporating this idea with a neo-piagetian learning theory, to make a new architecture for developmental learning. Chaput developed a “Constructivist Learning Architecture” (CLA) which is based on Leslie Cohen’s theory of infant cognitive development Cohen (1998). This theory essentially states that infants learn to process information at increasingly higher levels of abstraction by forming higher level units out of relationships among lower level units. There is a bias to process information using the highest formed units, unless the input becomes too complex, in which case the infant drops back to a lower level and attempts to refine its abstraction so as to be

able to handle the complex information at the higher level. The CLA is very much a Piagetian learning architecture, in which schemas (similar to Drescher’s) are learnt at each level. The architecture is quite generic; the higher levels can be formed by integrating multi-modal information from lower layers, or integrating time delayed versions of sensory input. As one of his experiments with the architecture, Chaput recreated the achievements of Drescher, in a more efficient way.

Chaput’s computational model (CLA) is based on Self Organising Maps (SOMs) which are built hierarchically (modelling the different levels of Cohen’s theory). Chaput uses SOMs as way for different representations to compete for the finite available space. As in Drescher’s work, the learning agent records context/action/result triples. A SOM is created for each possible action of the agent. The SOM is trained on vectors which represent the context and result of taking that action a number of times. The SOM thus finds reliable patterns of context/action/result (i.e. these come out with a strong representation in the SOM). Furthermore, the CLA learns schemas at a certain level first until it has some stable knowledge, before moving on to consider learning on the next level. Once the CLA has moved to a higher level, those schemas from the lower level are not updated anymore; learning on that level has frozen. Thus the learning resources can focus on one level at a time.

Chaput’s work did not address generalisation over real sensor values however. His robot forager example had no need for this as it only had seven binary sensors. For our work we will make use of the CLA as it is quite generic, and generally suits our purpose, however we will combine it with a generalisation technique to be described next.

2.3 Riedmiller’s Neural Fitted Q Iteration

In order to deal with real valued sensors we borrow a technique which uses a neural network for function approximation in reinforcement learning. Riedmiller’s Neural Fitted Q Iteration (NFQ) (Riedmiller, 2005) is a method which trains on a set of transition experiences, each of which has the form $\langle s, a, s' \rangle$, state, action, resulting state; these triples are similar to Drescher’s context/action/result. In addition each triple has a reward value (in practice most of the triples will typically have zero reward, and just a small proportion have a large reward). The system learns a Q value which represents the discounted expected reward to be obtained by executing a specific action in a given context. Thus the Q value function takes as input a context and action, and outputs a real value (the Q value). The Q value function is represented by a neural network. The training iterates, performing two main steps in each iteration, the first step is to do a sweep through all the transition experiences $\langle s, a, s' \rangle$, finding a new ideal value for Q, for the given context s and action a . This uses the following

equation:

$$Q_{k+1}(s, a) := (1-\alpha)Q_k(s, a) + \alpha(R(s, a) + \gamma \max_b Q_k(s', b))$$

where $Q_{k+1}(s, a)$ is the new ideal value for Q, given context s and action a ; α is the learning rate; $Q_k(s, a)$ is the old value for Q (from the neural network, before the update); $R(s, a)$ is the reward which was obtained for executing action a in context s ; γ is the discount; $\max_b Q_k(s', b)$ is the Q value of the best action b from the resulting state s' . The second step is to use these new ideal values to update the Q value function. This update is done using the RPROP neural network training algorithm (Riedmiller and Braun, 1993). After a number of iterations we have a Q value function which can recommend to us the best action to take in any given context, in order to maximise reward. To do this we can simply query the Q value for every action from the given context; the highest value is the best action.

The generalisation performed by the neural network means that the coverage of the sensor space by the transition experiences can be relatively sparse; when the Q learning is updating a Q value by looking ahead to the expected reward to be obtained by taking a particular transition, this transition does not need to connect directly to another, as the neural network will have interpolated between the data points given. Similarly, the final Q function can be queried for a context which is not present in any transition experience. The key innovation of NFQ is the fact that it maintains a record of all transition experiences, whereas other approaches use a transition experience to train online and then discard it. This gives NFQ greater stability, as its early learning will not be damaged or undone by later learning. A further added advantage is that transition experiences are not tied to a particular reward; transition experiences merely record how an action moves the agent from one state to another. This means that the same experiences could be used in the training for multiple different goals.

3. The Simulation

Our simulated infant lives in a simple 2D world with a single rigid square block which can be grabbed and moved. A rigid body physics engine simulates the physics of the world, including friction and collisions between blocks; gravity has been disabled.

The infant (see Figure 1) has a single movable arm, consisting of two rigid rectangular blocks: an upper arm and a lower arm. The upper arm can rotate at the shoulder, and the lower arm rotates from the elbow. There is a hand at the end of the lower arm (represented by a square). When the hand overlaps with a block, a touching sensation is generated, and if the grab action is then taken, the block will then move together with the hand, until released.

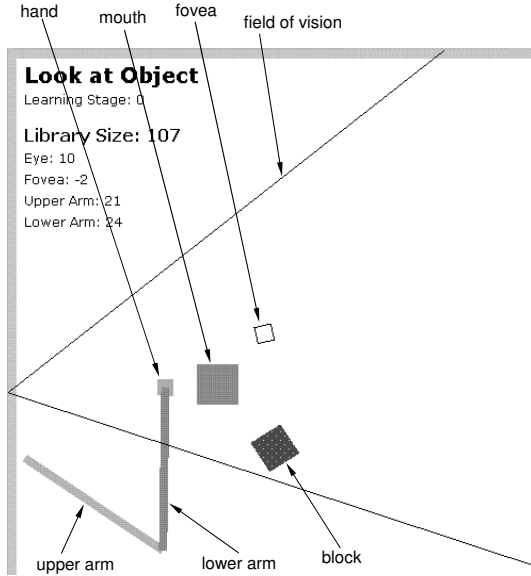


Figure 1: The simulation.

The infant’s mouth is the square at the centre of the figure. Note that the mouth position had originally been at the eye, but this made the learning of hand to mouth too easy, as it was nearly always optimal to bend the elbow more, from whatever position, in order to reach the mouth. The mouth position was moved to the middle to make the learning problem more interesting.

The infant also has a field of vision, bounded by the two lines emanating from the left wall. The point of intersection of the two lines is the “eye”. The field of vision can be rotated about the eye. As in Drescher’s simulation, the centre of visual attention has a fovea; in our system the fovea is capable of moving along a path between the two eye lines (and equidistant from them). The fovea is shown as an outlined box in the figure.

The infant has a number of sensors as shown in table 1. The last two sensors are part of the *seen_objects*

Type	Sensor
float	<i>lower_arm_angle</i>
float	<i>upper_arm_angle</i>
float	<i>eye_angle</i>
float	<i>eye_fovea_distance</i>
boolean	<i>fovea_sees</i>
boolean	<i>mouth_touched</i>
boolean	<i>mouth_touched_object</i>
boolean	<i>hand_touched</i>
boolean	<i>hand_holding_object</i>
float	<i>seen_objects_distance</i>
float	<i>seen_objects_angle</i>

Table 1: Infant’s sensors

sensor, which returns a set of objects that are in view (i.e., between the lines bounding the field of vision); each

object in the set is described by a pair: angular displacement from centre of visual field, distance from fovea. The “distance from fovea” is the object’s distance from the arc on which the fovea would move if rotated, or alternatively, the distance from the fovea if the object were on the centre line. In the experiments we ran for this paper there were only two objects to be seen: a block, or the infant’s own hand, and in fact only the hand was actually used as an input for learning. Note that there is no occlusion: objects which are behind others are still returned in the *seen_objects* list.

Table 2 shows the nine actions which the infant is capable of. As can be seen from the table, we distin-

Type	Action
continuous	UPARM_UP
continuous	UPARM_DOWN
continuous	LOWERARM_UP
continuous	LOWERARM_DOWN
continuous	EYE_UP
continuous	EYE_DOWN
discrete	LOOK_AT_OBJECT
discrete	HAND_GRAB
discrete	FIXATE

Table 2: Infant’s actions

guish continuous and discrete actions. Continuous here means that the action’s outcome is dependent on continuous (real valued) sensors, and it affects those sensors; i.e. the action is moving something in a continuous space. Discrete means that the action’s outcome depends only on Boolean sensors, and it affects only those; i.e. the action is setting some binary item in the world. There are four continuous arm actions: the upper arm and lower arm can both (independently) move up and down. There are two actions for the eye. The discrete action LOOK_AT_OBJECT is really a composite action which we have coded in innately rather than getting the system to learn it; this action moves the fovea to focus on the object, if the object is visible in the field of view (otherwise the action does nothing). The discrete action HAND_GRAB fixes the position of the object relative to the hand, if the hand is touching the object (otherwise the action does nothing). The action FIXATE fixes the position of the fovea, disabling eye movements for a time interval (this action is only effective if an object is in the fovea). Note that these last two discrete actions have been programmed in as reflexes so that they are automatically triggered when a sensor is activated: HAND_GRAB happens whenever *hand_touched* is on, and FIXATE happens whenever *fovea_sees* is on. The idea of FIXATE is that it forces the infant to spend a long time with the fovea focussed on the block, during which time other hand movement actions can be taken; this allows the program to learn how its arm movements

can affect the position of the hand, relative to the block in the centre of vision (and so potentially learn how to move the hand closer to the fovea and contact the block). There is no action for the hand to release what is grabbed; a grabbed object is simply released after a random time interval.

4. The Learning Mechanism

Our learning mechanism treats discrete actions and continuous actions differently. Discrete actions are used to learn context/action/result schemas, just like Chaput’s (see Section 2). These schemas reflect the most reliable context and result of the action when it was taken many times. On the other hand, when continuous actions are taken, we record transition experiences (following Riedmiller); although these are similar to the what many related works call schemas, we will reserve the term “schema” for higher level knowledge, generalised from many transition experiences. Thus from a set of transition experiences, we learn a Q value function, which means we have a policy for achieving a goal. We then treat this policy as a schema, and it becomes a new (composite) discrete action which can be taken, and context and result recorded for higher level learning.

Note that by making this distinction, and using different learning methods for discrete and continuous actions, we are effectively giving the infant some innate knowledge: i.e. the infant innately knows that the continuous actions’ effects depend on the real valued sensors (but it does not know which sensors exactly).

4.1 Learning For Discrete Actions

Following Chaput (2004) we record vectors of the values of all sensors before (context) and after (result) each action is taken. For the discrete actions we discard all the real-valued sensors. This gives us a vector of ten binary digits for each time an action has been taken (the five discrete sensors before and after, corresponding to context and result). We now change the five result values to represent the change that happened as a result of the action. This is computed by subtracting the context values from the result values. Therefore, if a discrete sensor changed from 1 to 0 then it will have a change of -1 whereas if it changed from 0 to 1 the change will be 1, otherwise it will be 0. An illustrative example follows:

Initial context/result vector: [1 0 0 1 0 / 0 1 0 1 0]

After change is computed: [1 0 0 1 0 / -1 1 0 0 0]

For each discrete action we create a 10x10 Self Organising Map (SOM) and train it with all the vectors for that action (we had approximately 1000 vectors recorded for each action). We then perform thresholding on the weights of the resulting SOMs. The threshold used was 0.9. This means that, for context values, any weight greater than 0.9 becomes 1, any weight less than 0.1

becomes -1, and any value in between becomes 0; for result values, any weight greater than 0.9 becomes 1, any weight less than -0.9 becomes -1, and any value in between becomes 0. After this any weight vector which has at least one positive result becomes a schema (unless it already exists as a schema). The SOM method is averaging, so vectors which consistently have ones (or zeros) in certain positions in the context and result will be harvested; wherever there is a mix of ones and zeros in a position, an intermediate value will result, which will not make it above the threshold. It is important that this applies to both context and result, because a result which occurs regularly would be useless without a context which says when it reliably occurs. This process is known as *harvesting* schemas, and follows Chaput (2004) exactly. A schema states that if its context is matched, and the action is taken, then the result can be expected to be achieved. Note that values of 0 in the context mean “don’t care” whereas 1 means “must be 1” and -1 means “must be 0”.

In our experiments this procedure proved to be good at identifying results that could be reliably achieved, but it found contexts which were overly specific. For example, the following are the schemas resulting from the HAND_GRAB action.

-1	-1	-1	0	-1	0	0	0	0	1	218
-1	-1	-1	1	-1	0	0	0	0	1	214
-1	0	-1	0	-1	0	0	0	0	1	296
-1	1	-1	1	-1	0	0	0	0	1	77
-1	0	-1	1	-1	0	0	0	0	1	291
0	0	-1	1	-1	0	0	0	0	1	454
0	1	-1	1	-1	0	0	0	0	1	117
1	1	-1	1	-1	0	0	0	0	1	40
1	0	-1	1	-1	0	0	0	0	1	163
1	0	-1	0	-1	0	0	0	0	1	173
1	-1	-1	1	-1	0	0	0	0	1	123

Each row is a schema with five context elements and five result elements. The final column gives the number of recorded vectors which support this schema. The SOM method of harvesting does not attempt to generalise over contexts; each instance that has a sufficiently large number of supporting vectors will get its own representation in the SOM. We also tried constraining the number of schemas produced by using a smaller SOM; we found a cut-off below which the SOM produced no schema, and above which it produced too many. Ideally we would like to get one schema for the above examples. To resolve this we used the 10x10 SOM and added a generalising step which simply groups together all schemas with the same result, and replaces any context items which take multiple values with a zero. This leads to the context [0 0 -1 0 -1] in the above example.

4.2 Learning For Continuous Actions

We tried to use the same harvesting approach with our continuous actions. We explain the idea behind this with an example: one might imagine that when the mouth is

touched, the arm sensors will tend to have a constrained range of values, and that this might be represented by a cell in the SOM. However, we found that when real values are in the context, the discrete result elements (such as *mouth_touched*) take on intermediate weight values in the SOM, which do not make it past the thresholding phase. Eventually we abandoned the SOM approach for continuous actions, and used a simple statistical method. Essentially we want to find which sensors are relevant to achieving a particular result. For example, if we want to learn to touch the mouth from any position, then we should only train our neural network with the two arm angles (and ignore all other sensors).

We will now describe the statistical method for one action (the same procedure is repeated for all actions). We record approximately 1000 vectors for each continuous action. The context of our vectors includes all eleven sensors (discrete and continuous). The result part of the vector computes the change in discrete sensors, as done in the discrete action case, and contains five elements. We want to find which sensors are likely to be important in the achievement of each result. We perform the analysis for each of the five results separately. For each result we split our 1000 vectors into two sets. The first set contains the context vectors when that result was zero, and the second set contains the context vectors when the result was (positive) one. To make this concrete consider the result “mouth_touched” and one of the arm movement actions; we create a set of all context vectors where the action led to the mouth being touched (this set turned out to have 54 vectors in our example), and another (larger) set of all context vectors where the action did not lead to the mouth being touched (this set turned out to have 992 vectors in our example). For each of the eleven sensors we now compute an ANOVA, comparing the distribution of values for that sensor both when the action achieves the result, and when it does not. For our example with the result “mouth_touched”, we had the following probabilities of the “null hypothesis” for the eleven sensors:

$6.7 \times 10^{-7}, 0, 0.37, 0.51, 0.90, 0.40, 1.0, 1.0, 1.0, 0.50, 0.56$

This very clearly shows that the first two sensors (corresponding to arm angles) are the only ones relevant to the result “mouth_touched”.

In general we follow this same process for all results. We discard any result elements where the number vectors in the smallest of the two sets (for zero or one) is less than two percent of the size of the original data. This corresponds to a result which is very rarely achieved by this action. After this we are left with results which are frequently achieved by the action, and we need to find which sensors are relevant to achieving the result. We set a threshold of two percent on the probability of the “null hypothesis”; i.e. we only pick up on sensors whose probability of being affected by the result (of 1 or 0) is

less than 0.02. This gives us a list of relevant sensors for each result which can be achieved by this action. This is repeated for all actions.

Now for each result we have a list of all those sensors affecting the result (if any), and the action that caused it. Any result that has a non-empty list of actions and sensors is used to create a new neural network, to represent a Q value function for achieving that result. The inputs to the network are the sensors and the actions. Continuous sensors give real valued inputs to the network, discrete sensors as well as actions give binary inputs to the network. This network is now trained by the NFQ algorithm, using all the transition experiences as training. Any transition experiences that achieve the result are given a maximal reward value; all other transition experiences have zero reward.

4.3 Learning For Composite Actions

After having learnt schemas for discrete actions, and policies for continuous actions, we form composite actions for any new result. This follows Drescher (1991) (and Chaput). Composite actions are formed by finding a chain of actions backwards from a result, which lead to the result being achieved, from a variety of different contexts. Thus the composite action has a set of contexts where it can be invoked, and a single result which it will achieve. The policies learnt above for continuous actions are also treated as composite actions in their own right.

For the next phase of learning we drop the six continuous actions of Table 2 above, and extend the set of actions with the new composite actions. This next phase of learning progresses using the method of harvesting of schemas which was used for the discrete actions. This can find correlations among the composite actions. In particular it can find new results which may be reliably achieved by the policies for continuous actions, because these composite actions are now taken frequently, and in sequences. These new results would be very unlikely to be achieved before the policies were learnt, because there was little chance that one result would be achieved after another by the random operation of individual continuous actions.

This completes our higher level of learning. As with Chaput’s work, in principle there is no reason why we cannot have progressively higher layers on top of this, but we have not explored this yet.

5. Results

To gather training data we ran our simulation to gather 7000 transition experiences, where actions were randomly taken. We randomly repositioned the hand after each movement. This is simply for expedience in gathering training data; it covers most of the space in less time. A random walk would be more faithful to a real

infant’s experience, but it needs to run for longer in order to cover most regions of the space. From this data we harvested schemas for discrete actions, resulting in two schemas:

$$\begin{array}{cccccccc} -1 & 0 & -1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 1 \end{array}$$

The first is for the LOOK_AT_OBJECT action and states that this action results in the *fovea_sees* sensor turning on. The second is for the HAND_GRAB action and states that it results in the *hand_holding_object* sensor turning on.

For the continuous actions we found actions to achieve three results. Firstly the result of *mouth_touched* was found to be caused by the four arm movement actions, with the sensors *lower_arm_angle* and *upper_arm_angle* being relevant. From this a Neural network with six inputs was created (i.e. four actions and two sensors) to represent the Q value function and learn a policy to achieve *mouth_touched*. Secondly the result of *hand_touched* was found to be caused by the four arm movement actions, with the sensors *seen_objects_distance* and *seen_objects_angle* and *mouth_touched* being relevant. The *mouth_touched* sensor is spurious here; it should not be relevant to achieving the goal, however it is probable that the fact that the object is often located close to the mouth causes the relationship to be inferred. From this a neural network was created to represent the Q value function and learn a policy to achieve *hand_touched*. Finally, the result of *fovea_sees* was found to be caused by the two eye movement actions, with the sensors *eye_angle* and *upper_arm_angle* being relevant. The *upper_arm_angle* sensor is spurious here. We did not train a network for this as it was deemed to be too simplistic, given the limited space in which the eye can rotate.

We trained the networks for the first two results using 1016 transition experiences (the same experiences were used in training for the two different results). This is much less data than had been used for the initial analysis; this reduction was simply to make the learning iterations faster; the data proved to be sufficient. We used 100 RPROP iterations for each training phase, followed by a Q-learning sweep, and this whole process was iterated approximately fifteen times to produce a reasonable result. A reasonably effective policy was learnt both for achieving *mouth_touched* and *hand_touched*. The policies were successful in achieving their result about 50% of the time, sometimes getting stuck in a back and forward loop. We suspect that if there were more than four possible actions it would likely lead to smoother and more reliable policies (although more training data and time would be required).

Given the simplicity of our scenario we were able to compile the composite actions manually. This gave us actions that could achieve *fovea_sees* by looking at an

object, and which could achieve *hand_holding_object* by moving the hand (using the policy learnt above) to the block in the fovea, and performing HAND_GRAB. The final harvesting of schemas using these composite actions was able to find the schema to achieve the result *mouth_touched_object* (i.e. to grab an object and take it to the mouth). Admittedly this result is a little contrived as the simulation has been designed just to make this possible, however it does show that the techniques combined here can work in principle.

6. Conclusion

This work has brought together two techniques in an effort to model sensorimotor skill acquisition, with a view to modelling Piagetian sensorimotor developments. We will first briefly evaluate the effectiveness of these two techniques, and then the combination. Firstly, Riedmiller’s NFQ works very well in our setting. It is particularly convenient that the same training experiences can be used to train for different goals. We can also illustrate its strength by comparing it with a naive reinforcement learning approach to our simulation; for example in learning to move the hand to touch an object in the fovea, a naive approach would require touching the object on each trial to propagate reward back to the actions that led there. Furthermore, the object would need to be in the fovea all the time. In contrast the NFQ approach can use any transition experience of a hand movement as part of its training data; the experiences merely describe the resulting state, and are separated from any particular goal. There are issues over biological plausibility however as this is hardly a realistic model of what an infant does; instead it is likely that memories are abstracted/generalised in some way rather than being stored precisely.

Secondly, Chaput’s method for harvesting schemas works reasonably well, apart from the issue of generalising over contexts, mentioned in Section 4 above. Despite the success of this method, we suspect that a simpler statistical averaging technique may obtain the same results in a more efficient way. The clustering ability of the SOM has not been exploited by Chaput, and we suspect that the basic idea of a hierarchy of SOMS could lead to a much more powerful learning approach if used in a different way. This is an interesting area for future work.

The combination of techniques we have used works for our simple examples but needs to be tested on a wider range of example scenarios. There will be challenges to be overcome when the space is larger, for example if sensors include more inputs from vision. The training data required will become inordinately large, and it is likely that there will need to be some gradual way to build up abstractions on sensor data, so that the learning is constrained initially, either by using some innate abstractions, or the lifting of constraints (Lee et al., 2007). Fur-

thermore, the efficiency of the learner could be improved considerably, in a number of ways. One obvious example is by using online learning, where actions are taken using a greedy strategy, which would mean that less of the state space would need to be recorded. This inefficiency does not concern us for the moment because we are currently unsure how a (relatively generic) learning system could be built to learn high level knowledge from its experiences. Once this question has been answered (even with an inefficient mechanism) we can then investigate optimisations.

Finally, let us look at this as a model of Piaget’s theory (which is our long-term goal), the work has achieved what it set out to do, in allowing continuous actions to be integrated with a schema learning mechanism, however, many aspects remain to be addressed to create a convincing model of Piagetian learning. To compare with Chaput’s (2004) work, we note that we have not implemented synthetic items. Synthetic items seem particularly useful when there are hidden aspects of the world, for example: (1) objects which have an existence in the world, but are not always observable (Drescher’s original motivation for introducing the synthetic item); (2) positions which are not accurately observable, as in Chaput’s robot forager world; (3) hidden properties such as the weight of an object (Morrison et al., 2001). These “hidden aspects” do not feature in our simulation for the moment, but we expect that they will play a part in future work when we want to model later stages of development. Nevertheless, we can claim a strong similarity with these works because of the hierarchical nature of the learning; the synthetic item is used to notice a correlation between the activation and success of schemas over time, which is also a function performed by our higher layer of learning.

The next immediate step for this work would be to allow the policies learnt on continuous actions to be adjusted to achieve new goals. We have frozen the policies once they are learnt, but to model Piagetian assimilation and accommodation in stage 4 means-end behaviours, there should be the possibility to adjust a policy when a new, slightly different, goal needs to be achieved.

Acknowledgements

We are grateful to Joey Lam, Bang Wu, and Cory Lowson who worked on various parts of the software.

References

- Chaput, H. H. (2004). *The constructivist learning architecture: a model of cognitive development for robust autonomous robots*. PhD thesis, AI Laboratory, The University of Texas at Austin. Supervisors: Kuipers and Miikkulainen.
- Cohen, L. B. (1998). An information-processing approach to infant perception and cognition. In Simion, F. and Butterworth, G., (Eds.), *The Development of Sensory, Motor, and Cognitive Capacities in Early Infancy*, pages 277–300. East Sussex: Psychology Press.
- Drescher, G. L. (1991). *Made-Up Minds, A Constructivist Approach to Artificial Intelligence*. MIT Press.
- Hart, S., Sen, S., and Grupen, R. (2008). Generalization and transfer in robot control. In *Proceedings of the Eighth International Conference on Epigenetic Robotics, University of Sussex, July 30-31*.
- Lee, M. H., Meng, Q., and Chao, F. (2007). Staged competence learning in developmental robotics. *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems*, 15(3):241–255.
- Morrison, C. T., Oates, T., and King, G. (2001). Grounding the unobservable in the observable: The role and representation of hidden state in concept formation and refinement. In *In AAAI Spring Symposium on Learning Grounded Representations*, pages 45–49. AAAI Press.
- Perotto, F. S. and Álvares, L. O. (2006). Learning regularities with a constructivist agent. In *AAMAS ’06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 807–809, New York, NY, USA. ACM.
- Piaget, J. (1936). *The Origins of Intelligence in Children*. London: Routledge & Kegan Paul. (French version published in 1936, translation by Margaret Cook published 1952).
- Prince, C., Helder, N., and Hollich, G. (2005). Ongoing emergence: A core concept in epigenetic robotics. In Berthouze, L., Kaplan, F., Kozima, H., Yano, H., Konczak, J., Metta, G., Nadel, J., Sandini, G., Stojanov, G., and Balkenius, C., (Eds.), *Proceedings of EpiRob05 - International Conference on Epigenetic Robotics*, pages 63–70. Lund University Cognitive Studies.
- Riedmiller, M. (2005). Neural reinforcement learning to swing-up and balance a real pole. In *Int. Conference on Systems, Man and Cybernetics, 2005, Big Island, USA*.
- Riedmiller, M. and Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The rprop algorithm. In Ruspini, H., (Ed.), *IEEE International Conference on Neural Networks (ICNN), San Francisco*, pages 586–591.
- Stojanov, G. (2001). Petitagé: A case study in developmental robotics. In Balkenius, C., Zlatev, J., Kozima, H., Dautenhahn, K., and Breazeal, C., (Eds.), *Proceedings of Epigenetic Robotics 1*.

Caregiver's Auto-mirroring and Infant's Articulatory Development Enable Vowel Sharing

Hisashi Ishihara* Yuichiro Yoshikawa** Minoru Asada**,*

*Graduate School of Eng., Osaka University

**Asada Synergistic Intelligence Project, ERATO, JST

2-1 Yamadaoka, Suita, Osaka, 565-0871 Japan

hisashi.ishihara@ams.eng.osaka-u.ac.jp, yoshikawa@jeap.org, asada@ams.eng.osaka-u.ac.jp

Abstract

We extend the auto-mirroring guidance model, which explains the process of sharing vowels between a caregiver and an infant, by introducing two transitional elements related to the infant's articulatory development: One is the accuracy of the infant's articulation improving along with the separation of his/her vowel prototypes. The other is the transition of the caregiver's auditory perception of mapping the infant's vowels onto her own ones. The extended model can simulate several additional aspects of vowel development, e.g., the rapid separation of infant vowels and their convergence and the transient rise of stretching motherese. Simulation results suggest a new picture of the process of vowel development, which explains how there are two transitional aspects of vowel separation and guidance, and they also suggest hypotheses on the causes of vowel separation and a caregiver's motherese.

1. Introduction

The process of sharing vowels with caregivers seems to be the first developmental step of an infant's language development. Kuhl and her colleagues pointed out the importance of regarding the process of vowel development as dynamic interaction between perceptual development, articulatory development, and a caregiver's address to the infant (Kuhl et al., 2008). However, their model is still too conceptual to understand the computational mechanism underlying such processes of development. To reveal such a mechanism, the use of synthetic studies has been considered one of the most promising approaches (Asada et al., 2009).

Some of these studies have focused on the perceptual development needed to learn a caregiver's vowel categories from her speech (McMurray et al., 2009, Vallabha et al., 2007). However, how the caregiver addresses her infant in speech should be taken into account to understand such perceptual development.

This addressing by the caretaker seems to depend on her observations and understanding of the infant's developmental stage, based on such input as the quality of vocalizations (Gros-Louis et al., 2006, Bloom and Lo, 1990).

Mutual imitation (Masur and Olson, 2008, Kokkinaki and Kugiumutzakis, 2000) is a typical and highly significant instance of caregiver-infant interaction. de Boer (de Boer, 2000) and Oudeyer (Oudeyer, 2005) have suggested that imitative interaction plays important roles in sharing vowels between agents. In particular, Oudeyer showed computationally that shared prototypes can be self-organized by virtue of a perceptual bias around vowel prototypes (perceptual magnet effect (Kuhl, 1991)). However, this research lacks consideration of an inevitable hurdle to infant development, namely the physical differences between the caregiver and infant: They cannot produce the same vowel sounds since their articulatory organs are very different from each other (Vorperian and Kent, 2007).

Miura *et al.* showed that a robot could acquire shared vowels with a human interactant who imitates its vocalization with a different articulatory organ from it (Miura et al., 2007). To investigate what properties of a caregiver's imitation permit the sharing of vowels, Ishihara *et al.* have constructed a computational model of the caregiver-infant imitative interaction (Ishihara et al., 2008). They proposed that an infant's prototypes are guided toward the anticipated ones by distinctive biases in the caregiver's imitation, made as if she were imitating not only the infant's utterance but also both her own usual utterance style (sensorimotor magnet bias) and her own previous utterance (auto-mirroring bias). Also, they considered the differences in utterable vowels. However, there remain other aspects of infant articulatory development such as proficiency of articulation control through self-monitoring experience of produced sound (Oller and Eilers, 1988) and expansion of one's utterable vowel area in vowel space (Ishizuka et al., 2007, Rvachew et al., 2006). Such immature articulation activities should be

introduced in their model, given the likelihood that the developmental conditions of perception and articulation could affect each other (Vihman and Nakai, 2003, van Beinum et al., 2001, Oller and Eilers, 1988).

Introducing the elements of an infant’s articulatory development in the auto-mirroring guidance model would allow us to examine the causes and effects of several phenomena appearing in a real infant’s vowel development, such as 1) rapid separation of distribution clusters of infant utterances and its subsequent convergence (Ishizuka et al., 2007) and 2) stretching motherese in which the distributional profile of the mother’s vowels addressed to her infant tends to stretch compared to that used in addressing other adults (Kuhl et al., 1997).

In this paper, we extend the auto-mirroring guidance model by introducing two transitional elements related to an infant’s articulatory development. First, the accuracy of an infant’s articulation is assumed to improve along with the separation of the infant’s own prototypes, since sensorimotor learning of these prototypes would be easier after they are separated more widely. Then, the caregiver’s auditory perception that maps the infant’s vowels on the caregiver’s own vowels is also modulated according to the infant’s articulatory development. We report that the extended model can simulate the rapid separation of an infant’s vowels and their convergence as well as the transient rise of stretching motherese. Furthermore, we suggest a new picture of the process of vowel development that explains that there are two transitional aspects, i.e., separation and the guidance, and we suggest hypotheses on the causes of infant vowel separation and the caregiver’s use of motherese.

2. Auto-mirroring guidance model

2.1 Overview

This model consists of imitation mechanisms for both a caregiver and an infant and a learning mechanism of a sensorimotor map for the infant. Imitation mechanisms convert the other’s vowel sound into the imitator’s own articulation command to produce the imitation vowel sound.

Another feature of this mechanism is to contain possible biasing elements, i.e., sensorimotor magnets and auto-mirroring bias, in the caregiver’s imitation arising from her anticipation of her infant’s utterance. Sensorimotor magnets are kinds of convergence bias of perception and articulation around the caregiver’s vowel prototypes. Part of this characteristic seems to originate from Kuhl’s perceptual magnet effect (Kuhl, 1991), which is a perceptual warp around a listener’s phoneme prototypes. This bias can be seen as an effect of the caregiver’s unconscious anticipation of her infant to articulate

vowel prototypes in a mother language. Another bias, auto-mirroring bias, is a kind of mixing bias of the other and the self, in which the perception of the other’s vowel is attracted toward the perceiver’s own last utterance. This bias can be seen as an effect of the caregiver’s anticipation that her infant will imitate her utterance correctly.

Figure 1 shows an overview of the model. At the t -th step of interaction, the infant utters a vowel $\mathbf{s}'(t) \in \mathcal{R}^{N_s}$ by the articulation command $\mathbf{a}'(t) \in \mathcal{R}^{N_a}$, and the caregiver listens to vowel sound $\mathbf{s}'(t)$ and utters $\mathbf{s}(t) \in \mathcal{R}^{N_s}$ by the articulation $\mathbf{a}(t) \in \mathcal{R}^{N_a}$ as an imitation of $\mathbf{s}'(t)$. Next, the infant listens to $\mathbf{s}(t)$ and updates his/her sensorimotor map based on both his/her articulation $\mathbf{a}'(t)$ and the caregiver’s reply $\mathbf{s}(t)$ and then tries to imitate $\mathbf{s}(t)$ by the articulation $\mathbf{a}'(t+1)$ using the updated map. The learning mechanism of the map is explained in section 2.3 below.

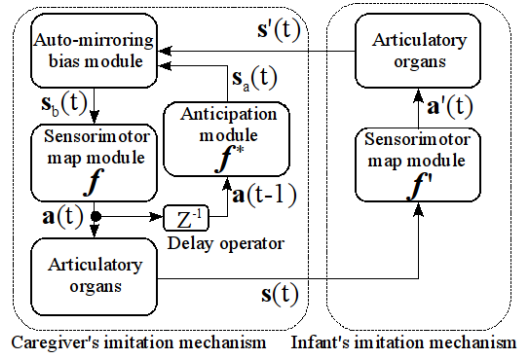


Figure 1: Overview of proposed model consisting of two imitation mechanisms for a caregiver and an infant.

2.2 Imitation mechanism

The caregiver’s imitation $\mathbf{a}(t)$ is modeled by the formula:

$$\mathbf{a}(t) = f(\mathbf{s}_b(t); \mathbf{p}_i, \lambda), \quad (1)$$

$$\mathbf{s}_b(t) = (1 - \eta)\mathbf{s}'(t) + \eta\mathbf{s}_a(t), \quad (2)$$

$$\mathbf{s}_a(t) = f^*(\mathbf{a}(t-1); \mathbf{p}_i^*), \quad (3)$$

where $f : \mathcal{R}^{N_s} \rightarrow \mathcal{R}^{N_a}$ is her sensorimotor map at the t -th step based on her prototypes $\mathbf{p}_i \in \mathcal{R}^{N_a}$, while $f^* : \mathcal{R}^{N_a} \rightarrow \mathcal{R}^{N_s}$ is its quasi-inverse map at the t -th step based on the infant’s anticipated prototypes $\mathbf{p}_i^* \in \mathcal{R}^{N_s}$. This term is used because \mathbf{p}_i^* represents the infant’s producible vowels that can be mapped onto the caregiver’s prototypes, i.e., the caregiver anticipates her infant matching her own prototypes \mathbf{p}_i^* with \mathbf{p}_i .

The caregiver’s articulation $\mathbf{a}(t-1)$ is input to her quasi-inverse sensorimotor map f^* and converted to the anticipation $\mathbf{s}_a(t)$ of her infant’s imitation of the articulation $\mathbf{a}(t-1)$. The anticipation is mixed with real infant utterance $\mathbf{s}'(t)$ with the mixing rate η ($0 \leq \eta \leq 1$), and this attraction is called the auto-mirroring bias. Then the attracted perception $\mathbf{s}_b(t)$ is converted to the articulation command $\mathbf{a}(t)$ by the sensorimotor map f , where its output is attracted

to prototypes $\mathbf{p}_i (i = 1, \dots, M)$ with the converging degree $\lambda (0 \leq \lambda \leq 1)$. This convergence is called the sensorimotor magnets. Thus, we can control the strength of the caregiver's biases, that is, the sensorimotor magnets and auto-mirroring bias, by changing the parameters λ and η .

2.2.1 Sensorimotor map

We model the sensorimotor map f with one of the linear regression mixture models, a Normalized Gaussian Network (NGnet) (Sato and Ishii, 2000, Moody and Darken, 1989). An NGnet has M Gaussian functions $g_i (i = 1, \dots, M)$ as basis functions in input space and maps the input data with mixed linear regression functions. Each mixture rate is decided according to the distance between the input and the center of each Gaussian, each of which has charge of one linear regression function. NGnet determines the caregiver's articulation $\mathbf{a}(t)$ by

$$\mathbf{a}(t) = f(\mathbf{s}_b(t); \mathbf{p}_i, \lambda) \quad (4)$$

$$= \sum_{i=1}^M \frac{g_i(\mathbf{s}_b(t); \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}{\sum_{j=1}^M g_j(\mathbf{s}_b(t); \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \mathbf{W}_i(\mathbf{p}_i, \lambda, \boldsymbol{\mu}_i) \mathbf{s}_b(t), \quad (5)$$

where $\boldsymbol{\mu}_i \in \mathbb{R}^{N_s}$ and $\boldsymbol{\Sigma}_i \in \mathbb{R}^{N_s \times N_s}$ are the center vector and the variance-covariance matrix of the i -th Gaussian. λ is a parameter that sets the eigenvalue of the representation matrix of linear transformation $\mathbf{W}_i(\mathbf{p}_i, \lambda, \boldsymbol{\mu}_i) \in \mathbb{R}^{N_a \times (N_s + 1)}$ to $(1 - \lambda)$. Furthermore, $\mathbf{s}_b(t) \equiv [\mathbf{s}_b^T(t), 1]^T \in \mathbb{R}^{N_s + 1}$ is the augmented matrix of $\mathbf{s}_b(t)$.

Figure 2 shows how sensorimotor magnets are modeled and controlled by the setting of λ when we assume that the NGnet has one Gaussian unit ($M = 1$), where the one-dimensional inputs (infant's vowel sounds) are normally distributed around its center. Each input is mapped by a matrix of linear transformation \mathbf{W}_1 , and thus the distribution of the outputs (caregiver's articulation commands) are determined by the eigenvalue (slope here) of the matrix: The smaller the eigenvalue $(1 - \lambda)$ of the transformation matrix \mathbf{W}_1 is, the more the distribution gathers around the image of the Gaussian center $\boldsymbol{\mu}_1$ under \mathbf{W}_1 , namely $\mathbf{W}_1 \boldsymbol{\mu}_1$. Therefore, we regard the image as a prototype to represent sensorimotor magnets, namely $\mathbf{p}_i \equiv \mathbf{W}_i \boldsymbol{\mu}_i$. Furthermore, we regard the center of Gaussian $\boldsymbol{\mu}_i$ as anticipated prototypes \mathbf{p}_i^* , namely $\mathbf{p}_i^* \equiv \boldsymbol{\mu}_i$, since they are mapped onto the caregiver's prototypes.

The caregiver's quasi-inverse sensorimotor map is also modeled by another NGnet so that it can map the caregiver's prototype \mathbf{p}_i to \mathbf{p}_i^* as opposed to the sensorimotor map that maps \mathbf{p}_i^* to \mathbf{p}_i . This quasi-inverse map works like a predictor of the infant's imitation and is updated at every step as the sensorimotor map changes, as mentioned in section 3.2 below.

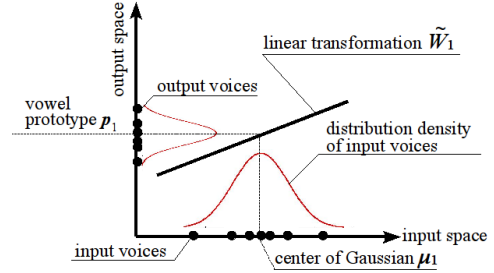


Figure 2: Illustration of how an NGnet constructs the sensorimotor magnets with one transformation matrix

2.3 Learning mechanism for an infant

An infant has the immature sensorimotor map f' represented by an NGnet that has M -Gaussian functions and learns its parameters in the T -th step of the interaction based on the n -step history $H(T)$ of the pairs of the infant's own articulation and the caregiver's reply. Namely, $H(T) = \{\mathbf{a}'(t), \mathbf{s}(t) | t = T - n + 1, \dots, T\}$. Here, the infant task is to tune the parameters $\{\boldsymbol{\mu}_i'(T), \boldsymbol{\Sigma}_i'(T), \mathbf{W}_i'(T) | i = 1, \dots, M\}$ of her sensorimotor map so that it can represent the input-output relationship from $\mathbf{s}(t)$ to $\mathbf{a}'(t)$ within $H(T)$. We use the EM algorithm (Sato and Ishii, 2000, Dempster et al., 1977), which is one of the maximum likelihood estimation methods for mixture models, to estimate the most appropriate parameters.

As a result of this update, the infant's prototypes $\mathbf{p}_i' \equiv \mathbf{W}_i' \boldsymbol{\mu}_i'$ are also updated at every step. The final goal of his/her development is to match his/her prototypes \mathbf{p}_i' to his/her anticipated prototypes by the caregiver \mathbf{p}_i^* , which he/she can not observe directly.

3. Extensions of the model

3.1 Accuracy of infant's articulation

To consider the possible effects of the articulatory development of an infant on the process of sharing vowels, we introduce a simplified model. The accuracy of the infant's articulation control is considered to be improved through self-monitoring of the sound resulting from his/her attempts at articulation (Oller and Eilers, 1988). Therefore, this accuracy is modeled so as to be related to the current distribution of the infant's produced sounds: We assume that his/her articulation error can be represented by a variance of a Gaussian distribution around his/her target articulation $\mathbf{a}'(t)$, and this variance is determined based on the extent to which his/her current prototypes are separated from each other.

Given the infant's target articulation $\mathbf{a}'(t)$, the produced articulation is determined by

$$\mathbf{a}'(t) = \mathcal{N}(\mathbf{a}'(t), \sigma^2(S(t), h)), \quad (6)$$

$$\sigma(S(t), h) = \frac{150}{1 + \exp\{0.02(S(t) - h)\}}, \quad (7)$$

$$S(t) = \sum_{i=1}^M \left(\frac{|\mathbf{p}_i'(t) - \sum_{j=1}^M \frac{\mathbf{p}_j'(t)}{M}|}{M} \right), \quad (8)$$

where $\mathcal{N}(\mathbf{a}', \sigma^2)$ represents a manipulation to add a Gaussian noise whose variance is σ^2 to the target articulation \mathbf{a}' . The variance σ^2 depends on the degree of the prototypes' separation $S(t)$, and their relationship is controlled by the parameter h . Here, we can control the difficulty of articulation development in the simulation by changing the parameter h ; for example, the larger h is, the larger σ^2 is, even under the same condition of $S(t)$.

3.2 Transitions of caregiver's perception

In the previous model, we assumed that anticipated prototypes \mathbf{p}_i^* are fixed throughout interactions. However, an experimental result of category identification of infants' vowels by caregivers shows that an adult's perception of infant vowels can alter, e.g., the geometry of perceived prototypes expands and shifts in vowel space as the infant becomes older (Vorperian and Kent, 2007, Ishizuka et al., 2007, Rvachew et al., 2006, Kuhl and Meltzoff, 1996). Therefore, anticipated prototypes should be altered through interactions for a more valid simulation.

We introduce the expansion of the geometry of anticipated prototypes in a fixed expansion rate as a first implementation. Anticipated prototypes are determined by

$$\mathbf{p}_i^*(t) = \mathbf{p}_i^*(0) + \frac{t}{T_L} (\mathbf{p}_i^*(T_L) - \mathbf{p}_i^*(0)), \quad (9)$$

$$\mathbf{p}_i^*(0) = \sum_{i=1}^M \frac{\mathbf{p}_i^*(T_L)}{M}, \quad (10)$$

where T_L is the number of total interaction steps and $\mathbf{p}_i^*(T_L)$ are fixed values in the current model.

4. Simulation of mutual imitation

4.1 Procedure

A caregiver imitates her infant's utterance at every step while her infant basically tries to imitate the caregiver's utterance at every step but sometimes utters randomly, i.e., the infant tries to utter one of the prototypes every step until the n -th step and continues to do so every fifth step even after the n -th step interaction. Furthermore, until the n -th step has passed, the infant does not update his/her sensorimotor map, since she can not utilize enough learning data. In this simulation, we set $n = 500$ and $T_L = 5000$.

4.2 General settings

We assume each vowel sound is represented by a two-dimensional vector, since vowel prototypes are known to be distinguishable at two frequency peaks, which are called first formant and second formant. Furthermore, for simplicity of simulation, we assume that an articulation command can be represented by the same vector as that of the vowel sound produced by the articulation, i.e., $\mathbf{s}(t) = \mathbf{a}(t)$ and $\mathbf{s}'(t) = \mathbf{a}'(t)$.

Figure 3 shows an overview of the settings of the caregiver's prototypes \mathbf{p}_i (blue dots), the anticipated

prototypes $\mathbf{p}_i^*(t)$ (red dots), and the infant's prototypes $\mathbf{p}_i'(t)$ (black dots) in the vowel/articulation feature space. Anticipated prototypes are set to be located at a distance from the caregiver's prototypes, since their articulation organs are different and thus a difference in their vowels can be expected. We set the number of prototypes M to 5, imagining a Japanese caregiver. The infant's initial prototypes are set more closely together, since a real infant's categories are not so widely separated from each other in the early period of development. The caregiver's prototypes are fixed throughout the interactions, while anticipated prototypes are gradually expanded as the interaction proceeds based on eq. (9).

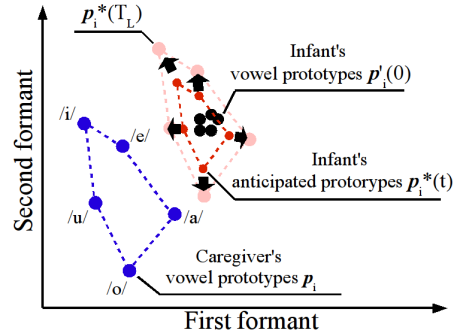


Figure 3: Overview of settings of the initial states of infant's prototypes (black dots) and caregiver's prototypes (blue dots) and the anticipated prototypes (red dots).

4.3 Settings of the caregiver

Assuming a Japanese caregiver and infant, we determined the caregiver's prototypes \mathbf{p}_i and anticipated prototypes at the last step $\mathbf{p}_i^*(T_L)$ as

$$\{\mathbf{p}_i\} = \left\{ \begin{array}{ccccc} 700 & 400 & 400 & 600 & 500 \\ 1200 & 1700 & 1300 & 1500 & 1000 \end{array} \right\}, \quad (11)$$

$$\mathbf{p}_i^*(T_L) = \mathbf{p}_i + \begin{array}{c} 400 \\ 600 \end{array} \quad (i = 1, \dots, M). \quad (12)$$

We determined the parameters of the caregiver's sensorimotor map so that all of our assumptions are satisfied as follows:

$$\boldsymbol{\mu}_i(t) = \mathbf{p}_i^*(t) \quad (i = 1, \dots, M), \quad (13)$$

$$\boldsymbol{\Sigma}_i = \begin{array}{cc} 3600 & 0 \\ 0 & 3600 \end{array} \quad (i = 1, \dots, M), \quad (14)$$

$$\tilde{\mathbf{W}}_i(\mathbf{p}_i, \lambda, \boldsymbol{\mu}_i(t)) = ((1 - \lambda)\mathbf{I}, \mathbf{p}_i - (1 - \lambda)\boldsymbol{\mu}_i(t)) \quad (i = 1, \dots, M, 0.0 \leq \lambda \leq 1.0). \quad (15)$$

In addition, we determined the parameters of the caregiver's quasi-inverse sensorimotor map so that all of our assumptions are satisfied as follows:

$$\boldsymbol{\mu}_i^* = \mathbf{p}_i \quad (i = 1, \dots, M), \quad (16)$$

$$\boldsymbol{\Sigma}_i^* = \begin{array}{cc} 3600 & 0 \\ 0 & 3600 \end{array} \quad (i = 1, \dots, M), \quad (17)$$

$$\tilde{\mathbf{W}}_i^*(\mathbf{p}_i^*(t), \boldsymbol{\mu}_i^*) = (\mathbf{I}, \mathbf{p}_i^*(t) - \boldsymbol{\mu}_i^*) \quad (i = 1, \dots, M). \quad (18)$$

Note that the infant’s anticipated prototypes $\mathbf{p}_i^*(t)$ change through the interactions according to eq. (9).

From the simulation results of the previous model, we know that infant prototypes $\mathbf{p}'_i(t)$ are gradually guided toward the anticipated prototypes $\mathbf{p}_i^*(t)$ by virtue of association with the caregiver’s sensorimotor magnets and auto-mirroring bias. The degree of such guidance depends on the degree of their strengths, and $(\eta = 0.5, \lambda = 0.6)$ is the setting pair that exerts the guidance effect most strongly. Therefore, we selected this setting pair for the current simulation.

5. Results

We simulated the caregiver-infant imitative interaction under several conditions of difficulty in articulatory development: (a) $h = 0$, (b) $h = 150$, (c) $h = 300$, and (d) $h = 450$.

5.1 Fading of articulation error

Figure 4 shows the degree of an infant’s articulation error $\sigma(t)$ processed throughout interaction steps under each condition of h . We can see that the infant’s articulation error $\sigma(t)$ is larger throughout the interactions under the condition where the infant’s articulation development is more difficult (h is larger), as we had expected. In addition, we can see that the error $\sigma(t)$ tends to decrease step by step rapidly in the first half of this period, especially under conditions (a), (b), and (c).

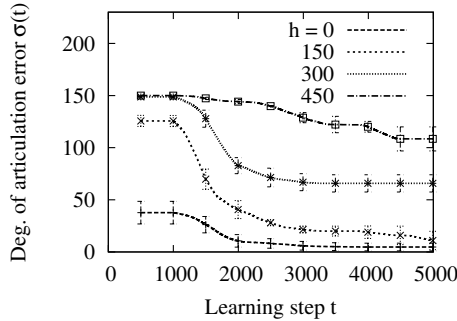


Figure 4: Differences in the transitions of infant articulation error $\sigma(t)$ under several conditions of difficulty h in articulation development.

5.2 Transitions of vowel distribution

Figure 5 illustrates the transitions of vowel distributions of both the caregiver and the infant under each condition of h . In these figures, utterances of the infant $\mathbf{s}'(t)$ (red dots) and those of the caregiver $\mathbf{s}(t)$ (blue dots) and the infant’s prototypes $\mathbf{p}'_i(t)$ (black dots) are plotted during each of three periods (at the last step for $\mathbf{p}'_i(t)$): first 1000 steps (left box), middle 1000 steps (middle box), and final 1000 steps (right box). The apexes of the red pentagons represent the infant’s anticipated prototypes $\mathbf{p}_i^*(t)$ at the last step of each period, while those of the blue pentagons represent the caregiver’s prototypes \mathbf{p}_i .

Large variations in the distributional patterns of utterances between conditions indicate that the difficulty of articulatory development heavily affected both the infant’s learning and the interactions: The infant’s utterances and prototypes were distributed more widely when h was larger and the number of uttered categories was different, i.e., only three vowel categories were uttered during the final period in condition (a) while five categories were uttered under conditions (c) and (d).

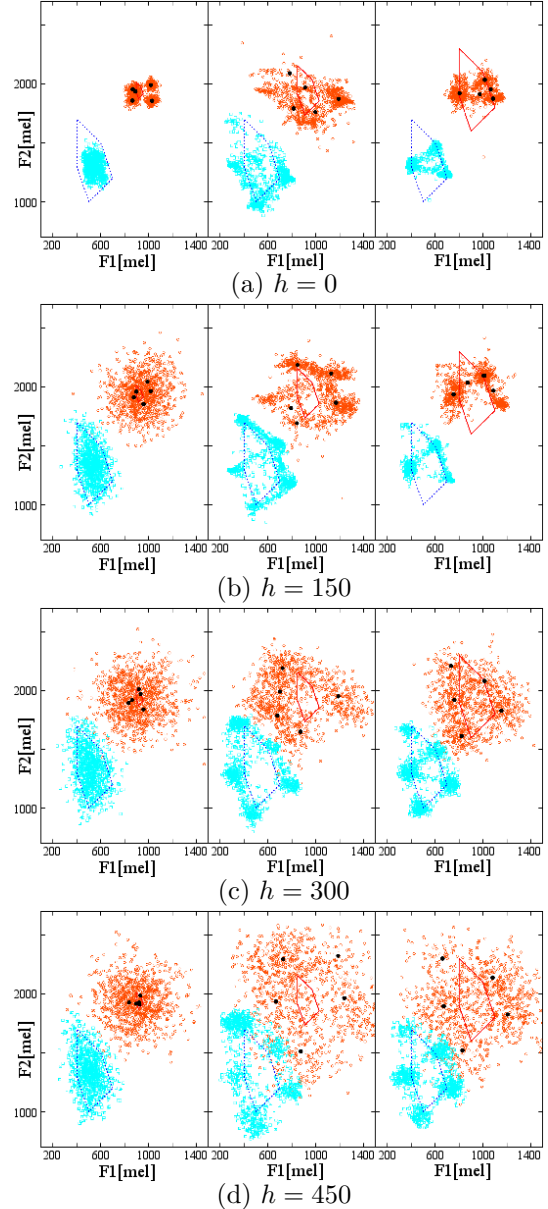


Figure 5: Transitions of vowel distributions of both the caregiver (blue dots) and the infant (red dots) and those of the infant’s prototypes (black dots) under different conditions of h . The geometry of the caregiver’s prototypes (apexes of blue pentagons) and the infant’s anticipated prototypes (apexes of red pentagons) are also depicted.

5.3 Separation of prototypes

Ishizuka and Mugitani investigated the distributions of real infants' utterances during the age span of 4-60 months (Ishizuka et al., 2007). They showed that the geometry of vowel categories tend to expand until age 24 months and the speed of their separation is rapid in the early stage and then becomes slower.

Figure 6 shows the transitions of the separation degrees $S(t)$ of infant prototypes defined in eq. (8). The counterpart for the anticipated prototypes is also depicted as a reference by the solid line. We can see that the infant's prototypes tend to expand rapidly, particularly in the first half of the period, and then the speed of expansion becomes slower until the prototypes gradually converge. Interestingly, this basically reproduces the real transition reported in the previous study (Ishizuka et al., 2007).

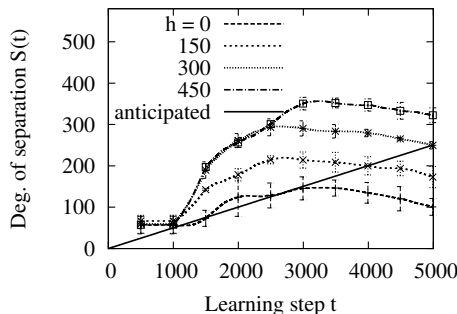


Figure 6: Transitions of the separation degree $S(t)$ of the infant prototypes under different conditions of h . The counterpart of the anticipated prototypes is also depicted.

5.4 Process of sharing vowels

Figure 7 shows the transition of the sharing degree of prototypes between the caregiver and the infant, which represents the as-a-whole closeness of the anticipated prototypes $\mathbf{p}_i^*(t)$ with the infant prototypes $\mathbf{p}'_i(t)$. The sharing degree is evaluated by the formula:

$$D(t) = \sum_{i=1}^M \frac{\text{Min}(\{\|\mathbf{p}_i^*(t) - \mathbf{p}'_j(t)\|\}_{j=1, \dots, M})}{M}, \quad (19)$$

where $D(t)$ represents the as-a-whole distance of anticipated prototypes from the infant's prototypes at the t -th step of the interaction; consequently, the sharing degree is higher when this index is lower. The as-a-whole distance $D(t)$ is small in the initial state under all conditions, and these values are not so different from each other since the infant's prototypes are set to gather around the initial point of anticipated prototypes $\mathbf{p}_i^*(0)$. The as-a-whole distance $D(t)$ continued to increase under conditions (a) and (b) to the end, while it increased rapidly during the first half of the period and then began to decrease under conditions (c) and (d).

We can also see such transitions in Fig. 5. Under conditions (a) and (b), infant prototypes did not separate so widely from each other and therefore they were guided to a smaller number of anticipated prototypes, indicating that the large as-a-whole distance remained. On the contrary, under conditions (c) and (d), the geometry of infant prototypes expanded more widely than did that of anticipated prototypes in the first half of the period, and then the prototypes were located near all of the anticipated prototypes, indicating an inverted U-shape transition of the as-a-whole distance.

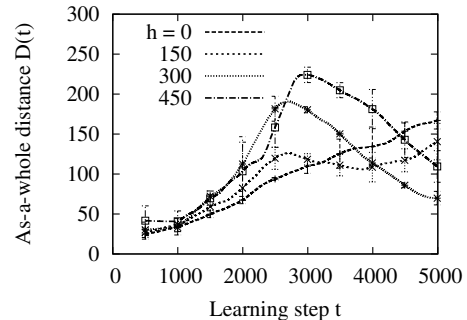


Figure 7: Transitions of the as-a-whole distance $D(t)$ of anticipated prototypes from infant prototypes under different conditions of h .

6. Discussion

6.1 Dominance of either of two aspects

There seems to be two aspects in the transitional process of sharing vowels: the separation of prototypes and the guidance of prototypes toward anticipated prototypes. The investigation in the previous work (Ishihara et al., 2008) revealed that the aspect of guidance can be caused by the balanced action of two biases in the caregiver's imitation, i.e., sensori-motor magnets and auto-mirroring bias. The other aspect, the separation of prototypes, can be considered a consequence of the infant's articulation error due to the larger error results in the larger distribution of the infant's utterances.

A simulated developmental process can be divided into two stages by focusing on which aspect is dominant. In the first stage, the aspect of separation is dominant due to the large articulation error. In this stage, prototypes become separated from each other because the separating effect surpasses the guiding effect at some point. The dominance of the aspect of separation becomes weakened gradually as prototypes become separated from each other, since an infant's articulation error decreases with the separation, as modeled in eq. (8). Then the next stage, where the aspect of guidance is dominant, begins. In this stage, the infant's prototypes can be guided toward anticipated prototypes as in the latter half of the period under conditions (c) and (d) shown in Fig. 7, since the guiding effect surpasses the sepa-

rating effect this time at some point.

The prototypes can be effectively shared when the transition of these dominances is achieved appropriately: There seems to be adequate difficulty h of articulatory development, as under condition (c), where all prototypes can correspond to any of the anticipated prototypes. If the separating effect is too weak, as under conditions (a) and (b), the prototypes cannot sufficiently spread to correspond to all anticipated prototypes. On the other hand, if the separation effect is too strong, as under condition (d), the prototypes are not guided sufficiently, since the separating effect could cancel out the guiding effect.

6.2 Conditions for rise of motherese

We can find the rise of motherese particularly in the middle of the period in Fig. 5 (c). The left side of Fig. 8 illustrates how the geometry of centers of the caregiver’s vowel clusters (blue pentagon with solid line) is stretched compared to that of her usual ones, or her prototypes (blue pentagon with dotted line). By comparing these results with those in the right side of Fig. 8, which shows the motherese reported by Kuhl *et al.* (Kuhl *et al.*, 1997), we can see that this stretching property resembles the property of a real caregiver, i.e., the distribution of the caregiver’s vowels addressed to infants (triangle with solid line) is stretched compared to that of vowels addressed to other adults (triangle with dotted line).

This characteristic seemed to arise in the simulation when the caregiver underestimated the degree of expansion of infant prototypes, in other words, when this degree exceeded that of the anticipated prototypes, as we can see in the left box of Fig. 8. In such cases, the caregiver perceives the infant’s utterances as exaggerated ones and thus the caregiver’s utterances also become exaggerated through her imitations.

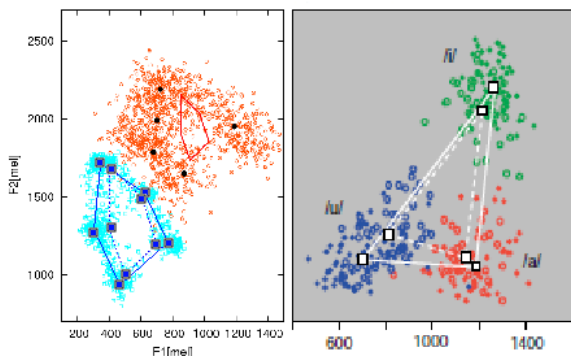


Figure 8: Stretching motherese in our simulation in the middle of the period under condition (c) (left box) compared to that reported by kuhl *et al.*, 1997 (right box).

7. Conclusion

According to the results obtained in this study, we suggest a more progressive picture of the develop-

mental process of sharing vowels through vocal imitation as follows:

1. An infant’s prototypes begin to separate from each other rapidly, since the separating effect caused by the infant’s articulation error exceeds the guiding effect caused by the caregiver’s biases. We can see this trend in the first half of the period in Figs. 5 and 6.
2. As the infant prototypes expand, he/she begins to utter vowels perceived by the caregiver as prototypes, and thus they come to utter more prototypical vowels as phonemes of their mother language. This trend can be seen in the first and middle parts of the period in Fig. 5.
3. The geometry of anticipated prototypes expands as the infant develops. Stretching motherese arises when the degree of expansion of the infant’s prototypes exceeds the degree of the caregiver’s anticipation, as shown in Fig. 8.
4. The accuracy of infant articulation improves along with the separation of his/her prototypes, and the aspect of guidance becomes dominant over the aspect of separation. This trend, where the infant’s prototypes are gradually guided toward the anticipated prototypes, can be seen in the last half of the period under conditions (c) and (d) in Fig. 7.

This picture of development includes two new hypotheses to be addressed: 1) It is the inaccuracy of the infant’s articulation that separates his/her prototypes, and 2) stretching motherese is a reaction of the caregiver’s underestimation of the infant’s prototypes expressed through the caregiver’s imitation.

We assume that the inaccuracy of infant articulation can be modeled as a Gaussian noise in articulation command space and that its degree is improved along with the separation of the infant’s prototypes. However, there seems several possible causes of this inaccuracy, such as the immature levels of three key factors: auditory-articulatory integration, articulatory muscles, and auditory perception. Some studies have addressed issues related to this type of development (Kanda *et al.*, 2008, Guenther and Perkell, 2004, Westermann and Miranda, 2004). Introducing their findings in our model would help us to improve it.

Motherese is one of the well-known characteristics of caregivers addressing infants (Kuhl *et al.*, 1997, Fernald and Simon, 1984), and many researchers have argued for its facilitating role in infant development (Kuhl *et al.*, 2008, Gogate *et al.*, 2006, Liu *et al.*, 2003, Masataka, 1993). However, there have been few explanations proposed for the mechanism behind the rise of motherese. Our results suggest that motherese comes from a caregiver’s unconscious anticipation in sharing vowels with her infant in an underestimating manner.

Acknowledgements

This work was supported in part by a Grant-in-Aid for JSPS Fellows.

References

- Asada, M., Hosoda, K., Kuniyoshi, Y., Ishiguro, H., Inui, T., Yoshikawa, Y., Ogino, M., and Yoshida, C. (2009). Cognitive developmental robotics: a survey. *IEEE Transactions on Autonomous Mental Development*, 1(1):12–34.
- Bloom, K. and Lo, E. (1990). Adult perceptions of vocalizing infant. *Infant Behavior and Development*, 13:209–219.
- de Boer, B. (2000). Self organization in vowel systems. *Journal of Phonetics*, 28(4):441–465.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *J. of the Royal Statistical Society. Series B(Methodological)*, 39:1–38.
- Fernald, A. and Simon, T. (1984). Expanded intonation contours in mothers' speech to newborns. *Developmental Psychology*, 20(1):104–113.
- Gogate, L. J., Bolzani, L. H., and Betancourt, E. A. (2006). Attention to maternal multimodal naming by 6- to 8-month-old infant and learning of word-object relations. *Infancy*, 9(3):259–288.
- Gros-Louis, J., West, M. J., Goldstein, M. H., and King, A. P. (2006). Mothers provide differential feedback to infants' prelinguistic sounds. *Int. J. of Behavioral Development*, 30:509–516.
- Guenther, F. H. and Perkell, J. S. (2004). A neural model of speech production and its application to studies of the role of auditory feedback in speech. In *Speech Motor Control in Normal and Disordered Speech*, pages 29–49.
- Ishihara, H., Yoshikawa, Y., Miura, K., and Asada, M. (2008). Caregiver's sensorimotor magnets lead infant's vowel acquisition through auto mirroring. In *Proc. of the IEEE 7th Int. Conf. on Development and Learning, CD-ROM*.
- Ishizuka, K., Mugitani, R., Kato, H., and Amano, S. (2007). Longitudinal developmental changes in spectral peaks of vowels produced by japanese infant. *Acoustical Society of America*, 121(4):2272–2282.
- Kanda, H., Ogata, T., Komatani, K., and Okuno, H. G. (2008). Segmenting acoustic signal with articulatory movement using recurrent neural network for phoneme acquisition. *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1712–1717.
- Kokkinaki, T. and Kugiumutzakis, G. (2000). Basic aspects of vocal imitation in infant-parent interaction during the first 6 months. *J. of Reproductive and Infant Psychology*, 18(3):173–187.
- Kuhl, P. K. (1991). Human adults and human infants show a perceptual magnet effect for the prototypes of speech categories, monkeys do not. *Perception & Psychophysics*, 50:93–107.
- Kuhl, P. K., Andruski, J. E., Chistovich, I. A., Chistovich, L. A., Kozhenikova, E. V., Ryskina, V. L., Stolyarova, E. I., Sundberg, U., and Lacerda, F. (1997). Cross-language analysis of phonetic units in language addressed to infant. *Science*, 277:684–686.
- Kuhl, P. K., Conboy, B. T., Cofey-Corina, S., Padgen, D., Rivera-Gaxiola, M., and Nelson, T. (2008). Learning as a pathway to language: new data and native language magnet theory expanded. *Philosophical Transactions of the Royal Society B*, 363:979–1000.
- Kuhl, P. K. and Meltzoff, A. N. (1996). Infant vocalizations in response to speech: Vocal imitation and developmental change. *J. of Acoustic Society of America*, 100:2415–2438.
- Liu, H.-M., Kuhl, P. K., and Tsao, F.-M. (2003). An association between mothers' speech clarity and infant' speech discrimination skills. *Developmental Science*, 6:3:F1–F10.
- Masataka, N. (1993). Effects of contingent and noncontingent maternal stimulation on the vocal behaviour of three- to four-month-old Japanese infants. *J. of Child Language*, 20:303–312.
- Masur, E. F. and Olson, J. (2008). Mothers' and infant' responses to their partners' spontaneous action and vocal/verbal imitation. *Infant Behavior and Development*, 31:704–715.
- McMurray, B., Aslin, R. N., and Tascano, J. C. (2009). Computational principles of language acquisition. *Developmental Science*, 12(3):369–378.
- Miura, K., Asada, M., and Yoshikawa, Y. (2007). Unconscious anchoring in maternal imitation that helps finding the correspondence of caregiver's vowel categories. *Advanced Robotics*, 21:1583–1600.
- Moody, J. and Darken, C. (1989). Fast learning in networks of locally-tuned processing units. *Neural Computation*, 1(2):281–294.
- Oller, D. K. and Eilers, R. E. (1988). The role of audition in infant babbling. *Child Development*, 59:441–449.
- Oudeyer, P.-Y. (2005). The self-organization of speech sounds. *J. of Theoretical Biology*, 233(3):435–449.
- Rvachew, S., Mattock, K., and Polka, L. (2006). Developmental and cross-linguistic variation in the infant vowel space: The case of Canadian English and Canadian French. *J. of Acoustical Society of America*, 120(4):2250–2259.
- Sato, M. and Ishii, S. (2000). On-line EM algorithm for the normalized gaussian network. *Neural Computation*, 12:407–432.
- Vallabha, G. K., McClelland, J. L., Pons, F., Werker, J. F., and Amano, S. (2007). Unsupervised learning of vowel categories from infant-directed speech. *Proc. of National Academy of Sciences USA*, 104(33):13273–13278.
- van Beinum, F. J. K., Clement, C. J., and van den Dikkenberg-Pot, I. (2001). Babbling and the lack of auditory speech perception: a matter of coordination? *Developmental Science*, 4:1:61–70.
- Vihman, M. M. and Nakai, S. (2003). Experimental evidence for an effect of vocal experience on infant speech perception. *Proc. of the 15th Int. Cong. of Phonetic Science*, pages 1017–1020.
- Vorperian, H. K. and Kent, R. D. (2007). Vowel acoustic space development in children: A synthesis of acoustic and anatomic data. *J. of Speech, Language, and Hearing Research*, 50:1510–1545.
- Westermann, G. and Miranda, E. R. (2004). A new model of sensorimotor coupling in the development of speech. *Brain and Language*, 89(2):393–400.

Self-Regulation Mechanism for Continual Autonomous Learning in Open-Ended Environments

Kenta Kawamoto Yukiko Hoshino Kuniaki Noda Kohtaro Sabe
System Technologies Laboratories, Sony Corporation
{Kenta.Kawamoto, Yukikoh, Kuniaki.Noda, Kohtaro.Sabe}@jp.sony.com

Abstract

Continual and autonomous learning are key features for a developmental agent in open-ended environments. This paper presents a mechanism of self-regulated learning to realize them. Considering the fact that learning progresses only when the learner is exposed to appropriate level of uncertainty, we propose that an agent's learning process be guided by the following two metacognitive strategies throughout its development: (a) Switch of behavioral strategies to regulate the level of expected uncertainty, and (b) Switch of learning strategies in accordance with the current subjective uncertainty. With this mechanism, we demonstrate efficient and stable online learning of a maze where only local perception is allowed: the agent autonomously explores an environment of significant-scale, and builds a model that describes the hidden structure perfectly.

1. Introduction

Imagine the life with a home companion robot having empathetic interaction with you and your family, sharing the same life space together. (Fujita, 2009) discussed issues for realizing such an intelligent robot and proposed a novel approach called Intelligence Dynamics. Among the issues discussed, ability of continual autonomous learning is crucial particularly for a self-developmental agent that deals with open-ended environments with many perceptual aliases. With that in mind, a mechanism for accelerating the learning process online is definitely required.

Besides studies on learning algorithms per se, learning acceleration is mostly studied in the area of active data sampling. In this domain, data to learn is not given: it is the learner itself that decides its behaviors and consequently the experience to be learned. Choice of experience has decisive influence not only on learning results but also on learning efficiency. Hence, strategy for exploring to gain proper experience (action and observation sequences) in the proper order is one topic of considerable significance to an autonomous learning agent.

Reinforcement learning is a major approach trying to find the best policy for maximizing reward. It is quite unique in that its exploration is focused only in areas where maximum reward is expected. This simplification contributes much to efficient learning but at the expense of abandoning reward-free complete model acquisition. (Şimşek and Barto, 2006) proposed a formulation of a reinforcement learning agent that focuses 'Optimal Exploration' by introducing intrinsic reward, but still it is not reward-free: extrinsic reward must be defined in advance of the exploration. Extended formulation in Partially Observable Markov Decision Process (POMDP) case needs more investigation, too.

The adoption of intrinsic motivations as a useful behavioral strategy is a common approach to exploration and efficient learning. (Sabe et al., 2006) proposed an open-ended system that autonomously develops by setting itself appropriate tasks using a motivational mechanism inspired by the Flow Theory (Csikszentmihalyi, 1990). Another self-developmental system with the idea of 'Intelligent Adaptive Curiosity' is proposed and the demonstrated results (Oudeyer et al., 2005) are quite attractive, though the experimental situations are rather small and a bit too ideal. In reality, curiosity driven behavioral strategies may be of little avail after the agent gets lost as a result of its curiosity, which fact is less-considered in most literatures. Learning efficiency falls off dramatically in such situations, and a recovering strategy becomes more important for continual autonomous learning.

We insist, therefore, that the behavioral strategy must be changed depending on the learner's status of understanding the world, in order to achieve the maximum efficiency in lifelong learning.

Yet another point of importance to argue is a timing regulation of the entire model update. A timing-predefined additional learning should be avoided, because forced learning in a completely incomprehensible situation may result in a breaking of the already acquired structures. Consequently, learning strategy also must be changed depending on a learner's subjective uncertainty in understanding the world.

Above discussion naturally leads to insights on

Table 1: Outline of our self-regulation mechanism. (a), (b), (c), and (d) represents different research areas. Quoted terms are the names of our strategy, described in sections 2.3 and 2.4. Meta-strategies for regulating them are not explicitly shown, but explained in the text.

	Metacognitive awareness	
	Known	Unknown
Behavioral strategy	(a) ‘Exploration’	(b) ‘Identification’
Learning strategy	(c) ‘Global update’	(d) ‘Local update’

the importance of metacognition in learning and on the advantages of appropriately regulated behavioral and learning strategies. This paper presents a mechanism of such metacognition and regulations with experimental demonstrations of its effectiveness.

Section 2 provides an overview of the self-regulated learning mechanism. Section 3 introduces the formulation of POMDP and an enhanced version of Hidden Markov Models (HMM) as a learning model. Then the implementation example of the proposed mechanism is described in this POMDP + HMM case. Experimental results, supporting our approach, are reported in section 4, including the demonstration of efficient and stable learning of a large HMM (the number of parameters to be learned is over a million). Section 5 discusses comparison with related works, and some limitations. Finally, conclusions and future works are presented in section 6.

2. Self-regulation mechanism for an autonomous learning agent

In educational psychology, it is commonly shared that changing one’s strategy for learning in accordance with the judgment on one’s knowing can greatly improve learning. This kind of learning process, guided by metacognition, is called Self-regulated Learning (Zimmerman, 1990), and shows a marked similarity to our approach.

The essence of self-regulated learning is metacognitive awareness and metacognitive regulation. The former is often described as “knowing about one’s knowing”, and corresponds to the process, in our approach, that judges whether current situation is well understood or not. For the latter mechanism, we have two meta-strategies: one for regulating behavioral strategy, and the other for regulating learning strategy. A brief outline of our self-regulation mechanism is shown in Table 1.

Researches on behavioral strategy in ‘known’ situations are common, particularly in domains related to reinforcement learning and intrinsic motivations. This research area is shown by (a) in Table 1. (The term ‘known’ here means the current position in state

space is known.) Researches in other areas are not well studied, especially in areas (c) and (d).

In the following discussions, we will use Z to indicate internal state variable of the learning model which variable uniquely describes a situation of the learner and surrounding environments. π_t is a prior probability distribution over Z at time t and z_t is an instance of Z . \mathbf{o}_t and \mathbf{u}_t are used for observations and actions at time t respectively, and their time sequence is sometimes abbreviated as \mathcal{O} and \mathcal{U} . Parameters of the learning model is denoted by λ .

2.1 Internal state estimation with variable-length recognition window approach

Internal state estimation for time t is defined as a process evaluating $P(Z_t|\pi_t, \mathbf{o}_t, \lambda)$ where the prior π_t is estimated using the past sequence of actions and observations: $\mathcal{U} \equiv \{\dots, \mathbf{u}_{t-1}\}$, $\mathcal{O} \equiv \{\dots, \mathbf{o}_{t-1}\}$. (An abbreviation $P(Z_t)$ may be used for convenience.)

In order to avoid difficulties caused by the fixed length of recognition window, i.e. the length of data sequence \mathcal{U} and \mathcal{O} , we employed a variable-length recognition window approach:

1. Start with the window length $w = 0$ and π_t as a uniform distribution.
2. Estimate the current internal state $P(Z_t|\pi_t, \mathbf{o}_t, \lambda)$ and calculate its entropy: $H_w(P(Z_t))$.
3. If the window length w is long enough and the sequence $\{H_0, \dots, H_w\}$ has converged, quit with the last estimation (of the longest window).
Abort, if the window length w gets too long.
Otherwise continue with $w = w + 1$.
4. Estimate the current prior $P(\pi_t|\pi_{t-w}, \mathcal{U}_w, \mathcal{O}_w, \lambda)$ with π_{t-w} as a uniform distribution and $\mathcal{U}_w \equiv \{\mathbf{u}_{t-w}, \dots, \mathbf{u}_{t-1}\}$, $\mathcal{O}_w \equiv \{\mathbf{o}_{t-w}, \dots, \mathbf{o}_{t-1}\}$.
Then go back to step 2.

Estimated result is a sequence of probability distribution over Z for each time step in $[t-w, t]$. Model dependent algorithms will be applied, to extract the most likely state transition sequence $\{z_{t-w}, \dots, z_t\}$ from the result.

2.2 Metacognitive awareness of ‘known’ and ‘unknown’

Metacognition of whether current situation is well known or unknown is the first step of self-regulation. It can be judged in the following way:

1. Estimate the time sequence of Z up to current time t : $\{\dots, z_{t-1}, z_t\}$ using the method described in subsection 2.1.
If the estimation process aborted, the current situation is clearly ‘unknown’.
2. Evaluate the entropy of the probability distribution of the current internal state $H(P(Z_t))$ and compare it with a threshold θ_H .

In case $H(P(Z_t)) > \theta_H$, then the current state is too ambiguous and thus ‘unknown’.

3. Evaluate the occurrence probabilities of all the observations and actions in the estimated time sequence of most likely Z . If there exists a time t s.t. $P(\mathbf{o}_t|z_t) < \theta_{P_o}$ or $P(\mathbf{u}_t|z_{t-1}, z_t) < \theta_{P_u}$ for appropriate thresholds θ_{P_o} and θ_{P_u} , then the current model will be judged as insufficient for explaining the recent experience. The current state is, of course, ‘unknown’.
4. Otherwise the state is ‘known’.

The advantage of metacognition is to provide a learner with a choice to continue with current situation kept as ‘unknown’: the most likely state is not necessarily the best estimation of the current state. Hence required are both behavioral and learning strategies that can work in ‘unknown’ situations.

2.3 Metacognitive regulation of behavior: Exploration and Identification

Learning only progresses when the learner is exposed to appropriate level of uncertainty. In situations full of uncertainty, learning is too difficult, while nothing is left to learn in situations without uncertainty. An autonomous learner therefore tries to regulate the level of uncertainty in order to achieve the maximum learning efficiency in the given environment.

In this subsection, we present a mechanism for this purpose, i.e. meta-strategy for switching the following two behavioral strategies: uncertainty increasing strategy and uncertainty reducing strategy, namely ‘exploration’ and ‘identification’, respectively.

Exploration – Uncertainty increasing strategy:

This strategy is to take a sequence of actions $\mathcal{U} \equiv \{\mathbf{u}_t, \dots\}$ to make the entropy $H(P(Z))$ maximum at some time in the future, given the current prior probability distribution π_t and observation \mathbf{o}_t : $\operatorname{argmax}_{\mathcal{U}} \sum_{\mathcal{O}} P(\mathcal{O}|\pi_t, \mathbf{o}_t, \mathcal{U}, \lambda) H(P(Z|\pi_t, \mathbf{o}_t, \mathcal{O}, \mathcal{U}, \lambda))$, where \mathcal{O} is a set of possible sequences of future observations.

What this strategy tries to maximize is an expectation of entropy of a posterior probability distribution over Z , given possible future sequence \mathcal{U} and \mathcal{O} . This strategy leads to such behaviors exploring a never tried transition rather than a randomly chosen transition. Note the difference with strategies such as $\operatorname{argmax}_{\mathcal{U}} H(P(Z|\pi_t, \mathbf{o}_t, \mathcal{U}, \lambda))$.

Identification – Uncertainty reducing strategy: $\operatorname{argmin}_{\mathcal{U}} \sum_{\mathcal{O}} P(\mathcal{O}|\pi_t, \mathbf{o}_t, \mathcal{U}, \lambda) H(P(Z|\pi_t, \mathbf{o}_t, \mathcal{O}, \mathcal{U}, \lambda))$.

This is just the opposite of the explorative strategy and tries to take such actions $\{\mathbf{u}_t, \dots\}$ that best reduce the ambiguity in internal variable Z . But in those cases where environments are fraught with uncertainties, calculation of the prior π_t may be

troublesome, because normal state estimation will not be reliable at all. Consequently, a variant of state estimation mechanism with variable-length recognition window (described in section 2.1) is used with modification that the occurrence evaluation procedure (step 3. in section 2.2) be in place instead of the original entropy convergence evaluation procedure (step 3. in section 2.1). Thus, π_t can be estimated using the longest segment of the past sequence that is consistent with the model λ .

With this strategy, the agent selects the most informative action with a hope of finding itself back in the known region. At least, it can avoid meaningless behaviors even in the unknown region.

Meta-strategy for switching Exploration and Identification:

The idea is just to switch strategies depending on the judgment of current situation: ‘exploration’ strategy in ‘known’ situation and ‘identification’, strategy in ‘unknown’ situation. This meta-strategy tries to increase uncertainty if the situation is known with less uncertainty, and to reduce it reversely in case of unknown situation which is full of uncertainty. As a result, the level of uncertainty is adequately regulated, and the agent’s learning efficiency is improved.

2.4 Metacognitive regulation of learning: Global update and Local update

One important fact we have found is that it is destructive to do normal incremental learning when the agent is in too much uncertainties. Two types of learning strategy are prepared and used tactfully to settle this problem.

Global update: This is just a normal incremental learning of the model employed. Model parameter λ is globally updated, adapting to the recently experienced action and observation sequences in balance with the past. In case the new experience can be grounded solidly on the model, this learning process is useful for refining the learning result.

Local update: This learning is rather improvising. The model is expanded locally with a flavor of one-shot learning, and most of the model parameters are left untouched. Relatively local and isolated representation describing the recent experience is generated and loosely associated to the rest of the model.

Meta-strategy for switching Global update and Local update: In familiar situations, where experienced action and observation sequences are well-grounded to the model, the agent continuously updates the model (Global update strategy). If the situation becomes ‘unknown’, it immediately stops global update and switches to another strategy: local update is carried out instead, and at the same time, all the sequences of the unfamiliar experience are stored for later use in global update. After the

situation gets familiar again, the stored sequences will become interpretable in the light of locally updated model. Then the agent can globally update the model using all the experience of the wandering: entering the unknown from the known, and finally coming back to the known again.

3. Implementation example of the self-regulation mechanism

The above formulation is non-specific and hence is applicable to many tasks, environments, and learning models. Here we apply it to an autonomous learning agent in POMDP environments and present its implementation details.

In this paper, a POMDP is described by the tuple (Z, π, A, B, V, W) , where Z is a set of hidden states, π is a prior probability distribution over these states, W is a set of actions, and A is a transition function that maps $Z \times W$ into discrete probability distributions over Z . The emission function B maps Z into discrete probability distributions over V . Thus, observations from V can be perceived, while the states in Z are not directly observable. We take it for granted that the size of Z may increase with time and experience in open-ended environments.

3.1 Enhanced HMM for modeling POMDP

We enhanced HMMs to serve as predictive models in POMDP environments. The enhancement itself has been found to be quite similar to what (Chrisman, 1992) did. Specifically, we use a discrete ergodic (fully-connected) HMM with a simple action representation extension: the 2-D transition probability matrix of size $Z \times Z$ is replaced by a 3-D matrix of size $Z \times Z \times W$ which can be viewed as a set of 2-D transition matrices: each matrix element represents the effect of each action $w \in W$.

When the environmental model of this type is fully acquired, the agent can execute any tasks reward-freely using the dynamic programming technique.

In order to estimate the parameters of the model, a slightly modified version of Baum-Welch algorithm is used: transition probability a_{ij} is simply replaced by a corresponding conditional probability $a_{ij}(u_t)$, where u_t is a discrete action at time t .

3.2 Split-and-merge technique for a stable convergence in HMM learning

Parameter estimation of an ergodic HMM can be easily trapped in a local minimum, because its full-connectedness provides too much amount of freedom for modeling the world. We, therefore, introduced two types of constraints in the learning process.

The first one, namely single-observation-per-state, is as follows: a hidden state which has high prob-

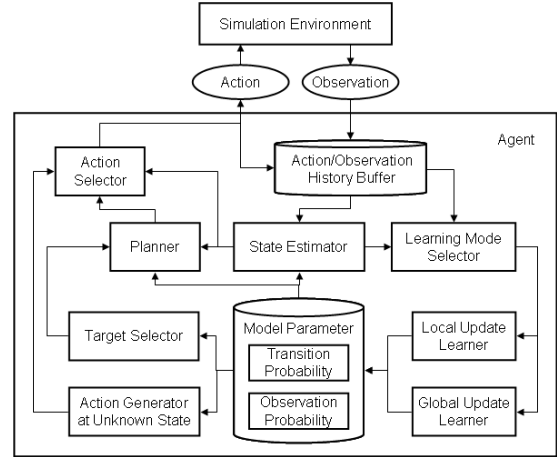


Figure 1: Block diagram of a self-developmental agent.

abilities for two or more observation symbols will be forcedly split into multiple states, one for each of the observation symbols. The second one is for reducing the redundant states: if there exist some states that have in common (a) the same transition source or destination state for the same action, and (b) the same observation symbol mainly associated, then they will be merged into a single state.

These regulations are processed every time after the convergence of the Baum-Welch iteration and in case there exist any nodes newly split or merged, the learning process will go back to the beginning of the Baum-Welch step.

3.3 Modeling a self-developmental agent

Figure 1 shows the entire block diagram of our model of a self-developmental agent. Utilizing these functional modules in coordinated manner, the agent autonomously explores the POMDP environment and incrementally acquires the model of its hidden states.

The interface between the environment and the agent is abstracted as a sequence of actions and observations. At every time step, the agent outputs an action symbol towards the environment and receives an observation symbol as a result of its action. The agent has no prior knowledge: neither on the relations among action/observation symbols nor on the relations between observations and action results.

3.4 Metacognitive awareness in case of the enhanced HMM

At every execution step, the state estimator module has to recognize whether the agent is in the ‘known’ situation or in the ‘unknown’ situation. The procedure described in sections 2.1 and 2.2 are implemented for the enhanced HMM in the following way. In order to estimate the most likely state transition sequence for the history of actions and observations,

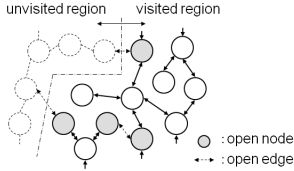


Figure 2: Definitions of ‘open edge’ and ‘open node’.

Viterbi algorithm is used with the same simple modification described in section 3.1: transition probability a_{ij} being replaced by $a_{ij}(u_t)$. The occurrence probabilities referred in step 3. in section 2.2 are directly represented in the parameters of the HMM: $P(\mathbf{o}_t|z_t)$ and $P(\mathbf{u}_t|z_{t-1}, z_t)$ can be found in the observation and transition probability parameters B and A respectively.

3.5 Open-spot exploration

When the agent is confident of its current state, it tries to increase the expected uncertainty of the internal states as described in section 2.3. We implement this behavior as a three-step-procedure: (1) find less-explored states, (2) plan the path to one of them, and (3) go there, trying one of the less-experienced transitions.

Figure 2 shows a basic idea to find less-explored states. Each node represents a hidden state of the HMM, and the arcs connecting the nodes are major transitions between the states. Dotted circles are states in the complete model which have not yet structured in the currently acquired model. Dotted arcs, namely ‘open edge’, represent undiscovered transitions due to the lack of experience. A node which have one or more open edges is called an ‘open node’ and the open node with the attached open edge(s) are collectively referred to as an ‘open-spot’.

If the transition probability from a state z by an action \mathbf{u} makes a widespread distribution, the pair of (z, \mathbf{u}) specifies an open-spot. Hence open-spots can be easily found by investigating the transition probability matrix of the learned HMM. Once an open-spot has been found, the action sequence to reach the corresponding open node can be planned using the dynamic programming method, and the action to explore the open edge from there is obviously \mathbf{u} .

With this behavioral strategy, i.e. ‘open-spot exploration’, the agent can efficiently move around in the environment, aiming to explore not-yet-discovered states and transitions and to gain useful information for the model learning.

3.6 Identification behavior

When the agent fails to determine its current state and gets lost, the behavioral strategy switches to the uncertainty reducing one.

In our implementation, the prior π_t is estimated first, as described in section 2.3 and breadth-first search is employed to find the minimum expectation of the entropy. In the search process, (a) unlikely observation branches are pruned early, and (b) the search depth is fixed at the first finding of a node with acceptably low entropy.

This simple mechanism sufficiently works: giving a shortest plan for identification in a known region, and avoiding meaningless behaviors (staying in the same place or just going back and forth, for example) even in the unknown region.

3.7 Learning strategies and their regulation

For the global update of an HMM, we employed a method referred to as “Ensemble Training for HMM within an Incremental Learning setting” (Cavalin et al., 2008) with the split-and-merge technique explained in section 3.2.

The local update is implemented as follows. (1) Every time step in unknown situation, a new state associated with the current observation is added in the HMM and linked from the previous state by the last action executed. (2) By recognizing a 1-step sequence $(\mathbf{u}_t, \mathbf{o}_t)$, analogous states which exhibit a high probability for the sequence are gathered from the known region. (3) The parameters related to the newly-added state are initialized using the common properties extracted from those analogous states.

Once the agent successfully recognizes that the situation has become well known again, those states added during the wandering in unknown situation are reinterpreted as a whole sequence. This process integrates the global model and the locally updated model with some adjustments of added states and transitions. Then global update process is executed to consolidate the whole sequence experienced during the unknown period and to refine the model.

4. Experimental evaluation of self-regulation mechanism

4.1 Experimental settings

In order to evaluate the effectiveness of our proposed mechanism, we arrange a maze-like environment shown in Fig. 3. In this environment, the agent observes one of 16 observation symbols (Fig. 3 (d)) depending on its position, which is the hidden factor to be structured. The agent can move one cell according to the selected action (Fig. 3 (c)), but in case it hits a wall, its position will not change.

Note that the agent has no prior knowledge, such as “go-right action cancels go-left”, “cannot move to the wall direction”, and etc. All the knowledge must be learned from the experience. Additionally, there are many confusing areas in the environment and

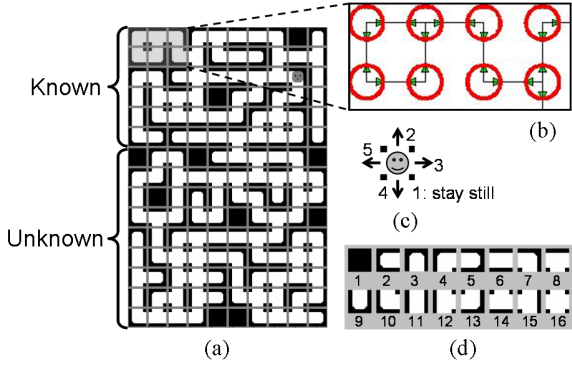


Figure 3: (a) Maze-like environment. (b) Example view of the acquired structure in the HMM. (Circle: internal state. Arrow: transition between internal states.) (c) 5 action symbols. (d) 16 observation symbols.

thus required is a high level of ability for finding the hidden structure behind observations.

At first, the agent moves around the region labeled as ‘known’ (Fig. 3 (a)) for 16000 time steps. Using this experience, the batch learning process of the HMM is carried out to build up an initial model of the known region in the environment.

After that, the agent starts exploring the rest of the world and continues self-developing. Parameters used in the metacognition process (section 2.2) are as follows: $[\theta_H = 1.0, \theta_{P_c} = 0.8, \theta_{P_u} = 0.1]$.

4.2 Efficiency of the open-spot exploration strategy and the identification strategy

In this subsection, we evaluate the effect of (a) the exploration strategy in known situation and (b) the identification strategy in unknown situation, from a point of view of learning efficiency. Learning efficiency is measured by a number of time steps elapsed since the agent first stepped into the ‘unknown’ region until a perfect model of the entire environment is acquired. Behavioral strategy for comparison is:

Random: move one cell randomly in the open direction. (True random action selection that allows wall-hitting is extremely inefficient and hence ignored.)

Forward: move one cell randomly in the open direction except for the backward direction which is chosen only when the agent faces a dead-end.

We executed the experiment 10 times for each of the strategies and calculate the mean and variance of the elapsed time. (Fig. 4) Small elapsed time is a proof of the contribution to learning efficiency, and small variance indicates a stable performance.

In the first experiment (Fig. 4 (a)), effects of the behavioral strategy in known situation are compared: One agent takes open-spot exploration strategy in known situation (‘Exploration’), while another takes ‘Forward’ strategy and the last takes ‘Random’ strategy. All agents take ‘Identification’ strategy in un-

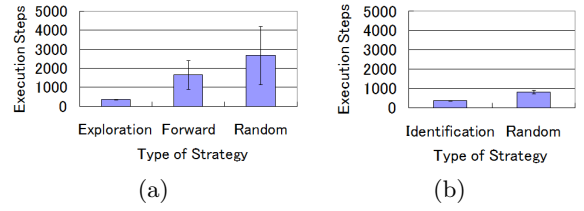


Figure 4: (a) Efficiency evaluation of open-spot exploration in known region. (b) Efficiency evaluation of identification strategy in unknown region.

known situation. Results show that systematic exploration strategy is crucial for the efficiency of self-development.

In the second experiment (Fig. 4 (b)), effects of the strategy in unknown situation are compared: One takes ‘Identification’ strategy, while the other takes ‘Random’ strategy. Both agents take ‘Exploration’ strategy in known situation. Results suggest that ‘Identification’ strategy sufficiently works in unknown situation, though its formulation utilizes only the knowledge in known situation.

4.3 Effectiveness of regulating the learning strategy

In order to evaluate the effectiveness of the metacognitive regulation of the learning strategy, we compare the performance of following two agents: (a) an agent with the proposed mechanism and (b) an agent which runs the incremental learning procedure every time step despite its metacognitive awareness. Accurately speaking, when the agent is in unknown situation, it adds a new state in the HMM just the same way as described in section 3.7 paragraph 2, and then executes global update. (section 3.7 paragraph 1). The behavioral strategies are metacognitively regulated in both agents.

The experiment is repeated 10 times for each agent and the results are always similar. One typical case is shown in Fig. 5.

The agent with learning-regulation can build the complete model of unknown region within 400 steps in every trial. In this case, temporarily added new states that represent the sequence in yet-unknown region are anchored in both ends to the model of known region, and then all the model parameters are optimized for maximizing the likelihood of its learning sequence. There are cases some redundant states persist after the optimization, but the behavioral strategy naturally guides the agent to such defective areas again, and additional experience around there progressively eliminates such duplications.

In contrast, the other agent cannot complete the learning even after 1000 steps.(Fig. 5 (c)) It seems that the newly added states are not sufficiently constrained (connected only to the previous states) at

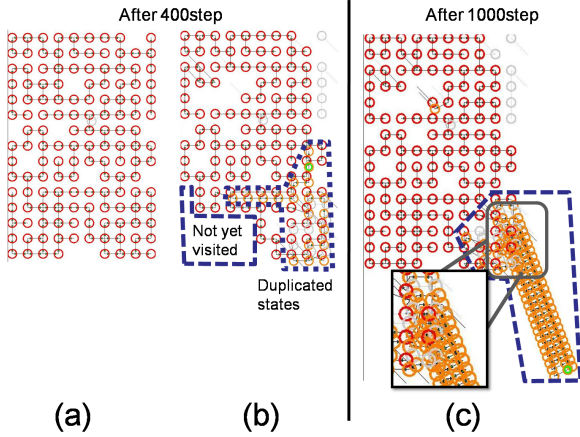


Figure 5: Effects of the learning strategy regulation. (a) Result of the proposed method: Hidden states are perfectly structured. (b) Result of a periodic incremental learning strategy. Many redundant states (drawn in the same position with a slight shift) show the agent’s uncertainty in understanding the environment. The agent was busy investigating the uncertain areas and hence unvisited area is left behind. (c) The uncertainty cannot be resolved even after 1000 steps.

the timing of global optimization, and thus the optimization process of the HMM is easily trapped in local maximum. Even if the agent visits the same place again, those temporarily added states disturb the right recognition, and thereby the agent tends to consider the place as a new state. Figure 5 (c) is a result of such negative spiral.

Those results show that the self-regulation of learning strategy plays a crucial role for the ability and efficiency of learning a new environment.

4.4 Efficiency and stability of learning in a large-scale environment

For comparison with our self-regulated incremental approach, commonly-used batch-learning approach is tested in the environment shown in Fig. 6 (a). 20000-step action and observation sequence, collected by wandering all over the maze, is used for learning. We tried 10 times with different data sets, but none of the learnings was successful (Fig. 6 (b)), revealing the non-scalability of the approach.

Then we tested our approach in a larger environment shown in Fig. 6 (c). The experimental setting is the same as previous sections except for the scale of the world. The agent autonomously explores and incrementally learns the environment under the regulation of the proposed mechanism. No confusion is found in the final result after 3723 steps. (Fig. 6 (d)) HMM with 587 internal states is obtained with its over a million ($587 \times (587 \times 5 + 16)$) parameters properly set, which is intractable by a batch-learning approach. We executed the experiment several times

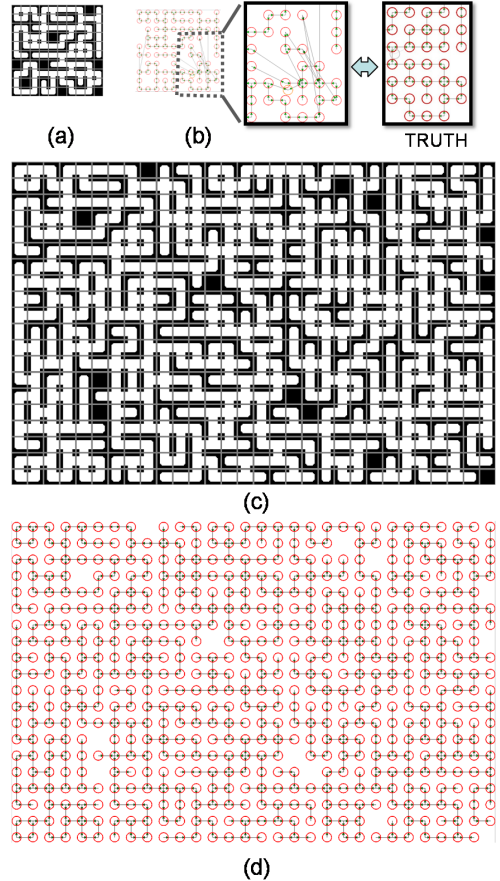


Figure 6: HMM learning results of significant-scale environments. (a) 12×10 world used for batch learning. (b) Result of batch learning using 20000-step sequence: confusions of similar areas are not cleared up. (c) 20×30 world used for evaluation of our approach. (d) Learning result of our approach: all of many confusing areas are perfectly distinguished in less than 4000 steps. The agent can move to anywhere using the shortest path.

and the results were always perfect, which demonstrates the efficiency and stability of our proposed mechanism even in a large scale environment.

5. Discussion

Firstly, we discuss the scalability of our mechanism. Even in recent reinforcement learning researches, the numbers of the hidden states of the learned environments are rather small, up to 80 at most (Info et al., 2004), and the environmental configurations are also simple: just an E-shape maze, for example. In researches that model large-scale environments, such as the one with Hierarchical POMDP (Theocharous et al., 2004), the structures of the internal states are not self-organized from the learning data, but are manually built. We thus say that one of our major contributions is achieving a stable learning of a large-scale POMDP environment.

Secondly, we discuss the generality of the mecha-

nism. In our formulation of self-regulated learning, nothing is assumed for the action and observation symbols: their representations can be anything, and they don't need to have any topological or metric relations. In addition, only the internal parameters of the HMM are used for the agent's decision making and the values of input/output data have no direct effect on its strategy. Our formulation, therefore, is essentially task independent and can be applied to many tasks, that can be modeled with a sequence of action and observation symbols: Dynamics of a pendulum, arm manipulations, an interaction model between an artificial agent and a human, and etc.

Another point of discussion is continuous data handling. Evaluations are limited only for discrete symbols, because the implementation of the learning model is based on discrete HMM. In case of continuous observation, extension is straightforward: using a continuous HMM with unimodal Gaussian observation model will be enough. But extension to continuous action is different. This is a limitation and researches are needed for overcoming it.

The last topic is about noise robustness. In our experiments, both observation and action process have no noise. We examined the influence of noises by carrying out additional experiments with noisy observations. When the agent gets a contaminated observation, the estimation changes to 'unknown' and the agent adds a new internal state with the wrong observation associated. If the noise ratio is not so high, such as 1%, learning and planning are not so much influenced. Learning mechanisms, i.e. open-spot exploration combined with HMM incremental learning and split-and-merge technique, successfully work to reduce the unnecessary redundancy in the learned model. But in case the noise rate is around 10%, the internal states created by wrong observations increase too quickly before the agent identify the current state correctly, and many redundant states are left unresolved. This is a current disadvantage of our model, and improvements are required in the way of modeling and handling noisy data.

6. Conclusion and future works

We propose a self-regulation mechanism for realizing continual and autonomous learning in open-ended environments. The key points are metacognitive awareness and regulation: both behavioral and learning strategies are regulated in accordance with the agent's subjective uncertainty in understanding the current situation. The effectiveness of our approach is shown with several experiments. Among others, the agent autonomously explores and learns a significantly large POMDP environment efficiently.

Future works are to apply this mechanism (a) to tasks in dynamically reconfigurable environments, (b) to agents with continuous action and observa-

tion signals, and (c) to environments ruled by different type of dynamics. Extensions of the internal model formulation are necessary, but we expect that the self-regulation mechanism can be used as is.

References

- Cavalin, P. R., Sabourin, R., Suen, C. Y., and Jr., A. S. B. (2008). Evaluation of incremental learning algorithms for an HMM-based handwritten isolated digits recognizer. In *Proceedings of The 11th International Conference on Frontiers in Handwriting Recognition*, pages 1–6.
- Chrisman, L. (1992). Reinforcement learning with perceptual aliasing: The perceptual distinctions approach. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 183–188. AAAI Press.
- Csikszentmihalyi, M. (1990). *Flow: The psychology of optimal experience*. New York: Harper and Row.
- Fujita, M. (2009). Intelligence Dynamics: a concept and preliminary experiments for open-ended learning agents. *Autonomous Agents and Multi-Agent Systems*.
- Info, D. W., Wierstra, D., and Wiering, M. (2004). Utile distinction hidden Markov models. In *Proceedings of International Conference on Machine Learning*.
- Oudeyer, P.-Y., Kaplan, F., Hafner, V. V., and Whyte, A. (2005). The playground experiment: Task-independent development of a curious robot. In *Proceedings of the AAAI Spring Symposium on Developmental Robotics*, pages 42–47.
- Sabe, K., Hidai, K., Kawamoto, K., and Suzuki, H. (2006). A proposal for intelligence model, MINDY for open ended learning system. In *Proceedings of the international workshop on intelligence dynamics at IEEE/RSJ Humanoids*.
- Şimşek, Ö. and Barto, A. G. (2006). An intrinsic reward mechanism for efficient exploration. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 841–848.
- Theocharous, G., Murphy, K., and Kaelbling, L. P. (2004). Representing hierarchical POMDPs as DBNs for multi-scale robot localization. In *Proceedings of IEEE Conference on Robotics and Automation*, pages 1045–1051.
- Zimmerman, B. J. (1990). Self-regulated learning and academic achievement: An overview. *Educational Psychologist*, 25:3–17.

Category-Based Intrinsic Motivation

Rachel Lee*
rlee1@swarthmore.edu

Ryan Walker*
rwalker1@cs.swarthmore.edu

Lisa Meeden*
meeden@cs.swarthmore.edu

James Marshall**
jmarshall@slc.edu

*Swarthmore College
Computer Science Department

**Sarah Lawrence College
Computer Science Department

Abstract

A goal of epigenetic robotics is to design a control architecture that implements an ongoing, autonomous developmental process which is unsupervised, unscheduled, and task-independent. The developmental process we are currently exploring contains three essential mechanisms: categorization, prediction, and intrinsic motivation. In this paper we describe a hybrid approach that uses Growing Neural Gas for categorization, neural networks for prediction, and Intelligent Adaptive Curiosity for intrinsic motivation. We apply this system to a physical robot operating in a dynamic visual environment and analyze the types of categories it forms.

1. Introduction

In a realistic environment, a robot is flooded with a constant stream of perceptual information. In order to use this information effectively for determining actions, a robot must have the ability to categorize its experience. Based on these categories, a robot must be able to predict how the environment will change as a result of its actions. Most importantly, this process of development should be driven by an intrinsic motivation to explore the categories of its experience where it can make the most learning progress. The developmental process we are currently exploring contains these three essential mechanisms: categorization, prediction, and intrinsic motivation.

Psychologists Ryan and Deci define intrinsic motivation as “the inherent tendency to seek out novelty and challenges, to extend and exercise one’s capacities, to explore, and to learn” (Ryan and Deci, 2000, p. 70). In order to determine what to explore, an organism must compare the incoming stimuli from the environment to its internal memory to discover differences and similarities. This collative process, a term coined by the psychologist Berlyne, is necessary to evaluate the degree of novelty or incongruity of the current stimuli with respect to the organism’s

past experiences or expectations (Berlyne, 1966). By categorizing its experience, an organism can more effectively decide which aspects of its environment are novel and should be explored. In a recent survey of developmental robotics, Lungarella et. al. state that categorization “is of such fundamental importance for cognition and intelligent behavior that a natural organism incapable of forming categories does not have much chance of survival” (Lungarella et al., 2003, p. 161).

Although intrinsic motivation and categorization are clearly intertwined, recent work in epigenetic robotics has focused primarily on intrinsic motivation alone (Barto et al., 2004, Marshall et al., 2004, Schmidhuber, 2006). One approach to intrinsic motivation, known as Intelligent Adaptive Curiosity (IAC), employs a limited form of categorization that divides the sensorimotor space into a set of similarity-based regions (Oudeyer et al., 2007). However, no abstractions of the sensorimotor data are formed, and this top-down approach may take a long time to create categories that accurately reflect the structure of the sensorimotor space. In addition, category formation is triggered by the number of exemplars and not by the uniqueness of the exemplars, which can lead to an excessive number of similar categories. Finally, IAC’s memory grows linearly with each additional experience.

In this paper we propose a hybrid system that combines IAC’s approach to intrinsic motivation with a mechanism known as Growing Neural Gas (GNG), which discovers relevant categories in sensorimotor data (Fritzke, 1995). GNG’s bottom-up approach to category formation quickly matches the structure of the sensorimotor space and only forms new categories when the existing ones are sufficiently different from the current data. We call this system Category-Based Intrinsic Motivation (CBIM). We apply CBIM to a physical robot operating in a dynamic visual environment and analyze the types of categories it forms. First, we summarize GNG, IAC, and introduce our hybrid system CBIM. Next, we describe the physical robot and the experiment. Fi-

nally, we present the results and discuss implications and future work.

1.1 Growing Neural Gas

GNG is an unsupervised learning method for dimensionality reduction (Fritzke, 1995). Given some high dimensional distribution of data, such as the sensorimotor data of a robot, a GNG will find a topological structure that closely matches the given distribution.

A GNG consists of a network of units and edges that are used to characterize the topological space in which its input vectors reside. Each unit contains a model vector that characterizes a portion of the overall distribution. Taken together, the units and edges of the GNG serve as a representative summary of the given distribution. The dimensionality of the network itself is not fixed in advance. The resulting graph is able to expand or contract as necessary by adding or deleting units and edges.

A given input vector is matched to the nearest and next-nearest GNG units based on Euclidean distance. This distance is also used as a measure of error, which the GNG stores in the nearest unit. All units connected to the nearest unit are moved toward the input vector by a fraction of the error. In this way, the GNG dynamically adapts to slight variations in the input signal that do not require the addition of new units.

Each edge in the GNG is assigned an age that is initially set to 0. If the nearest unit and the next-nearest units are not connected, an edge is placed between them; if they are connected, the age of the edge between them is reset to 0. Edges throughout the GNG above a given age threshold are pruned; if this results in isolated units, those units are also removed from the GNG.

A GNG begins with two units that are assigned random initial model vectors. In the original GNG (Fritzke, 1995), a new unit is added after a fixed number of time steps determined by the user. This unit’s model vector is placed between the unit with the greatest accumulated error and its neighbor with the greatest accumulated error. In an alternative implementation called an Equilibrium GNG (Provost et al., 2006), units are only added when the average error of the GNG’s units exceeds a given threshold. This approach makes it possible to grow the GNG in response to new data that doesn’t fit the current topology of the network, but prevents the addition of unnecessary units when the incoming data is similar to existing model vectors.

Because a GNG is able to autonomously grow and adapt over time, it is a suitable categorization mechanism for the open-ended learning system we propose in this work.

1.2 Intelligent Adaptive Curiosity

IAC is a method for implementing intrinsic motivation. IAC has been successfully tested on a Sony AIBO robot operating on a baby play mat with various toys that can be bitten, swatted, and observed (Oudeyer et al., 2007). Using IAC as its control mechanism, the AIBO clearly exhibited a developmental progression, first learning about simpler aspects and later focusing on more complex aspects of its environment.

The key idea of IAC is that the drive to learn is based on maximizing learning progress. This is achieved by creating a memory of all the experiences encountered by the robot and subdividing this memory into similarity-based regions. Each region contains an “expert” that is trying to learn to predict the effect of taking actions in particular sensory situations. More formally, the expert is trying to map the sensorimotor information at time t to the sensory outcome at time $t + 1$: $SM(t) \rightarrow S(t + 1)$. Each region monitors the errors of the expert over time and generates a measure of learning progress, which is essentially the change in the current mean error rate with respect to an earlier mean error rate.

On each time step the robot consults this memory in order to determine which action to take. First the robot senses the world. Next, it generates a set of candidate actions, either by enumerating all possibilities or, if the space of actions is continuous, by generating a random sample of possible actions. Then it concatenates each candidate action with the current sensory information and probes the memory to find all matching regions. With some high probability it selects the candidate action associated with the region with the maximal learning progress. Otherwise it chooses a random region from the matched set. It then executes the selected action, observes the outcome, and uses this data to train the expert associated with the selected region.

When a region’s sensorimotor context is predictable, initially its expert will make good progress and be chosen frequently. As the expert succeeds in learning, its progress will slow, and the learning progress of other regions will surpass it. In this way, IAC guides the robot to explore its environment in a sensible way, focusing on those aspects where it can make the best gains, and ignoring aspects that have already been learned or are unlearnable.

Although each region of IAC is in a limited sense a category, there is no abstraction taking place. Every experience the robot has had ($SM(t)$ paired with $S(t + 1)$) is explicitly stored within the appropriate region. Each region is limited to a fixed maximum size (usually 250 exemplars). Once this maximum is exceeded the region is split into two new sub-regions. If the robot continues to experience very similar situations it will repeatedly form additional regions, even

though they may not represent any significant differences from existing regions. This excess region formation will limit the effectiveness of IAC as it is applied to richer, more complex domains.

1.3 Category-based Intrinsic Motivation

CBIM is an open-ended learning system that combines GNG’s flexibility and power of abstraction with IAC’s notion of region-based maximal learning progress. We use the Equilibrium GNG, which only adds units based on accumulated error. Each IAC region is associated with one GNG unit. Each GNG unit model vector is determined by all of the sensorimotor exemplars that have been mapped to it. Each region stores a fixed number of exemplars, only enough to calculate learning progress; the oldest exemplars are removed as newer exemplars are encountered.

Unlike IAC, the growth of CBIM’s memory is bounded by the complexity of the robot’s sensory and motor capabilities as well as the environment because categories in CBIM are formed based on error and not simply on the quantity of experience. If the robot repeatedly experiences very similar situations, the associated GNG model vectors will adjust slightly to each experience. However, a new model vector will only be created when the error across the GNG grows too high. Then the new unit will be added at precisely the point in the GNG where the model vectors are least representative of the robot’s experiences. In this way, CBIM’s categories mirror the robot’s experiences, growing to handle new information or shrinking to remove spurious categories that are not consistent with later experiences.

In the original IAC model, the region experts were implemented as k -nearest neighbors (with $k=1$). In CBIM, the region experts are implemented as feed-forward neural networks with a single layer of weights. Every time a sensorimotor vector is mapped to a particular GNG unit, in the associated region the weights of the neural network expert are updated using standard backpropagation with $SM(t)$ as the input and $S(t+1)$ as the target. By using neural networks as the experts, CBIM incorporates another form of abstraction not found in IAC. Each neural network expert makes generalizations of the sensorimotor data in the process of learning to predict the outcomes of actions. These generalizations are likely to provide more robust behavior throughout the developmental process.

2. Experiments

In order to demonstrate the viability of CBIM, we designed a physical environment for a robot to do open-ended learning. For these experiments we used a Rovio, which is a consumer-level robot equipped

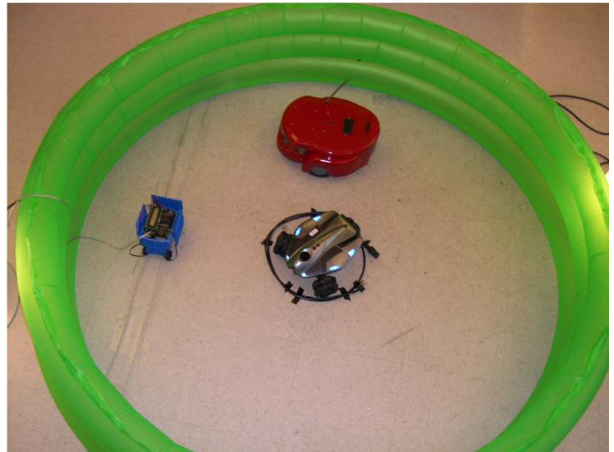


Figure 1: The experimental environment with the developing Rovio robot in the center, a larger static red robot, and a smaller moving blue robot.

with a camera. We wrote a Python interface using the open source Rovio API. The Python interface allowed us to control the Rovio through Pyrobot, a robotics control platform that implements a common API for both real and simulated robots (Blank et al., 2006). Image processing was also handled in Pyrobot.

The environment consisted of the Rovio in the center of a green inflatable pool as shown in Figure 1. This provided a uniform backdrop, limiting the robot’s vision to the arena, thus simplifying the visual stimulus. In addition to the green background, two robots were placed within the arena to serve as other objects of focus. A large, inactive red robot was placed to one side of the environment, but close enough to completely fill Rovio’s somewhat narrow field of vision when the Rovio looked directly at it. A smaller blue robot continuously moved back and forth on a track, using sensors to move from wall to wall. The Rovio was placed in the middle of this environment, capable only of rotating left or right. The Rovio could also choose not to move at all.

The developing Rovio robot experienced the world through vision, receiving sensory input extracted from its camera images. The Pyrobot vision system was configured to filter images from the Rovio’s camera to find the particular colors associated with each object in the environment: the green walls, the blue robot, and the red robot. However, due to variability in lighting conditions, the filters were not completely accurate. For example, a particular object might be present in the image, but its color might not be recognized by any of the filters. Each color channel was further filtered into a blob—a bounding box surrounding the largest mass of the respective color in the image. Figure 2 shows an image of the blue robot from the Rovio’s camera that has been



Figure 2: A camera image from the Rovio robot to which a blue blobify filter has been applied.

filtered for blue and then blobified.

The filter results were summarized in the sensory data as follows. Binary inputs indicated whether each color channel was active, meaning that an object of the specified color was present in the current image. Additionally, the robot received information about the color channel it *chose* to focus on. In other words, on each time step the robot could only attend to one of the color channels. This choice was part of its action decision. The area and relative position of the largest blob in this chosen channel were provided. The area was scaled to a value between 0 and 1, normalized by the size of the entire camera image. The relative position was represented as 0 for left, 0.5 for centered, and 1 for right. In summary, the Rovio had access to five sensory inputs:

$$S(t) = (red, green, blue, blobArea, blobPosition)$$

It was frequently the case in this environment that multiple color channels were active simultaneously. For instance, if the robot was facing the upper-left section of the environment (see Figure 1), it often had all three color channels active: it could see the red robot on its right, the blue robot on its left and the green background in between. Because the green background was almost always present in the camera image, the green channel tended to be active most of the time. Only when the Rovio was looking directly at the red robot, which tended to fill its entire camera image, would the green background not be seen. On rare occasions (about 1% of the time) none of the color channels were active due to the color variability caused by lighting conditions.

The Rovio had two output commands: which color channel to focus on and how much to rotate. The color channel choice was a value between 0 and 1 that was divided into three equal bins. For values in the range $[0.0, 0.33]$ the choice was red; for values in the range $[0.34, 0.66]$ the choice was green; for values in the range $[0.67, 1.0]$ the choice was blue. Rotation

was a value between 0 and 1 that was divided into seven equal bins, ranging from a hard left, to staying still, to hard right. Thus the Rovio’s motor action consisted of two values:

$$M(t) = (channelFocus, rotation)$$

For CBIM, this framework results in sensorimotor vectors of 7 dimensions. Therefore each GNG unit contains a 7-dimensional model vector. Each IAC region expert tries to predict the mapping from 7-dimensional sensorimotor vectors to 5-dimensional sensory vectors.

The goal of the experiment is for the robot to categorize its world, learning and making progress in predicting how the objects in this environment behave. Each object in the environment is brightly and evenly colored, so as to provide clear visual stimulus for the Rovio. Because each object in the environment has a unique color, each color channel can view only one object, eliminating possible confusion between objects.

In addition to its distinguishing color, each object offers unique learning opportunities with varying levels of predictability. The green walls offer a constant, large background making them quite predictable. For example, if on the current time step only the green channel is active and the Rovio chooses to focus on green and make a small turn to either the left or the right, it is highly likely that on the next time step the green channel will remain active and its area and relative position will be nearly identical to the previous time step. The red, static robot is also predictable, but is visible in only a few positions. For example, if the Rovio is directly facing the red robot with only the red channel active and chooses to focus on red and turn right, on the next time step it is likely that both the red and green channels will be active, and the red blob will be positioned to the left and be half the size it was previously. The blue robot is much harder to predict because it is constantly moving.

In this environment the robot must learn to predict the relative position and size of the objects in its visual field. What it will see at the next time step depends both on what it is currently seeing and what action it chooses to take. Over time, the developing robot should focus on all three objects, associating each object with a particular color channel.

Experiments lasted for 5000 time steps and took approximately 2 hours. We conducted 10 CBIM experiments using the Rovio. The IAC parameter settings were: 10 randomly generated candidate actions; 15% chance of selecting a random action; mean error rate was smoothed over 15 time steps; learning progress was calculated by comparing mean error rates separated by 10 time steps; and experts were feed-forward neural networks with a single layer of weights using a learning rate of 0.5 and no mo-

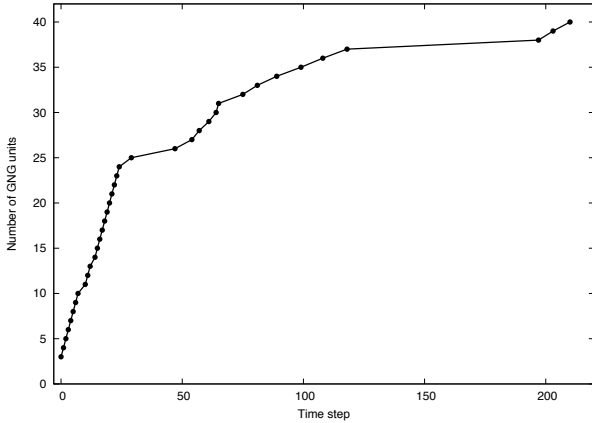


Figure 3: The number of units in the GNG of a typical CBIM run. In this run, after 210 time steps no additional units were inserted.

mentum. The Equilibrium GNG parameter settings were: error threshold of 0.5 to add a new unit; winning unit learning rate of 0.2; and neighbor unit learning rate of 0.006.

At the beginning of each experiment, the Rovio was positioned in an initial pose, facing away from the red and blue robots in the environment. This was done so as not to bias the early category formation toward either red or blue. If the developing robot initially decides to turn right, it will see the blue robot first and form categories to cover this situation. If instead, it initially decides to turn left, it will see the red robot first and form a different sequence of categories. The Rovio rapidly explored the world early in the experiment, forming many categories in the first few hundred time steps. The Rovio then attended to particular features of the environment. Common behaviors included investigating the bounds of the red robot, attempting to track the blue robot, and focusing on the area in which the blue robot occupied the same field of view as the red robot. Each experiment varied in the order that CBIM created categories and in the amount of time the developing robot spent focusing on each color channel. The next section analyzes the results in depth.

3. Results

Because categorization is of such fundamental importance to CBIM, we will first focus our analysis on how the GNG evolves over time. Then we will discuss the role that intrinsic motivation plays in the robot’s choice of actions in its vision-based environment.

3.1 GNG categories

Although results differ from run to run, there are clear GNG formation trends that directly correspond

to the robot encountering new sensory experiences in its environment. Recall that the GNG begins with two units which are assigned random model vectors. Once the robot senses the world for the first time, these initial units immediately accumulate enough error to trigger the formation of new units. At the beginning of the run, only the environment’s green background is within the robot’s camera view. Therefore, the first new GNG units are added to reflect that the green channel is active. As the robot begins to turn, it will either turn to the left and encounter the red robot or turn to the right and encounter the blue robot. As soon as one of these robots is in view, again the GNG immediately responds by constructing new units to represent that a new color channel has been activated for the first time. As the run continues, the robot will eventually see both the red and blue robot simultaneously. The first time this occurs, the GNG creates new units to represent this unfamiliar event.

Early on, much of the incoming sensorimotor data is novel, thus the bulk of new GNG units are added in the first 100 time steps and taper off rather quickly after that. Figure 3 shows how fast the GNG grows at the start of one particular run. In this case the GNG ceases to add more units by time step 210; in other runs, the last unit is added between time steps 250 and 400. This is a clear improvement on the linear growth of IAC regions.

Figure 7 shows a series of two-dimensional representations of the GNG model vectors and edges at different time steps during the run. To create these plots, a principal component analysis was performed on the final configuration of the 7-dimensional GNG model vectors from the last time step (step 5000). The projections of the model vectors onto the first two principal components were then plotted for each of the time steps shown, with respect to the computed eigenbasis. The plots show the evolution of the model vectors and edges of the GNG in more detail.

In Figure 7, the GNG model vectors are represented by the large points that are connected via edges. The clusters of small points show the sensorimotor inputs presented to the GNG during the run, with the input on the current time step indicated by a small circle. Each cluster of points represents a similar set of sensorimotor contexts experienced by the robot. The labels represent which color channels are active.

At time step 0, the two random initial model vectors happen to both represent having the red and blue channels active simultaneously. All of the initial experiences of the robot have only the green channel active, and new GNG model vectors are added to try to reduce the error between the existing units and the new sensory data. By time step 5, it is clear

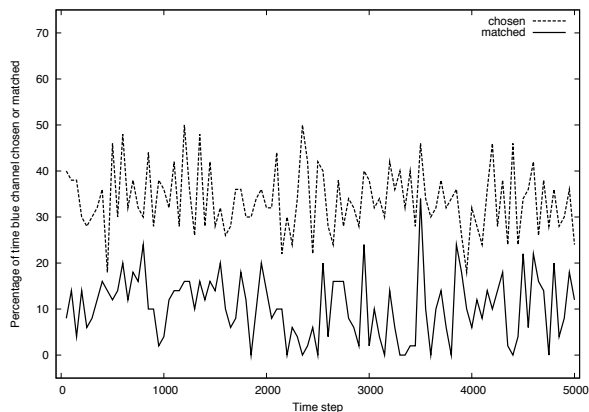


Figure 4: Results of a typical random controller.

that the GNG is growing to try to accommodate the repeated green channel exemplars near the center. By time step 25, a number of GNG units are now covering the green channel exemplars. At time step 35, the robot has turned enough to see both the blue robot and red robot simultaneously (with the green walls) for the first time. The GNG is again growing to accommodate this new event. Because the GNG always adds units between existing units, it can create intermediate model vectors that may not be representative of any of the data encountered so far. By time step 65, the model vectors at the exterior of the GNG are well matched with the data, while those at the center are not. Near the end of the run, at time step 4000, many of these intermediate units and edges have been removed, and the structure of the GNG more closely matches the underlying data representation.

In the original IAC model, regions are sub-divided based only on the quantity of exemplars. Initially every experience is grouped within a single IAC region. Once this region grows beyond the size limit, it splits into two sub-regions. It can take quite a long time before the repeated splitting of IAC regions begins to accurately reflect the sensorimotor data. In contrast, CBIM's GNG regions are formed based on encountering novel experiences. Each unfamiliar event encountered by the robot is immediately marked by a new category, and only novel experiences will trigger the formation of categories.

3.2 *Intrinsically motivated behavior*

In order to demonstrate that CBIM categorizes its sensorimotor space appropriately and uses these categories to effectively select learning experiences, a series of control experiments were executed for comparison. In the control experiments, actions were simply selected randomly without the use of categorization, prediction, or intrinsic motivation. We will refer to occasions when the Rovio selects a color

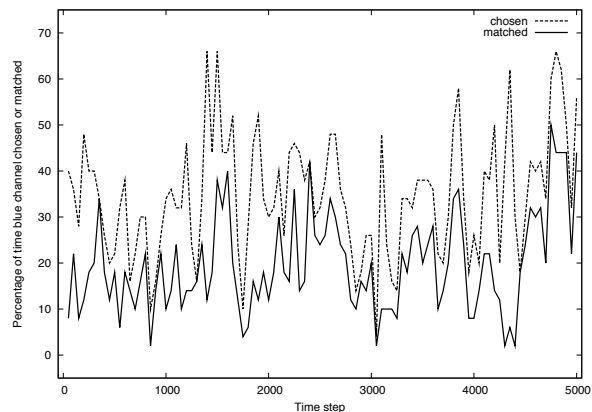


Figure 5: Results of a typical CBIM controller.

channel while an object of the corresponding color is in its camera view as a matched selection. Figure 4 shows the results of a typical control run. The correlation coefficient between the percentage of time when the blue channel is chosen and when the blue channel is actually matched is only 0.17.

However, in Figure 5, the channel choice and matched selection for CBIM is much more tightly coupled, with a correlation coefficient of 0.57. When this run is divided into thirds, the correlation coefficients for each third improves from 0.42, to 0.59, ending at 0.65. This increase in correlation between choosing the blue channel and seeing the blue object indicates that Rovio was progressively learning to track the movement of the small blue robot over time.

Based on the predictability of each object, we expected that CBIM would cause the Rovio to first focus on the red stationary robot and then on the moving blue robot. Figure 6 shows the same CBIM run from Figure 5 in which a shift in focus from the red object to the blue object can be seen. In the first 1500 steps, the Rovio was not very successful at finding either the red or blue object. Then, in the middle of the run, it was able to find and focus on both the red and blue objects in turn, with a peak in finding the red object at about 2800 steps. After about 4300 steps, there is a clear shift in focus away from the red object and toward the blue object. This provides evidence of a developmental trajectory.

In the five control experiments done, the random action selection led to a focus of 33% on each of the three color channels, as expected. In contrast, in the 10 CBIM experiments done, the intrinsically motivated action selection led to a more varied focus. In three of the experiments, blue was the primary focus 37% of the time on average. In three of the experiments, red was the primary focus 37% of the time on average. Finally, in the remaining four experiments, both blue and red were the primary focus 34% of the time on average for each. In a statistical analysis of

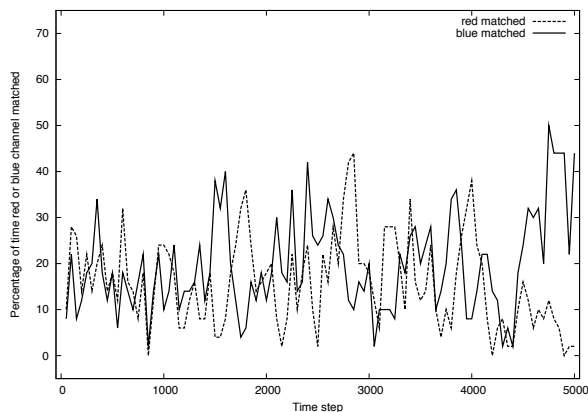


Figure 6: Evidence of a developmental shift from focusing on the more predictable red object to the harder to predict blue object.

the focus data, CBIM’s green focus was significantly lower ($p < .01$) than either its red or blue focus across all of the experiments. Given that the green background was visible on nearly every time step, and was the easiest of the color channels to predict, the fact that CBIM’s overall focus is primarily on the other two channels indicates that the intrinsic motivation is pushing the robot to explore the more challenging aspects of its environment.

4. Discussion

The results of our experiment suggest that the categorizational power of the GNG combined with the strength of IAC’s measure of learning progress is effective at developing a useful set of categories that allow the robot to maximize its learning potential in the given environment. The set of model vectors developed by the GNG is a reflection of the particular characteristics of the sensorimotor stream experienced by the robot, which grows only as much as is necessary to capture the topological relationships between the data. This approach avoids the use of ad hoc mechanisms such as region-splitting and the addition of unnecessary model vectors at fixed time intervals that were present in earlier models. Recently a new variation of IAC has been developed to address some of the inefficiencies of the region-splitting approach (Baranes and Oudeyer, 2009). Yet even this improved version could benefit from the bottom-up categorization approach used in CBIM.

One remaining challenge for CBIM is its limited ability to handle time-dependent relationships in the robot’s sensorimotor stream. Each expert bases its response only on the sensorimotor input at the current time step, without taking into account the recent past experiences of the robot. One possible direction of future work would be to incorporate recurrent neural networks into the model in order to take

advantage of temporal information.

References

- Baranes, A. and Oudeyer, P.-Y. (2009). R-IAC: Robust intrinsically motivated active learning. In *Proceedings of the IEEE International Conference on Learning and Development*, pages 1–6.
- Barto, A. G., Singh, S., and Chentanez, N. (2004). Intrinsically motivated learning of hierarchical collections of skills. In *The Third International Conference on Development and Learning*, pages 112–119.
- Berlyne, D. E. (1966). Curiosity and exploration. *Science*, 153(3731):25–33.
- Blank, D., Kumar, D., Meeden, L., and Yanco, H. (2006). The Pyro toolkit for AI and robotics. *AI Magazine*, 27(1):39–50.
- Fritzke, B. (1995). A growing neural gas network learns topologies. In Tesauro, G., Touretzky, D. S., and Leen, T. K., (Eds.), *Advances in Neural Information Processing Systems 7*, pages 625–632. MIT Press.
- Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. (2003). Developmental robotics: A survey. *Connection Science*, 15(4):151–190.
- Marshall, J., Blank, D., and Meeden, L. (2004). An emergent framework for self-motivation in developmental robotics. In *The Third International Conference on Development and Learning*, pages 104–111.
- Oudeyer, P.-Y., Kaplan, F., and Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(2):265–286.
- Provost, J., Kuipers, B., and Miikkulainen, R. (2006). Developing navigation behavior through self-organizing distinctive state abstraction. *Connection Science*, 18(2):159–172.
- Ryan, R. M. and Deci, E. L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, 55(1):68–78.
- Schmidhuber, J. (2006). Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. *Connection Science*, 18(2):173–187.

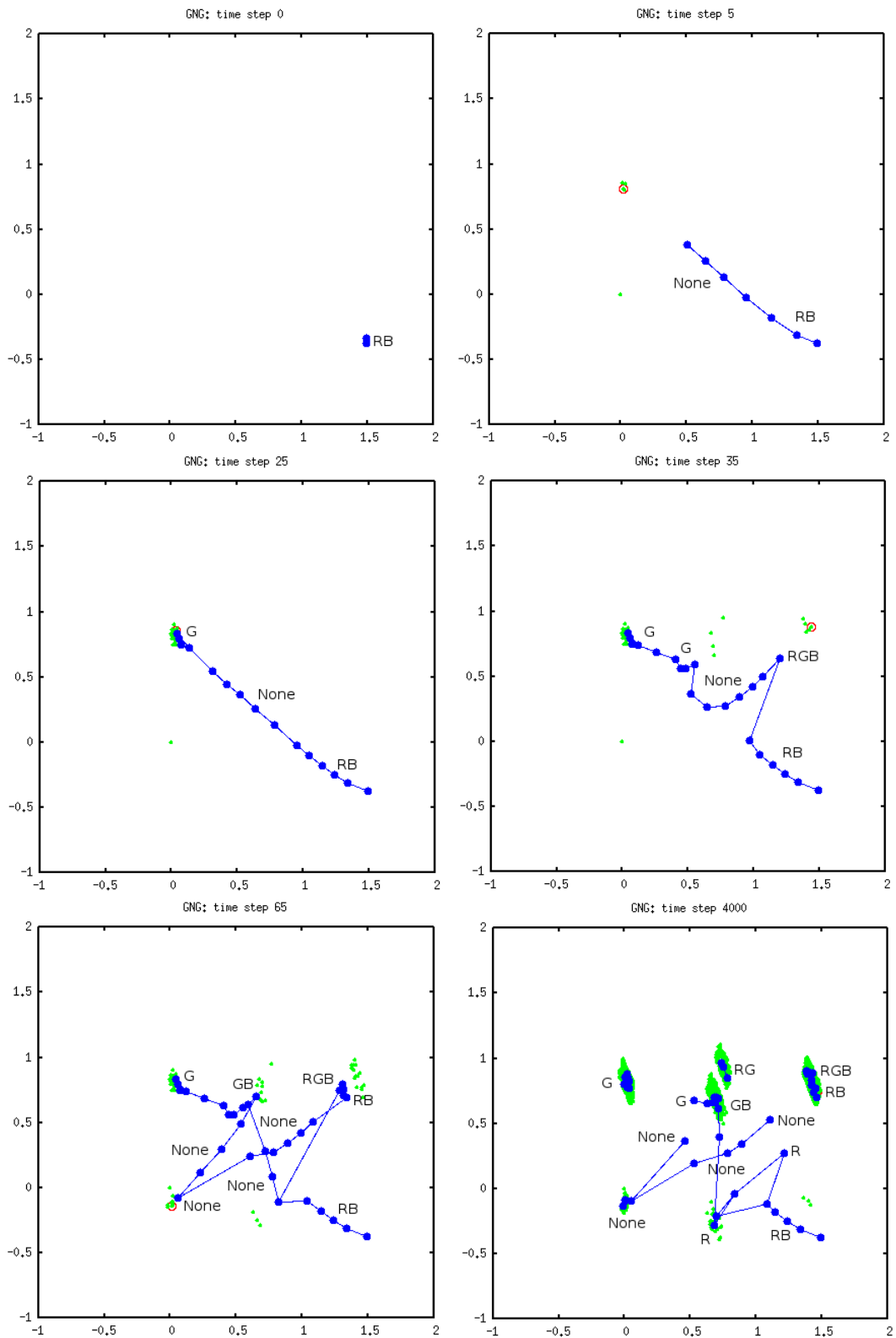


Figure 7: A 2-dimensional visualization of the 7-dimensional GNG model vectors and edges as they evolve over time in one CBIM run. The labels indicate which color channels are active for each sub-group of GNG units.

A Cognitive Robotic Model of Grasping

Zoran Macura* Angelo Cangelosi* Rob Ellis** Davi Bugmann**
Martin H. Fischer*** Andriy Myachykov***

*School of Computing & Mathematics

**School of Psychology

University of Plymouth

Plymouth, PL4 8AA UK

{zoran.macura,a.cangelosi,r.ellis,d.bugmann}@plymouth.ac.uk

***Department of Psychology

University of Dundee

Dundee, DD1 4HN UK

{m.h.fischer,a.myachykov}@dundee.ac.uk

Abstract

In this paper we present a cognitive robotic model of object manipulation (i.e. grasping) based on psychologically plausible embodied cognition principles. Specifically, the robotic simulation model is inspired by recent theories of embodied cognition, in which vision, action and semantic systems are linked together in a dynamic and mutually interactive manner. The robotic agent is based on a simulation model of the iCub humanoid robot. It uses a connectionist control system trained with experimental data on object manipulation. Simulation analyses show that the robot is capable to reproduce phenomena observed in human experiments, such as the Stimulus-Response Compatibility effect.

1. Introduction

The primary aim of our work is to develop a cognitive robotic model of the processes involved in object grasping and manipulation following the embodied cognition view of action and vision integration and micro-affordance effects (Tucker and Ellis, 2001). The task typically involves how to select, based on the agent's knowledge and representations of the world, one object from several, grasp the object and use it in an appropriate manner. This mundane activity in fact requires the simultaneous solution of several deep problems at various levels. The agent's visual system must represent potential target objects, the target must be selected based on task instructions or the agent's knowledge of the functions of the represented objects, and the hand (in this case) must be moved to the target and shaped so as to grip it in a manner appropriate for its use.

This work will first be framed within the current literature on the psychological investigation on action, vision and language integration, and on the robotics and computational models of these cognitive phenomena. We will then present a simulation model of grasping based on the iCub humanoid platform. We discuss how this will be extended to perform experiments replicating known psychological data on micro-affordance effects and action/vision integration.

1.1 *Psychological Studies on Vision, Action and Language*

It is increasingly recognised that cognition should not be regarded as a set of disembodied processes, but is strongly determined by the constraints of its bodily implementation and it being situated in the world with which it interacts. In the case of visual cognition this embodied approach has led to an emphasis on the role of active vision in exploring the world, and therefore on the integration of vision and action (see for instance O'Regan and Noe, 2001). There is certainly accumulating human behavioural evidence that vision and action form a closely integrated and highly dynamic system (e.g. Tucker and Ellis, 1998, 2001; Craighero et al., 2002; Fischer and Hoellen, 2004).

One consequence of this integration of the vision and action systems is that seeing an object, even when there is no intention to handle it, potentiates elements of the actions needed to reach and grasp it. For instance participants who viewed photographs of common objects in order to decide whether they were manufactured or organic were facilitated in responding if the grip needed to make the response was one that could be used to handle the viewed object (Tucker and Ellis, 2001). So, for example, sig-

nalling that a pea was organic was easier (faster and more accurate) if a precision grip (using only the thumb and forefinger) was needed for the response compared to using a power grip (between the four fingers and palm). Similar object to action compatibility effects are observed for the hand of reach and the wrist rotation required to align the hand with an object (Tucker and Ellis, 1998; Ellis and Tucker, 2000). The authors coined the term ‘micro-affordances’ to describe these potentiated elements of an action.

Visual attention and eye movements are obviously fundamental components of human exploratory behaviour, and implicated in the integration of vision, action and language. Our eyes are exquisitely sensitive to the combined demands of vision, action and language processing. We move our eyes to project objects of interest onto the foveal area of high visual resolution. When we interact with objects, our eyes move ahead of the hand to support the on-line control of grasping (e.g. Bekkering and Neggers, 2002). Merely seeing objects activates plans for actions directed to them (e.g. Tucker and Ellis, 2001; Fischer and Dahl, 2007).

1.2 Computational Modelling of Vision, Action and Language

Researchers from different fields such as engineering and cognitive science, to name a few, have greatly benefited from the use of computational models. This has resulted in a plethora of computational approaches, amongst which some are based on cognitive and developmental robotics approaches. Such approaches provide us with a more integrative vision of action, language and cognition.

In the cognitive modelling literature, there has also been some work specifically focused on the integration of action and vision knowledge in cognitive agents and in connectionist models. For example, Arbib and colleagues have developed a neural model for action learning directly inspired by brain imaging studies on grasping in primates, and applied to action imitation learning simulations (Arbib et al., 2000). Haruno et al. (2001) proposed the Mosaic architecture for simulated object manipulation tasks, demonstrating that the model can generalise action-object associations depending on the object shape. Demiris and Simmons (2006) present a computational architecture using a hierarchical controller based on the minimum variance model of movement control (HAMMER: Hierarchical Attentive Multiple Models for Execution and Recognition) for implementing biologically plausible human reaching and grasping movements. Tsiotas et al. (2005) developed an artificial life model for simulating some of Tucker and Ellis (2001) findings. They used a simplified 2D arm model to study the evolutionary learning of ob-

ject micro-affordances.¹ In the area of connectionist modelling, Yoon et al. (2002) have proposed a neural network model for action and name selection for objects (NAM: Naming and Action Model) that supports the role of a direct perception-action route for action selection. This model uses abstract (localist) encoding of action, perceptual and semantic information, rather than providing a robotic implementation, but is useful as it focuses on the comparison of perceptual vs semantic information in action selection.

More recently, Caligiore et al. (2008) developed a biomimetic neural network constrained by anatomical, physiological and behavioural data in which an embodied ‘eye-hand’ system was used to interact with objects of varying sizes (i.e. small and large). Using this model they replicated Tucker and Ellis (2001) compatibility effect between object size and the type of grip used in a categorisation task on whether objects were natural or artefacts. The modules of this neural network system are directly inspired by known brain processing mechanism. The action properties of the agents behaviour are however limited to a static representation of the action representing the final grasping configuration.

Models of action and vision integration also provide a framework to develop models of language learning based on the symbol grounding approach (Harnad, 1990; Cangelosi et al., 2005). Numerous studies have recently focused on the design of linguistic communication between autonomous agents, such as robots or simulated agents. The agents’ linguistic abilities in these models are strictly dependent on, and grounded in, other behaviours and skills such as vision and action. Numerous sensorimotor, cognitive, neural, social and evolutionary factors contribute to the emergence and establishment of communication and language. For example, in these models there exists an intrinsic link between the communication symbols (words) used by the agent and its own cognitive representations (meanings) of the perceptual and sensorimotor interaction with the external world (referents), as denoted by these symbols. Such a grounded and embodied approach to language design is consistent with the psychologically-plausible theories of the grounding of language (Cangelosi and Riga, 2006).

In such cognitive robotic models, communication results from the dynamical interaction between the robot’s physical body, its cognitive system and the external physical and social environment. Some studies stress the grounding in action and sensorimotor processes, such as Marocco et al.’s (2003) model of robotic arms and Vogt’s (2001) mobile

¹Recall that a micro-affordance is a quality of an object which is perceivable by an individual and suggests to this individual a range of possible actions associated with it.

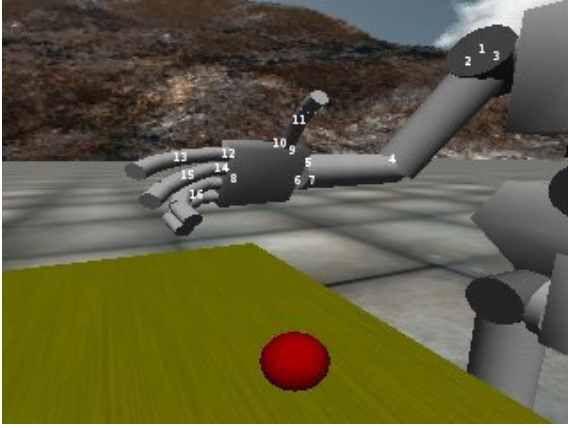


Figure 1: Simulated robot arm and hand with 16 controlled DoF and their corresponding movement ranges

#	joint	minimum angle (degrees)	maximum angle (degrees)
1	shoulder pitch	-95	90
2	shoulder roll	0	161
3	shoulder yaw	-37	100
4	elbow	6	106
5	wrist pronosupination	-90	90
6	wrist pitch	-90	10
7	wrist yaw	-20	40
8	hand finger adduction/abduction	-20	30
9	thumb opposition	-15	105
10	thumb proximal flexion	0	90
11	thumb distal flexion	0	90
12	index proximal flexion	0	90
13	index distal flexion	0	90
14	middle proximal flexion	0	90
15	middle distal flexion	0	90
16	ring & little flexion	0	115

robots. Other robotic models highlight the grounding through social interaction, such as Steels and Kaplan’s (2001) AIBO robots. On the other hand, some studies are based on simulating adaptive agents. They model the agent and its environment with a good degree of detail upon which emergent meanings can be directly constructed. These simulation models have focused on grounding in perceptual experience and in cognitive representations and sensorimotor interactions (e.g. Cangelosi, 2001).

In the next section we present a preliminary robotic model of action and vision integration for a grasping task that is directly inspired by this experimental literature on embodiment. This model provides us with a test-bed for the simulation of the vision-action-language integration processes observed in psychology experiments, and generates further insights and prediction on such phenomena.

2. Model

The cognitive robotic model presented here is directly inspired by recent theories of embodied cognition, in which the vision, action and semantic systems are linked together, in a dynamic and mutually interactive manner, within a connectionist architecture. We take inspiration from the Caligiore et al. (2008) model described above and extend it to consider a more realistic simulation of grasping behaviour and its time dynamics. This model proposes a combination of the epigenetic robotics methodologies with the “embodied connectionist” modelling approach. Epigenetic (developmental) robotics is based on the use of embodied robotic systems that are situated in a physical and social environment and are subject to a prolonged epigenetic developmental process for the acquisition of cognitive capabilities (Weng et al., 2001; Lungarella et al., 2003; Schlesinger et al., 2008). Embodied connectionism refers to the use of artificial neural networks for the learning and control of behaviour in cognitive robotic

agents. The integration of robotics and connectionist methodologies permits the transfer of the principles and advantages of connectionism and parallel distributed processing systems into embodied robotic agents (Cangelosi and Riga, 2006).

2.1 Simulated Robot

The robotic agent used in the simulation experiments is based on the humanoid iCub robot (Metta et al., 2008). In particular, the experiments use the recently developed open-source simulator of the iCub robot (Tikhanoff et al., 2008). The simulator has been designed to reproduce, as accurately as possible, the physics and the dynamics of iCub robot and its environment. The simulated iCub robot is composed of multiple rigid bodies connected via joint structures. It has been constructed collecting data directly from the robot design specifications in order to achieve an exact replication (e.g. height, mass, Degrees of Freedom) of the first iCub prototype developed at the Italian Institute of Technology in Genoa. The environment parameters on gravity, objects mass, friction and joints are based on known environment conditions.

The iCub robot is around 105cm high, weighs approximately 20.3kg and has a total of 53 degrees of freedom (DoF). These include 12 controlled DoF for the legs, three controlled DoF for the torso, six for the head and 32 for the arms. In particular, each arm is made up of three components (the arm, the forearm and the hand) where the arm and forearm include eight DoF and the hand another eight DoF, where each DoF movement range is constrained with the respective human DoF movement range (Fig. 1).

The robot’s vision system consists of two cameras located at the eyes of the robot. The simulated robot also has touch and force/torque sensors which receive tactile information and proprioceptive data on its own body posture. The proprioceptive sensors are located on the robot’s arm and hand and encode the

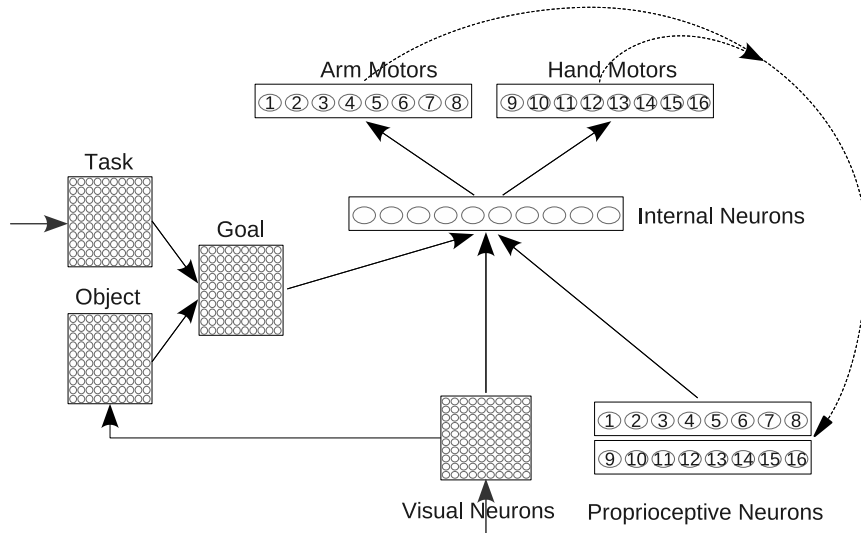


Figure 2: Neural network architecture for the robotic agent in which according to the visual input, the proprioceptive data and the task instruction an appropriate grasping movement is made

current angles of all 16 DoF of the arm (as listed in Figure 1). In addition, there are six tactile sensors in each hand – one on each finger and one on the palm – which indicate whether that particular body part is in physical contact with another object.

The simulator has full interaction with the world/environment. The objects within this world can be dynamically created, modified and queried which enables us to train the robotic agent to interact with objects so that it can acquire a sensorimotor representation of the objects through eye and hand movements. In learning to act on objects the robots neural controller will form embodied representations of those objects and as a consequence future encounters with these objects will cause them to afford the associated actions (micro-affordances).

2.2 Neural Network Architecture

A connectionist network is used to learn and guide the behaviour of the robot and to acquire embodied representations of objects and actions. The neural architecture, based on the Jordan recurrent architecture, has recurrent connections to permit information integration and the execution of actions such as grasping (Marocco et al., 2003). The network is depicted in Figure 2, and it is made up of four 2D maps of 10x10 neurons, 16 proprioceptive neurons, 10 internal neurons and 16 motor neurons. Specifically, the 16 output motor neurons control the DoF of the robot’s right arm (see Fig. 1) that performs the grasping task and the 16 proprioceptive neurons in input encode the current angles of the right arm’s DoF, which feed into the neural network thus creating a recursive structure (Fig. 2).

The visual input to the robot’s neural controller

consists of pre-processed information regarding visual object properties (i.e. shape and size). This information is processed directly from the physics simulator (Fig. 3: *Real Image*) by using three edge-detection Sobel filters – where each filter is sensitive to either red, green or blue component of the object’s colour. The result of the Sobel filtering is an image where only the edges of the objects in vision are encoded (Fig. 3: *Sobel*). An assumption in this model is that the eyes always foveate the target object. The foveated area of the image is in turn processed – where the activated edges are encoded as 1 and everything else as 0 – resulting in a 10x10 2D map encoding the shape and size of the foveated object (Fig 3: *Visual Input*), which constitutes the visual input for the neural network.

As well as being fed into the internal neurons, the visual input is also fed into the *Object* map, which is used to encode the objects’ identity. On the other hand, the *Task* map encodes the different tasks that can be performed on objects, namely a normal grasping task or a categorisation task akin to Tucker and Ellis (2001) psychological experiments. *Object* and *Task* maps in turn feed into the *Goal* map, which encodes the information about the current goal of action depending on the task and object identity. These three maps are implemented as Kohonen self-organising maps (SOMs) and are analogous to inferior temporal cortex (IT), medial temporal cortex (MT) and prefrontal cortex (PFC) in Caligiore et al.’s (2008) model, where the use of Kohonen maps for IT and PFC is justified by studies suggesting that these cortical areas are involved in high-level visual processing and categorisation (Miller et al., 2002; Shima et al., 2007). Note that these SOMs are just an approximation of the relevant cortices

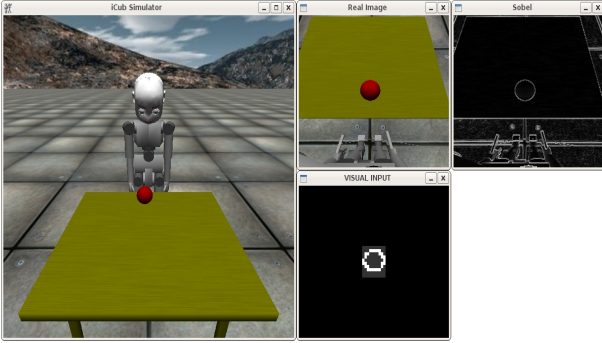


Figure 3: Visual processing in the robotic simulation. *Real Image* is the image taken from the robot’s eyes. The eyes always foveate on the target object, after which Sobel filters are applied to the *Real Image* producing the *Sobel* image. The result of the visual processing is the *Visual Input* image – a 10x10 2D map of 0s and 1s encoding the shape and size of the foveated object

in functional terms, and not the real physiological-anatomic analogues.

2.3 Training

Simulation experiments focus on the training of the robot to use objects using different manipulation modalities (e.g. precision grip vs power grip, respectively, for small objects – “cherries” – and for big objects – “apples”) and also to be able to replicate psychological experiments where the objects can be categorised using different grips (e.g. precision grip for artefacts and power grip for natural objects).

There are four objects in the simulation: 2 larger objects (‘big-ball’ and ‘big-cube’) for power grips; and 2 smaller objects (‘small-ball’ and ‘small-cube’) for precision grips. In this model, the round objects (big and small balls) are viewed as natural objects, whereas the cubes are viewed as artefacts. The training data consists of a set of grasping sequences for each object, which have been normalised in the range of 0–1 from the movement ranges shown in Figure 1. Each sequence is made up of 10 time-based steps where the first step represents the initial arm and hand position (pre-grasp) whereas the last step represents the final grasping posture (appropriate grasp for the target object).

There are two training phases in the model. In the first training phase the robot learns to appropriately grasp objects, while in the second phase the robot learns how to categorise objects using power and precision grips, as seen in psychological experiments. Before the main training begins, the *Object*, *Task* and *Goal* SOMs are trained individually off-line. The *Object* SOM is trained to categorise the four objects where a different cluster of neurons is activated for each object, and the size of the cluster

is dependant on the object size (e.g. large objects activate a greater cluster of neurons) (see Caligiore et al., 2008). The *Task* SOM is trained to activate two different patterns of neurons to represent the two different tasks in psychological experiments, namely normal grasp and categorisation task. Finally, the *Goal* SOM is trained to represent the current goal, where there are eight different clusters of neurons that can get activated depending on the object and the task.

During the first training phase, the four objects are repeatedly presented to the simulated robot, which in turn tries to learn the micro-affordance-based behaviours for each object. At the beginning of each trial an object is placed at the same position on the table and the robot foveates the object. The processed visual input is then fed into the neural network along with the proprioceptive data encoding the current position of the robot’s right arm DoF. The *Task* SOM in this phase is always activated with the pattern representing the grasping task. Each internal neuron performs a weighted summation of the inputs, which then passes a sigmoid (nonlinear) activation function. The motor neurons perform a similar weighted summation of the internal neurons’ outputs and their outputs result in a grasping movement appropriate for the target object. In this learning phase the network parameters are continuously adjusted using a back-propagation algorithm until the robot learns to form appropriate associations between the object’s shape and the hand shape.

In the second training phase, the robot learns to categorise objects with different grips. The training follows a similar procedure to the one outlined above, with the main difference being in the way the *Task* SOM is activated. In this phase, the *Task* SOM instead of always being activated with the pattern representing the grasping task (as was the case in the first training phase), is first activated with a new (random) pattern representing the categorisation tasks, and in the next cycle with the previous pattern of the grasping task. This enables the robot to learn suitable grasps to correctly categorise objects depending on the *Object* and *Task* SOMs activations.

3. Results

A total of five different grasping sequences were defined for each object – of which four sequences were used for training the robot and the fifth was used for testing purposes. The five grasping postures differ by the final position and rotation of the hand with respect to the object. The learning rate for the back-propagation algorithm was set to 0.075. For each training cycle an object and one of its four grasping sequences was chosen randomly and presented to the robot. This was repeated 12000 times, where

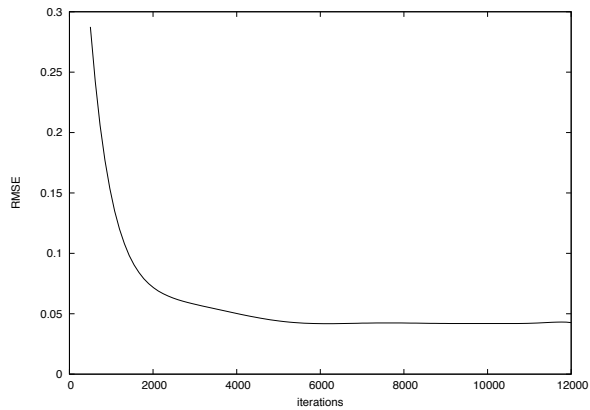


Figure 4: Average RMSE during the neural network training

both training phases lasted 6000 iterations each.² After training, the network was tested on both grasping and categorisation tasks for all five grasping sequences – including the fifth (unseen) grasping sequence – for each of the four objects in order to establish whether the robot learned how to grasp and categorise objects appropriately. The results presented here are all averages of 12 trained neural networks, where the first six networks were trained to categorise natural objects (balls) with a power grip and artefacts (cubes) with a precision grip and the other six networks were trained to do the opposite (power grip for artefacts and precision grip for natural objects).

Figure 4 shows the root mean squared error (RMSE) of the network during the training phase. As expected, at the beginning of the training the error between the motor outputs and the desired targets (joint positions of the right arm) is high (around 0.3). After roughly 2000 iterations the RMSE drops to 0.05 and stabilises around this value, indicating that the simulated robot has been able to successfully learn appropriate grasps for the four objects.

One important test in this model of object grasping and micro-affordances is the comparison of the congruent (where the categorisation grip is in agreement with the natural grip) and incongruent (where there is mismatch between the categorisation grip and the natural grip) conditions. The trained neural networks were presented each object in turn, where the desired target depended on the task being performed. The results are depicted in Figure 5, which shows the average RMSE values of the 12 networks for congruent and incongruent trials. We assume that RMSE is analogous to reaction time used in

²Recall that in the first phase the *Task* SOM is always activated with the pattern representing the grasping task. In the second phase this happens only in half of the cases (3000 iterations) and in the other half the *Task* SOM is activated with the pattern representing the categorisation task.

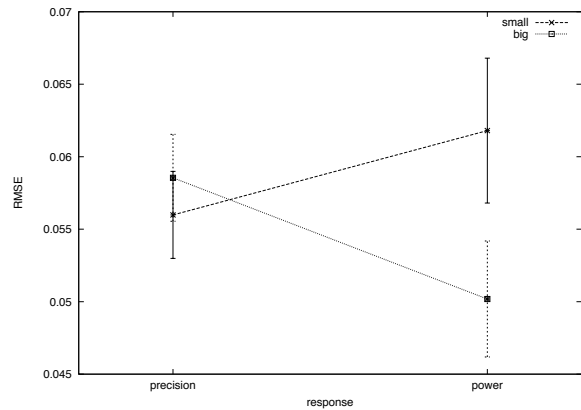


Figure 5: Compatibility effect in congruent and incongruent trials

psychological experiments done in the Tucker and Ellis (2001) study. An ANOVA on response times was performed with two factors: congruency and object size, and both factors were statistically significant. As can be seen in Figure 5 the results are in agreement with psychological experiments where reaction times are faster in congruent than in incongruent trials. In addition, the reaction times for larger objects were faster than for smaller object, as was also the case in psychological experiments. This indicates that the robot was able to generalise a grasping sequence for each task and object from the four grasping sequences used in training, hence learning to appropriately grasp and categorise objects based on their shapes and sizes.

4. Conclusion & Future Work

We have shown how the proposed cognitive robotic model was able to learn object micro-affordances and appropriately grasp and categorise an object depending on its shape and size. Tests on congruent/incongruent tasks also demonstrate that the robot's neural controller uses micro-affordance information about the objects replicating the well known Stimulus-Response Compatibility phenomenon observed in psychology experiments. Future analyses will investigate the internal representations used by the network in responses to various task demands (grasping vs. categorisation), to different level of object/grasp congruency and to the interaction between objects with conflicting micro-affordance. Analyses of the neural network representations in controlling behaviour, and of the time-course of processes and representation activated by the robot's neural controller, will be used to better understand behaviour observed in human participants and to derive novel predictions about interactions between vision and action.

One additional extension of this model regards the

inclusion of linguistic information during training, such as for the names of objects and actions. This extended model will permit detailed investigations of the effects of language on micro-affordance effect (Tucker and Ellis, 2004).

The main goal of this psychologically-plausible model for the study of grasping behaviour in humanoid robots, in addition to advancing our understanding of vision-action-language integration, will provide us with a set of cognitively-plausible design principles for developing vision, action and linguistic capabilities in robots and their use in interactive cognitive systems and autonomous robotics.

The cognitive robotic platform developed here can be used as a tool to test feasibility of the vision-action-language integration mechanisms identified during experimental studies, in addition to demonstrating the technological potential in such an approach. Observation and analyses of the robot's cognitive and linguistic capabilities will also result in the production and test of new predictions about mechanisms integrating vision, action and language. The replication in a robotic model of the psychological phenomena observed in experimental studies will have the advantage of permitting the fine analysis and understanding of the neural and behavioural processes that contribute to action-vision-language integration (Cangelosi and Parisi, 2002).

Acknowledgements

This research is supported by the VALUE project (EPSRC Grant EP/F026471) and the iTalk project (EU FP7).

References

- Arbib, M. A., Billard, A., Iacoboni, M., and Oztop, E. (2000). Synthetic brain imaging: Grasping, mirror neurons and imitation. *Neural Networks*, 13:975–997.
- Bekkering, H. and Neggers, S. F. W. (2002). Visual search is modulated by action intentions. *Psychological Science*, 13:370–374.
- Caligiore, D., Borghi, A. M., Parisi, D., and Baldassarre, G. (2008). Affordances and compatibility effects: a neural-network computational model. In *The 11th Neural Computation and Psychology Workshop*. University of Oxford, Oxford, UK.
- Cangelosi, A. (2001). Evolution of communication and language using signals, symbols, and words. *IEEE Transactions on Evolutionary Computation*, 5(2):93–101.
- Cangelosi, A., Bugmann, G., and Borisyuk, R., (Eds.) (2005). *Modeling Language, Cognition and Action: Proceedings of the 9th Neural Computation and Psychology Workshop*. World Scientific, Singapore.
- Cangelosi, A. and Parisi, D., (Eds.) (2002). *Simulating the evolution of language*. Springer-Verlag, London.
- Cangelosi, A. and Riga, T. (2006). An embodied model for sensorimotor grounding and grounding transfer: Experiments with epigenetic robots. *Cognitive Science*, 30(4):673–689.
- Craighero, L., Bello, A., Fadiga, L., and Rizzolatti, G. (2002). Hand action preparation influences the responses to hand pictures. *Neuropsychologia*, 40:492–502.
- Demiris, Y. and Simmons, G. (2006). Perceiving the unusual: temporal properties of hierarchical motor representations for action perception. *Neural Networks*, 19(3):272–284.
- Ellis, R. and Tucker, M. (2000). Micro-affordance: The potentiation of components of action by seen objects. *British Journal of Psychology*, 91:451–471.
- Fischer, M. H. and Dahl, C. (2007). The time course of visuo-motor affordances. *Experimental Brain Research*, 176(3):519–524.
- Fischer, M. H. and Hoellen, N. (2004). Space-based and object-based attention depend on motor intention. *Journal of General Psychology*, 131(4):365–377.
- Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42:335–346.
- Haruno, M., Wolpert, D. M., and Kawato, M. (2001). Mosaic model for sensorimotor learning. *Neural Computation*, 13:2201–2220.
- Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. (2003). Developmental robotics: A survey. *Connection Science*, 15(4):151–190.
- Marocco, D., Cangelosi, A., and Nolfi, S. (2003). The emergence of communication in evolutionary robots. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 361(1811):2397–2421.
- Metta, G., Sandini, G., Vernon, D., Natale, L., and Nori, F. (2008). The icub humanoid robot: an open platform for research in embodied cognition. In *Proceedings of IEEE Workshop on Performance Metrics for Intelligent Systems (PerMIS08)*.

- Miller, E. K., Freedman, D. J., and Wallis, J. D. (2002). The prefrontal cortex: categories, concepts, and cognition. *Philosophical Transactions of The Royal Society B: Biological Sciences*, 357:1123–1136.
- O'Regan, J. K. and Noe, A. (2001). A sensorimotor account of vision and visual consciousness. *Brain and Behavioral Sciences*, 24(5):939–1031.
- Schlesinger, M., Berthouze, L., and Balkenius, C., (Eds.) (2008). *Proceedings of the Eighth International Conference on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*. Lund University Cognitive Studies, 139, LUND: LUCS.
- Shima, K., Isoda, M., Mushiake, H., and Tanji, J. (2007). Categorization of behavioural sequences in the prefrontal cortex. *Nature*, 445:315–318.
- Steels, L. and Kaplan, F. (2001). Aibo's first words: The social learning of language and meaning. *Evolution of Communication*, 4(1):3–32.
- Tikhonoff, V., Cangelosi, A., Fitzpatrick, P., Metta, G., Natale, L., and Nori, F. (2008). An open-source simulator for cognitive robotics research: The prototype of the icub humanoid robot simulator. In *Proceedings of IEEE Workshop on Performance Metrics for Intelligent Systems (PerMIS08)*.
- Tsiotas, G., Borghi, A., and Parisi, D. (2005). Objects and affordances: An artificial life simulation. In *Proceedings of the Cognitive Science Society*, pages 2212–2217.
- Tucker, M. and Ellis, R. (1998). On the relations between seen objects and components of potential actions. *Journal of Experimental Psychology: Human Perception and Performance*, 24:830–846.
- Tucker, M. and Ellis, R. (2001). The potentiation of grasp types during visual object categorization. *Visual Cognition*, 8(6):769–800.
- Tucker, M. and Ellis, R. (2004). Action priming by briefly presented objects. *Acta Psychologica*, 116:185–203.
- Vogt, P. (2001). Bootstrapping grounded symbols by minimal autonomous robots. *Evolution of Communication*, 4(1):87–116.
- Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., and Thelen, E. (2001). Autonomous mental development by robots and animals. *Science*, 291:599–600.
- Yoon, E. Y., Heinke, D., and Humphreys, G. W. (2002). Modelling direct perceptual constraints on action selection: The naming and action model. *Visual Cognition*, 9(4/5):615–661.

Navigation via Pavlovian Conditioning: A Robotic Bio-Constrained Model of Autoshaping in Rats

Francesco Mannella¹ Ansgar Koene² Gianluca Baldassarre¹

¹Laboratory of Computational Embodied Neuroscience,
Istituto di Scienze e Tecnologie della Cognizione, Consiglio Nazionale delle Ricerche
Via San Martino della Battaglia 44, I-00185 Roma, Italy

² Adaptive Behaviour Research Group,
Department of Psychology, University of Sheffield, Sheffield S10 2TP, UK

Abstract

Within the autonomous robotics literature, bio-inspired models of navigation in organisms (e.g. rats) usually rely on instrumental conditioning processes based on the learning of associations between places in the environment and navigation actions leading to rewarded goal places. This paper presents a neural-network model capable of solving navigation tasks on the basis of Pavlovian conditioning processes ('autoshaping') which allow transferring innate approaching behaviours from biologically salient stimuli (e.g., food) to neutral stimuli (e.g., a landmark seen from far away and close to the food). The overall architecture and functioning of the model is biologically constrained on the basis of relevant neuroscientific anatomical and physiological knowledge on amygdala, nucleus accumbens, and ventral tegmental area. The model is tested with a simulated robotic rat engaged in autoshaping and devaluation experiments. The results show that, although the model allows solving only simple navigation tasks, it produces fast learning and a flexible sensitivity of behaviour to internal states typical of Pavlovian processes. The model is also important for the investigation of adaptive behaviour in general as it clarifies the nature of some Pavlovian core mechanisms which play a key role in several forms of learning.

1. Introduction

Navigation is a fundamental adaptive behaviour which allows organisms to displace in space so to get in contact with resources scattered in the environment and use them to increase their survival and reproduction chances. For this reason, the brain machinery emerged during evolution to subservise navigation behaviours is rather sophisticated and based on

multiple systems. Most models of animal navigation proposed within autonomous robotic literature are based on instrumental processes (for some classical reviews, see Trullier et al., 1997; Filliat and Meyer, 2003a,b). Instrumental processes allow organisms to form associations between stimuli and actions on the basis of the resulting reinforcing outcomes (Domjan, 2006). Some of the most influential models use reinforcement-learning algorithms (e.g., based on the Temporal Difference rule, Sutton and Barto, 1998) to form, via a *long* training, associations between places and the actions directed to achieve rewarded places. Those of these models which are more biologically constrained assume that places are represented in 'place cells' of hippocampus (HIP) (O'Keefe et al., 1998) and that actions are selected and triggered in a *reactive* fashion by nucleus accumbens core (NAccC) (Arleo and Gerstner, 2000), or, alternatively, that actions are triggered in a *proactive* fashion based on planning processes located in prefrontal cortex (PFC) (Martinet et al., inpr).

The important processes involving complex spatial elaborations performed by HIP, NAccC and PFC has led to overlook some processes underlying navigation behaviours which are simpler but also faster and more flexible than instrumental ones. In this respect, a main tenet of the paper is that an important class of these simpler processes are based on Pavlovian conditioning mechanisms. Pavlovian conditioning (Lieberman, 1993) is an experimental paradigm in which a stereotyped 'unconditioned response' (UR), innately associated with, and triggered by, a biologically salient 'unconditioned stimulus' (US), might become associated with, and so triggered by (so becoming a 'conditioned response', CR), an innately neutral 'conditioned stimulus' (CS), if the CS regularly precedes the US. For example, the UR of salivation, innately triggered by the US of the taste or smell of food, might become associated and triggered by a CS consisting in the sight of food if the CS is repeatedly followed by the US.

Approaching food or conditioned stimuli (e.g., a light) is a typical UR/CR studied in Pavlovian experiments (in this case called ‘autoshaping’). Autoshaping mechanisms allow organisms to approach (CR) a neutral stimulus (CS) if this has been regularly paired with an appetitive stimulus (US).

Pavlovian mechanisms related to approaching have a great evolutionary advantage. The approaching behaviour is formed by a set of motor routines which involve a complex rhythmic pattern of muscle activations which reduce the spatial distance with the target. The advantage rendered by autoshaping mechanisms is that the formation of a *fast-learnable and simple association* between an US (e.g., food) and a CS (e.g., a big landmark close in space to the food and visible from far away) can allow organisms to *rapidly transfer the whole complex target-approaching behaviour* (UR) to the CS.

Pavlovian navigation has also a second important advantage in terms of flexibility as it can be modulated by body states. In fact, internal representations of USs (via the activation of which approaching responses are triggered) can be directly modulated by internal states. For example, the satiation for a particular food (US) can prevent its internal representation from being activated by the activation of a CS associated to it, so stopping the triggering of costly and inuseful URs associated to it (e.g., salivation and approaching).

The main contribution of the paper is the proposal of a model which represents a first important step towards a full and detailed understanding of Pavlovian-based navigation processes. This not only has great relevance for neuroscience and psychology, but also for autonomous robotics for two reasons: (a) it suggests specific mechanisms for implementing quickly-learnable and flexible navigation behaviours; (b) Pavlovian mechanisms play a key role in many learning processes and so have an importance which goes well beyond navigation behaviours (see Mirolli et al., sub).

The rest of the paper is organised as follows. Section 2. illustrates the biological constraints of the model, Section 3. the setup of the simulated experiments, and Section 4. the model in detail. Section 5. presents the results of the autoshaping and devaluation tests, whereas Section 6. draws the conclusions.

2. Biological Evidence on Pavlovian Navigation Mechanisms

This section presents biological evidence which on one side supports the claim that organisms acquire some kinds of navigation skills based on Pavlovian mechanisms, and on the other side furnishes the anatomical and physiological constraints used to design the architecture and functioning of the model.

A first piece of evidence is that lesions of HIP does not prevent the acquisition and expression of autoshaping behaviours (Parkinson et al., 2000). This is fundamental as rules out that the spatial computations performed by HIP underlie such behaviours.

Another important piece of evidence is related to the basolateral complex of AMG (BLA). BLA is the main locus where CS-US Pavlovian association processes take place (Cardinal et al., 2002a; Knapska et al., 2007; McDonald, 1998; Pitkänen et al., 2000). Surprisingly, BLA is not necessary for learning and expression of autoshaping (Parkinson et al., 2000).

BLA, however, is necessary for the flexible modulation of Pavlovian mechanisms based on internal states. An example of this, relevant to this work, is that it is necessary to allow satiation for one food to inhibit not only approaching to such food but also approaching to a CSs associated with it (Blundell et al., 2003). This without the need of relearning.

BLA is also necessary for the functioning of *second order conditioning*, that is conditioning of a neutral stimulus on the basis of the presentation of another neutral stimulus previously associated with it (this can be done ‘in extinction’, i.e. without presenting the US after the first CS; Cardinal et al., 2002a). This might be relevant to extend the model in the future and let it learn to approach a landmark (CS2) if this is followed by another landmark (CS1) previously associated with reward (US).

BLA is also capable of triggering *phasic dopamine* (DA) bursts via its connections with lateral hypothalamus (LH; Pitkänen et al., 2000). These types of DA signals are very important for learning.

Another important fact to consider is that the central complex of AMG (CEA) is needed for learning conditioned approach behaviours but not for expressing them (Cardinal et al., 2002b). This property seems related to the capacity of CEA of causing a population diffused activation of the ventral tegmental area (VTA) and a consequent production of *tonic dopamine*: this acts as a necessary precondition for phasic DA to trigger learning.

Tonic DA is also at the basis of *vigor* of actions, that is of the mechanisms for which the intensity and frequency of execution of actions can increase due to expectation of appetitive stimuli (cf. Niv et al., 2006).

A further important piece of evidence is that the ventral part of the striato-cortical system (Kandel et al., 2000) is needed to learn and express conditioned approach behaviours. In particular, lesions of the basal-ganglia and cortical components of such loops, namely respectively the nucleus accumbens core (NAccC; Cardinal et al., 2002b) and anterior anterior cingulate cortex (ACC; Cardinal et al., 2002b, 2003) prevent both learning and expression of conditioned approach.

3. The Simulated Rat, the Maze, and the Tasks

The robot used to test the model is a robotic rat ('ICEAsim') developed within the EU funded project ICEA on the basis of the physics 3D simulator WebotsTM. The model was written in MatlabTM (Webots has an interface for Matlab code). The numerical integration of the equations of the model is performed with the Euler method and an integration time step of 0.05 (also used for the 3D simulator). The robotic setup used to test the model is shown in Figure 1 and it is now briefly described.

The training and test environment is composed by a grey-walled Y maze (only the two upper arms of it were used: the lower arm will be used in future work). Each upper arm contains a different landmark which the rat can see from far away, and a rectangular food dispenser, which the rat can see only from the middle of the arm onward. The two food dispensers contain food A and food B respectively. When the rat touches a food dispenser it receives a rewarding signal corresponding to the ingestion of the food.

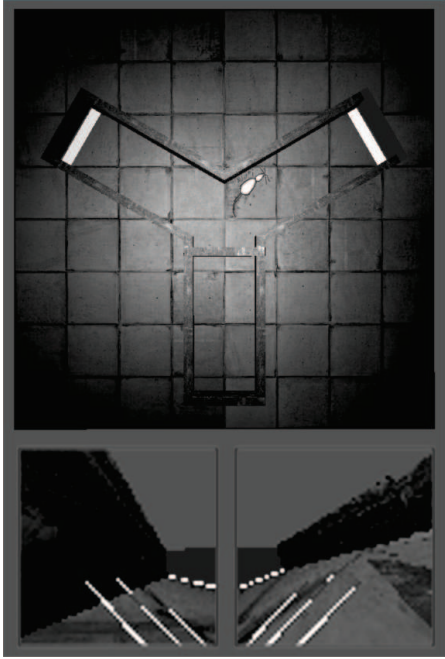


Figure 1: Top: The simulated Y maze and robot. Bottom: The left and right retina images perceived by the rat while positioned as indicated in the top graph.

The simulated rat is a two-wheel robot equipped with various sensors. Among these, the tests reported here use two cameras (furnishing a panoramic 300 degrees view) and the whisker sensors. The rat uses the cameras to detect the landmarks (red and blue) and the food dispensers (green and yellow). Suitably tuned pre-processing colour filters allow the

system to perceive stimuli as binary signals. Landmarks are seen from far away, for example from the crossing of the Y maze, but only when positioned in the frontal zone of the rat (within a range of 90°). Also the food dispensers are visible only if within the frontal zone, but their visibility is limited to positions within a half-arm distance. The rat is also endowed with two binary sensors which detect the ingestion of respectively food A or B, and with two binary *internal* sensors respectively encoding satiety for either food A or B.

The rat also uses the whiskers, activated with one if bent beyond a certain threshold and zero otherwise, to detect contacts with obstacles. The whiskers are used to control a low-level hardwired 'obstacle avoidance routine' which 'overwrites' all other actions and leads the rat away from obstacles.

The actuators of the rat are two motors which can independently control the speed of the two wheels. The system controls such speed by selecting one of three hardwired routines: 'turn-left' and 'turn-right', which lead the robot to respectively turn anticlockwise or clockwise on the spot, and 'go-straight' which leads the robot to move forward. If none of these routines is selected and active, the speed of wheels is set to zero. A further 'consummatory routine', mimicking eating, is triggered when the rat is on a dispenser and perceives the related US.

The rat undergoes three training/testing phases:

1. *Pre-training phase*. In this phase, the rat is first trained for 2 mins, divided in trials, in the food-B maze arm without the landmark and blocked with a wall at the central end; then it is trained in a similar condition in the food-A arm. Trials terminate either after 20 sec or when the rat ingests the food. In this phase the rat learns to associate the seen foods (CSs) with the ingested foods (USs).
2. *Training phase*. This phase lasts 2 mins, divided in trials as in the first phase, and involves the two upper arms. In this phase the rat learns to associate the landmarks (CSs) with the seen foods (CSs) and the ingested foods (USs).
3. *Devaluation phase*. This phase is composed of three sub-phases of 4 mins each: one with both fully-valued foods, one with the devalued food A, and one with the devalued food B. Each sub-phase is divided in trials as in the other two phases. In this phase the learning coefficients are set to zero to collect more controlled data. This phase allows testing if the rat has a tendency to explore more extensively the maze arm where the non-devalued food is located.

4. The model

This section uses the following conventions: bold capital letters (\mathbf{X}) represent matrices, bold small letters (\mathbf{x}) represent vectors and small letters (x) represent scalars. The notation $[x]^+$ means that the

positive part of x is considered, while the notation $[x]^-$ means that the negative part of x is considered. The function $\phi(x, \theta)$ returns 1 if $x > \theta$, 0 otherwise. Note that each unit activation is here assumed to represent the firing rate of a population of neurons reached by a similar input pattern.

Figure 2 shows the architecture of the model based on three main components: (a) the AMG: this is responsible for implementing the stimuli associations of Pavlovian conditioning; (b) the striatocortical system formed by the ventral basal ganglia (VBG: these are a set of nuclei formed by the NAccC, the subthalamic nucleus, STN, and the substantia nigra pars reticulata, SNpr) the dorsomedial thalamus (DM) and the ACC: this is responsible for selecting the actions to execute; (c) the dopaminergic system formed by LH and VTA: DA modulates both the learning processes and the speed of selection and duration of execution of actions (the latter is the correspondent of action vigor in the model, see Section 2.).

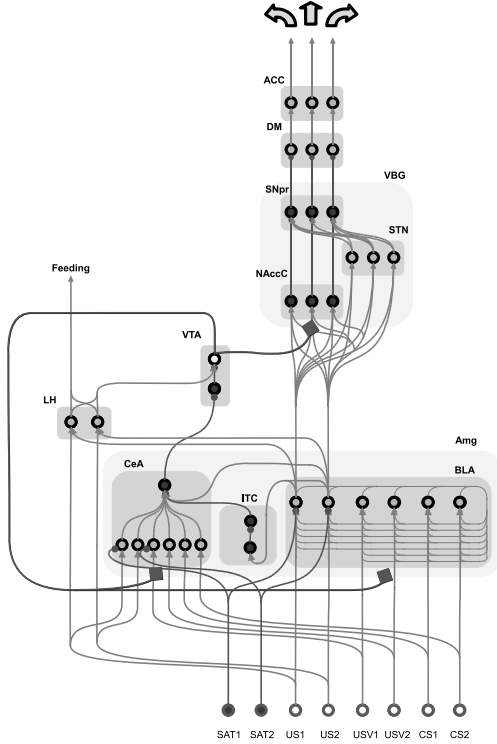


Figure 2: The architecture of the model.

With the exception of the units of AMG (see Section 4.1), all the units of the model are leaky integrators as described in Amari (1977):

$$\begin{aligned} \tau \dot{u}_i &= -u_i + \kappa_u I + \sum_j w_{ij} \cdot v_j \\ v_i &= [\tanh[u_i]]^+ \end{aligned} \quad (1)$$

where u_i and v_i are respectively the potential and the activation of unit i , I is the input signal from

either the external environment or the body, κ_u is a multiplying coefficient, and w_{ij} is the weight of an afferent connection from another unit j .

4.1 The Amygdala, an CS-CR and CS-US Associator

This section first describes the general functioning and learning of AMG units and then the specific functions of BLA and CEA.

BLA and CEA are each formed by six input units which receive one-to-one input signals from the six external input units of the model: two encoding visual conditioned stimuli, two encoding the two seen foods, and two encoding the taste of ingested food. Two additional internal input units of the model, respectively encoding the satiation for the two foods, send strong one-to-one inhibitory signals to the two units of BLA and CEA encoding the two food tastes. Another group of units (intercalated nuclei, ITC) serve as a disinhibitory interface between BLA and CEA (see Paré et al., 2004)

The units of BLA and CEA (denoted with **bla** and **cea**) are different from the other units, in particular each one activates in correspondence to stimuli onset and then fades away (many single neurons in brain have this property). For each AMG unit, this onset-detection function is achieved on the basis of two leaky integrators, o_{in} and o_{out} :

$$\begin{aligned} \tau_1 \dot{o}_{in} &= -o_{in} + I \\ \tau_2 \dot{o}_{out} &= -o_{out} + [I - o_{in}]^+ \end{aligned} \quad (2)$$

This kind of activation is needed to allow the internal connections of BLA and CEA to be updated on the basis of a ‘differential Hebb rule’ (Porr and Wörgötter, 2003; Mannella et al., 2007). This rule captures the temporal correlation (or ‘apparent causality’) existing in incoming input patterns. In particular, if one has two units with two reciprocal connections and the first unit tends to be activated within a certain time window before the second unit, the rule tends to increase the weight of the connection which goes from the first unit to the second unit, and at the same time tends to decrease the weight that goes from the second unit to the first unit. In detail, the learning rule works as follows. First the leaky traces of the derivatives of the activation of the onset units are computed:

$$\tau_{tr} \dot{tr} = -tr + \kappa_{tr} \cdot \dot{o}_{out} \quad (3)$$

where κ_{tr} is a multiplying factor. Then a difference in the sign of the traces of the presynaptic and postsynaptic unit determines the amount of the increment of the weights:

$$\begin{aligned} \Delta w_{ij} = & \\ & \eta \cdot (\phi(da, \theta_{da}) \cdot (da - \theta_{da})) \cdot \\ & \left([tr_i]^- \cdot [tr_j]^+ - [tr_i]^+ \cdot [tr_j]^- \right) \cdot \\ & (\theta_{w_{ij}} - |w_{ij}|) \end{aligned} \quad (4)$$

where $\theta_{w_{ij}}$ is a weight-saturation threshold, da is the dopamine, and θ_{da} is the dopamine level above which learning takes place.

BLA units have lateral connections. When visual stimuli units and food-taste units are strengthened on the basis of Equation 4, the former ones acquire the ability to activate the output unit in the same way as done by USs.

BLA output responses consist in triggering, via LH, the activation of VTA output units: this leads to a phasic dopaminergic signal underlying learning (see Section 4.3). A second output reaches NAccC: this has the function of biasing the selection of actions taking place within VBG. A last output reaches CEA, and allows BLA processes to exert control on the output of CEA.

BLA US units are also reached by internal signals about satiety. Through these connections the activity of these units can be modulated by internal states, for example suppressed by satiation. In this way, the US can dynamically change its motivational value. This property is also transferred to CSs if they have been associated to USs within AMG.

CEA has six input units and one output unit connected to VTA. All internal connections are trained with the differential Hebb rule mentioned above, with the exception of those carrying the information about the USs which are fixed ('innate'). This learning process allows the formation of CS-CR associations (stimulus-response associations).

CEA can cause DA release via a disinhibition of the internal population of VTA. This mechanism is able to maintain tonic dopaminergic efflux upon baseline through time. This DA is not sufficient to trigger learning within NAccC but at the same time it is necessary to allow the BLA signal to VTA (via LH) to cause DA-based learning (see Section 4.3). Moreover, tonic DA acts as a multiplier of signals from BLA to NAccC, so implementing a 'vigor' function (see Section 2. and 4.2).

CEA receives input not only from external stimuli, but also from BLA. This allows BLA to have access to the output of CEA (DA in this case). Moreover, the internal signals related to satiety modulate the US input units of CEA similarly to what happens for BLA.

4.2 The Striatocortical System

The VBG component is a simplified implementation of the basal ganglia 'GPR' model proposed by Gurney et al. (2001a,b). We implemented a three channel version of the model consisting of the basal ganglia 'direct pathway' (from NAccC to SNpr) and 'indirect pathway' (STN to SNpr; cf. Kandel et al., 2000). When active, the three channels activate respectively the 'turn-left', 'go-straight', and 'turn-right' routines (see Section 3.). As in the GPR model, the input to NAccC is amplified by DA:

$$\begin{aligned} \tau_{nacc} nacc_i = & -nacc_i + \\ & \sum_j [w_{bla_j \rightarrow nacc_i} \cdot bla_j] \cdot \\ & (bl_{nacc} + w_{da \rightarrow nacc} \cdot da) \end{aligned} \quad (5)$$

where bla_j is the j^{th} output unit of BLA and $w_{bla_j \rightarrow nacc_i}$ is its connection weight to $nacc_j$, bl_{nacc} and $w_{da \rightarrow nacc}$ are respectively a baseline and a multiplication coefficient of the amplification effects of DA on input.

Another important aspect of VBG is that the input signal it receives from BLA is affected by noise. This noise is generated in the form of a random number, uniformly drawn in $[0, 1]$ with a probability of 0.05 at each step of the simulation, added to each VBG input signal received by BLA.

The connections from BLA to NAccC are trained on the basis of an Hebb rule modulated by DA:

$$\begin{aligned} \Delta w_{bla_i \rightarrow nacc_j} = & \\ & \eta_{bla \rightarrow nacc} \cdot \\ & (\phi[da, \theta_{da}] \cdot (da - \theta_{da})) \cdot \\ & (\phi[nacc_i, \theta_{nacc}] \cdot nacc_j) \cdot bla_j \cdot \\ & (\theta_{bla \rightarrow nacc} - |w_{bla_i \rightarrow nacc_j}|) \end{aligned} \quad (6)$$

where $\eta_{bla \rightarrow nacc}$ is a learning rate, θ_{nacc} is a learning threshold for the activation of NAccC units, and $\theta_{bla \rightarrow nacc}$ is a threshold for saturating the weights. Note that in this learning rule the information related to $nacc_j$ should be brought to the NAccC units by ACC-NAccC backward connections not explicitly simulated in the model.

4.3 The Dopamine System

The dopaminergic activity in the model depends on the LH-VTA system. VTA is formed by one input and one output unit. The input unit is activated by CEA and inhibits the output unit. The output unit receives also an excitatory input from LH and produces as output the dopaminergic signals. Figure 3 shows an example of the overall functioning of VTA. The first graph of the figure shows the negative input received by the input unit from CEA. The

second graph shows the excitatory input received by the output unit from LH. The last two graphs show respectively the activation of the input and output units. It can be seen that the inhibition of the input unit (caused by CEA) can augment dopaminergic activity but never lead it over a certain threshold, e.g. necessary to trigger learning of the DA target areas. Similarly, an excitatory signal (from LH) to the output unit is not sufficient to lead DA level over the threshold when presented alone. This implies that both disinhibition and excitation are needed for the DA signal to trigger learning.

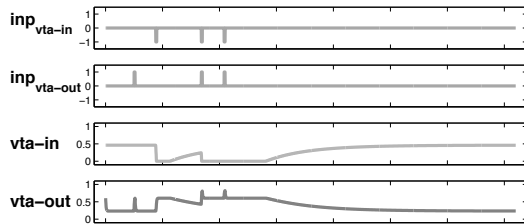


Figure 3: An ‘in-vitro’ test on the VTA responses.

5. Results

This section reports the outcome of the tests of the rat in the three learning/training phases described in Section 3. During the pre-training phase, the rat initially randomly explores the maze arm where it is by triggering sporadic actions under the effect of noise affecting NAcc. Motion is rather slow due to the low levels of DA. Eventually, this behaviour leads the rat to step on the food dispenser and eat the food (US). The resulting dopaminergic signal leads CEA to form associations between the seen-food units and the output unit triggering the tonic DA in VTA, and BLA to form associations between the seen-food units and the taste-food units. Learning of BLA and CEA leads the system to increase the frequency of selection of actions and the duration of their execution: overall the vigor of the rat seems increased when the rat sees the food. Figure 4 shows the activation of BLA caused by these learning processes. Notice how the activation of the CS units pre-activates the corresponding US units.

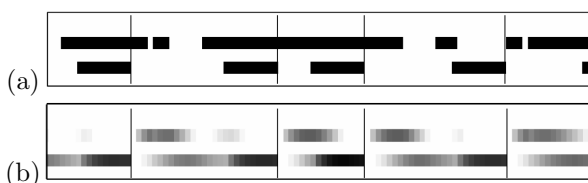


Figure 4: (a) Example of input stimuli during the pre-training phase (vertical bars mark different trials). (b) Corresponding activation of BLA units.

During the training phase, the rat initially explores the environment and speeds up its actions when the food becomes in sight. This leads it to rapidly approach the food dispenser while the coloured landmark of the arm is visible. Within CEA, this causes the formation of the associations between the units encoding the seen landmarks and the output unit. In parallel, BLA forms associations between units encoding the seen landmarks and units encoding the sight and the taste of foods. Figure 5 shows the connection weights formed during the pre-training and training phases. Notice how the system has formed positive connection weights from CS units to US units and negative weights in the opposite direction due to the differential Hebb learning rule.

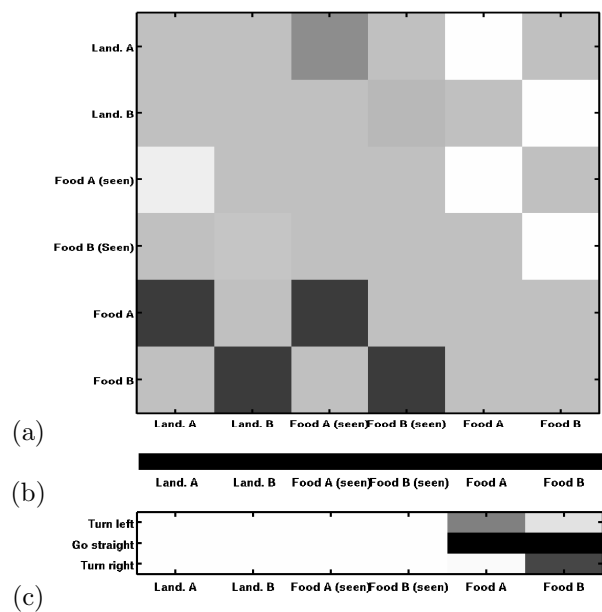


Figure 5: Connection weights after the pre-training and training phases (black = positive; white = negative, or zero for the BLA-NAccC connections). (a) BLA lateral-connection weights. (b) CEA connection weights. (c) BLA-NAccC connection weights.

Figure 6 and 7 show how in the devaluation test the rat exhibits a tendency to move with a higher frequency and vigor towards the non-devalued food and the corresponding landmark. Figure 8 shows the activations of the striatocortical system during the devaluation tests. Notice how NaccC, STN and ACC are biased toward the selection of the ‘go straight’ action when no food is satiated, whereas only vision of landmark A produces such bias when food B is satiated.

Interestingly, the intercalated neurons revealed important in this phase as they prevented the CEA from performing its non-selective effects on vigor (the CSs have access to the CEA output unit without being affected by satiety). Indeed, setting low values

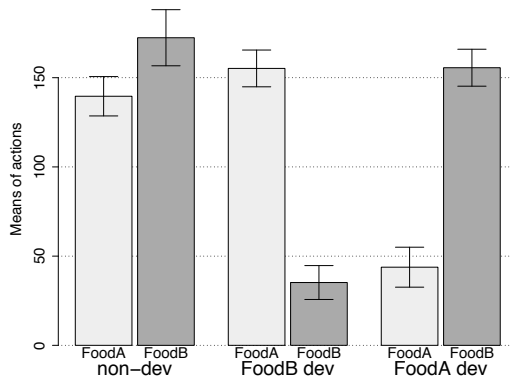


Figure 6: Number of contacts with the (empty) dispensers during the devaluation test in three conditions: no devaluation, food B devaluation, food A devaluation.

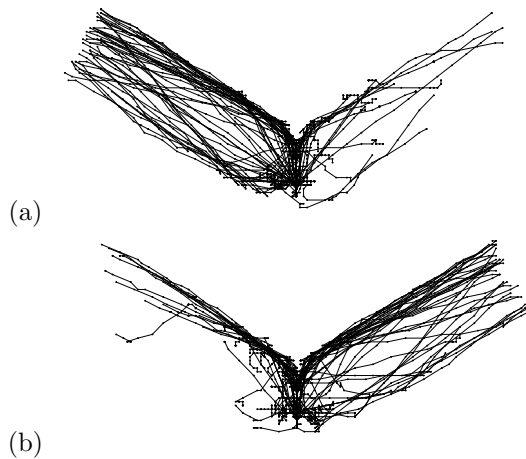


Figure 7: Paths followed by the rat during the test phases with food B devaluation (a) and food A devaluation (b).

of the inhibition exerted by these neurons on CEA produced less pronounced devaluation effects (data non reported).

6. Conclusions

This paper presented a bio-constrained model aiming at furnishing a coherent overall picture of Pavlovian mechanisms underlying navigation behaviours. The architecture and functioning of the model were designed by fulfilling a number of biological constraints related to: (a) the anatomy and Pavlovian associative processes of amygdala; (b) the anatomy and action-selection processes of nucleus accumbens; (c) the processes of hypothalamus and ventral tegmental controlling dopamine. The test of the model with autoshaping and devaluation experiments, run with a simulated rat, show that the behaviour exhibited by the model is comparable to that of real rats. These constraints and results render the model a neuroscientific and psychological operational theory furnish-

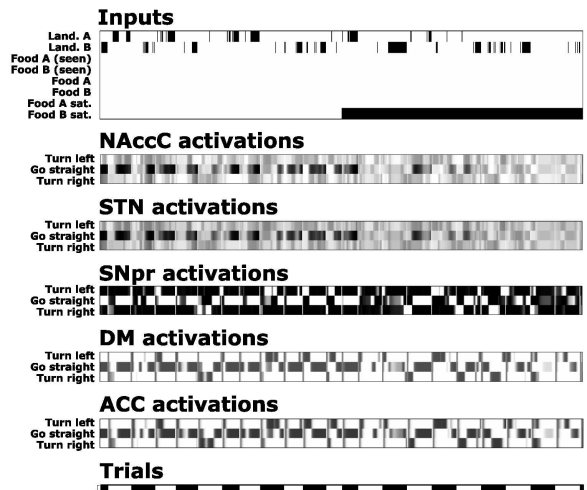


Figure 8: Activation of the striatocortical system units with no-devalued foods and food B devaluation.

ing a comprehensive picture of the Pavlovian mechanisms underlying navigation behaviours.

We believe the model is also very important for autonomous robotics for two reasons. The first is that it starts to investigate in detail how Pavlovian mechanisms might underly some navigation behaviours. This is important as, contrary to instrumental mechanisms usually used, Pavlovian mechanisms render such navigation behaviors (a) *fast learnable*, as Pavlovian association mechanisms allow complex ‘approach target’ behavioural routines to be quickly associated with new targets, and (b) *flexible*, as the triggering of such routines can be dynamically controlled by the internal states of robots. The second is that the Pavlovian processes investigated with the model have a paramount importance for several other cognitive and learning processes (Mirolli et al., sub).

Future work will further refine the model by aiming to account for all the biological constraints and behavioural evidence reported in Section 2.

Acknowledgements

This research was supported by the EU funded Project ‘ICEA – Integrating Cognition, Emotion and Autonomy’, contract no. FP6-IST-027819-IP.

References

- Amari, S. (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biol Cybern*, 27(2):77–87.
- Arleo, A. and Gerstner, W. (2000). Spatial cognition and neuro-mimetic navigation: A model of hip-

- pocampal place cell activity. *Biol Cybern*, 83:287–299.
- Blundell, P., Hall, G., and Killcross, S. (2003). Preserved sensitivity to outcome value after lesions of the basolateral amygdala. *J Neurosci*, 23(20):7702–7709.
- Cardinal, R. N., Parkinson, J. A., Hall, J., and Everitt, B. J. (2002a). Emotion and motivation: the role of the amygdala, ventral striatum, and prefrontal cortex. *Neurosci Biobehav Rev*, 26(3):321–352.
- Cardinal, R. N., Parkinson, J. A., Lachenal, G., Halkerston, K. M., Rudarakanchana, N., Hall, J., Morrison, C. H., Howes, S. R., Robbins, T. W., and Everitt, B. J. (2002b). Effects of selective excitotoxic lesions of the nucleus accumbens core, anterior cingulate cortex, and central nucleus of the amygdala on autoshaping performance in rats. *Behav Neurosci*, 116(4):553–567.
- Cardinal, R. N., Parkinson, J. A., Marbini, H. D., Toner, A. J., Bussey, T. J., Robbins, T. W., and Everitt, B. J. (2003). Role of the anterior cingulate cortex in the control over behavior by Pavlovian conditioned stimuli in rats. *Behav Neurosci*, 117(3):566–587.
- Domjan, M. (2006). *Principles of Learning and Behaviour*. Thomson Wadsworth, Belmont, CA.
- Filliat, D. and Meyer, J.-A. (2003a). Map-based navigation in mobile robots - I. A review of localisation strategies. *J Cog Sys Res*, 4(4):243–282.
- Filliat, D. and Meyer, J.-A. (2003b). Map-based navigation in mobile robots - II. A review of map-learning and path-planning strategies. *J Cog Sys Res*, 4(4):283–317.
- Gurney, K., Prescott, T. J., and Redgrave, P. (2001a). A computational model of action selection in the basal ganglia. I. A new functional anatomy. *Biol Cybern*, 84(6):401–410.
- Gurney, K., Prescott, T. J., and Redgrave, P. (2001b). A computational model of action selection in the basal ganglia. II. Analysis and simulation of behaviour. *Biol Cybern*, 84(6):411–423.
- Kandel, E. R., Schwartz, J. H., and Jessel, T. M. (2000). *Principles of Neural Science*. McGraw-Hill, New York.
- Knapska, E., Radwanska, K., Werka, T., and Kaczmarek, L. (2007). Functional internal complexity of amygdala: focus on gene activity mapping after behavioral training and drugs of abuse. *Physiol Rev*, 87(4):1113–1173.
- Lieberman, D. A. (1993). *Behavior and Cognition*. Brooks/Cole, Pacific Grove, CA.
- Mannella, F., Mirolli, M., and Baldassarre, G. (2007). The role of amygdala in devaluation: a model tested with a simulated robot. In *Proceedings of the Seventh International Conference on Epigenetic Robotics*, pages 77–84. Lund University Cognitive Studies, Lund.
- Martinet, L., Sheynikhovich, D., Meyer, J.-A., and Arleo, A. (inpr). A cortical column model for studying spatial navigation planning. *Neurocomp*.
- McDonald, A. J. (1998). Cortical pathways to the mammalian amygdala. *Prog Neurobiol*, 55(3):257–332.
- Mirolli, M., Mannella, F., and Baldassarre, G. (sub). The roles of amygdala in the affective regulation of body, brain and behaviour. *Connec Sci*.
- Niv, Y., Daw, N., Joel, D., and Dayan, P. (2006). Tonic dopamine: opportunity costs and the control of response vigor. *Psychopharmacology (Berl)*.
- O’Keefe, J., Burgess, N., Donnett, J. G., Jeffery, K. J., and Maguire, E. A. (1998). Place cells, navigational accuracy, and the human hippocampus. *Philosophical Transactions of the Royal Society of London - B*, 353(1373):1333–1340.
- Paré, D., Quirk, G. J., and Ledoux, J. E. (2004). New vistas on amygdala networks in conditioned fear. *J Neurophysiol*, 92(1):1–9.
- Parkinson, J. A., Robbins, T. W., and Everitt, B. J. (2000). Dissociable roles of the central and basolateral amygdala in appetitive emotional learning. *Eur J Neurosci*, 12(1):405–413.
- Pitkänen, A., Jolkkonen, E., and Kempainen, S. (2000). Anatomic heterogeneity of the rat amygdaloid complex. *Folia Morphol*, 59(1):1–23.
- Porr, B. and Wörgötter, F. (2003). Isotropic sequence order learning. *Neural Computation*, 15:831–864.
- Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. The MIT Press, Cambridge MA.
- Trullier, O., Wiener, S. I., Berthoz, A., and Meyer, J.-A. (1997). Biologically based artificial navigation systems: review and prospects. *Progr Neurobiol*, 51(5):483–544.

Evaluating Intrinsically Motivated Robots using Affordances and Point-Cloud Matrices

Kathryn Merrick,
University of New South Wales
Australian Defence Force Academy
School of Engineering and Information Technology
k.merrick@adfa.edu.au

Abstract

A key challenge developing intrinsically motivated robots is evaluation of the robots' emergent behaviour. Evaluation techniques for intrinsically motivated robots must be open-ended enough to identify any emergent behaviours, but specific enough to quantify those behaviours in a meaningful way. This paper describes a novel use of point-cloud matrices for detecting cycles of affordances in robots' behaviour. The technique is demonstrated by evaluating two motivated reinforcement learning algorithms on four *Lego Mindstorms NXT* critter-bots. Results show that the evaluation technique can identify changing attention focus, periods of exploration and exploitation and repetitive, cyclic behaviour.

1. Introduction

Intrinsically motivated robots are characterised by their ability to select their own goals. They use an embedded computational model of motivation – such as novelty (Huang and Weng, 2007), interest (Merrick and Huntingon, 2008) or curiosity (Oudeyer et al., 2007) – to select salient environmental stimuli on which to focus their attention. The capacity for autonomous, open-ended goal-selection gives intrinsically motivated robots the potential to adapt to unexpected changes in their environment. In addition, they can develop novel or creative behaviours that were not explicitly programmed by engineers.

However, a key challenge developing intrinsically motivated robots is the evaluation of a robot's emergent behaviour. Evaluation techniques for intrinsically motivated robots must be open-ended enough to identify any emergent behaviours, but specific enough to quantify those behaviours in a meaningful way. Evaluation is difficult for intrinsically motivated robots because these robots can select and change their own goals. This means that traditional, task-oriented evaluation is inappropriate as there is no fixed set of 'correct' tasks to be addressed.

This paper adapts a technique used to evaluate repetitive patterns in human motion for use with robots. Point-cloud matrices are used to visualise cycles of affordances acted on by a robot. Section 2 begins with a brief survey of techniques for evaluating the behaviour of intrinsically motivated robots and natural systems. Section 3 describes how point-cloud matrices and affordances can be used to evaluate robots' behaviour. Section 4 demonstrates the technique by evaluating two motivated reinforcement learning algorithms on four critter-bots using the *Lego Mindstorms NXT* platform. Results show that the evaluation techniques can identify changing attention focus, periods of exploration and exploitation and repetitive, cyclic behaviour learned by a robot.

2. Evaluating Intrinsically Motivated Robots

Various techniques have been used to evaluate intrinsically motivated robots. For example, Oudeyer et al. (2007), use the idea of 'affordant' and 'non-affordant' behaviour for a particular task to evaluate an intrinsic motivation system for autonomous mental development in a robot. This allows them to evaluate the success of their system in terms of the increase in affordant behaviours and the decrease in non-affordance behaviours for a particular task.

Other approaches to the evaluation of intrinsically motivated robots include algorithm specific approaches (Huang and Weng, 2007), case studies and bifurcation diagrams (Merrick and Huntingon, 2008). In contrast, this paper presents a general approach in which the performance of a robot is characterised in terms of its ability to act in structured, cyclic patterns. This allows evaluation of the emergent behaviour of a robot, independent of a specific task or controlling algorithm.

The importance of cyclic behaviour in natural systems such as animals has been identified by biologists (Ahlgren and Halberg, 1990; Dunlap et al., 2003).

Common examples include the circadian rhythm, migratory cycles, and cycles associated with seasons or tides. A number of techniques for identifying repetitive cyclic patterns in human motion have been proposed (Li and Holstein, 2002; Kovar and Gleicher, 2004; Forbes and Fiume, 2005; Tang et al., 2008). In this paper we adapt the point-cloud technique proposed by Tang et al. (2008) to identify patterns in robot motion.

2.1 Point-Cloud Matrices

Tang et al. (2008) use point-cloud matrices to visualise posture similarity in motion-capture data, as shown in Figure 1. Diagonal patterns represent cycles of repeated postures. Cycles can be either continuous or distributed. Continuous cycles, represented by bottom-left to top-right diagonals, repeat a sequence of postures at adjacent time periods. Distributed cycles, represented by cross patterns, repeat a sequence of postures at intermittent time periods.

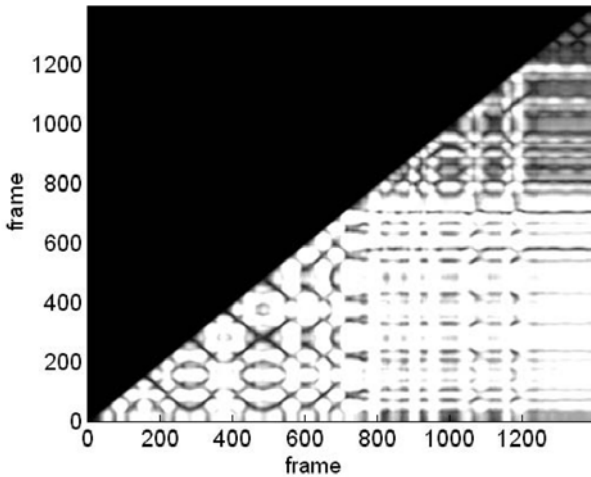


Figure 1. Tang et al. (2008) visualise human motion capture data using a point-cloud matrix showing posture similarity. Diagonal patterns represent cyclic behaviour.

Tang et al. (2008) use point-cloud matrices to visualise posture similarity for human dancers. Because we are often interested not only in the posture of a robot, but also the actions it performs, this paper proposes an alternative technique where affordance similarity rather than posture similarity is visualised.

2.2 Affordances

The concept of affordances is generally attributed to Gibson (1979) as an approach to understanding visual perception in natural systems. His theory is that organisms perceive their environment, or objects in their environment, in terms of the opportunities those objects provide for the organism to act. Thus affordances capture both the state of an environment and the actions available to an organism in that state.

While there is no universal definition or notation for affordances in robotics, the concept has been considered as an approach to a range of robotic problems (Rome et al., 2008a; Rome et al., 2008b). These include tool-use (Stoytchev, 2005), interaction (Hafner and Kaplan, 2008), machine vision (Paletta and Fritz, 2008; Modayil and Kuipers, 2008) and navigation (Modayil and Kuipers, 2008; Hertzberg et al., 2008). This paper extends existing work with affordances in robotics to the challenge of evaluating the behaviour of intrinsically motivated robots.

3. Cyclic Evaluation of Robot Behaviour using Affordances and Point-Cloud Matrices

Affordances can be thought of as mappings or relationships between some aspect of a robot's environment – such as an object (Stoytchev, 2005; Modayil and Kuipers, 2008) or another agent (Hafner and Kaplan, 2008) and the actions a robot can perform. This paper focuses on the relationship between the actions a robot can perform, its physical structure and its external environment. The total environment of a robot is considered to comprise data describing both its internal and external state. For example, a *Lego* robot such as the one shown in Figure 3(b) may be described by the internal state of its motor (on/off, power level etc.) and by the state of its external environment detected by its colour sensor (red level, green level, blue level etc.). More formally, a robot's state $S_{(t)}$ at time t is described by its internal state $S_{I(t)}$ and its external state $S_{E(t)}$:

$$S_{(t)} = S_{I(t)} + S_{E(t)}$$

An attribute-based representation $S = (s_1, s_2, s_3, \dots)$ is required for application of the technique in this paper. The set \mathbf{A} of actions afforded by a state S at time t is:

$$F(S_{(t)}) = \mathbf{A}$$

This notation implies that actions afforded by a state are determined by both the state itself and the time at which the state occurs.

While a robot may perceive, or learn to perceive, a number of affordances in any state, its emergent behaviour is defined by the affordance it chooses to act on or execute at each time-step. We denote the affordance executed at time t by:

$$X_{(t)} = \{S, A\}$$

where A is a numeric action identifier.

The point-cloud visualization is constructed by computing the Euclidean distance $\text{dist}(X_{(t)}, X_{(t')})$ between pairs of affordances at all times t and t' . The intensity of a pixel (t, t') on the point-cloud diagram is

determined by $\text{dist}(X_{(t)}, X_{(t')})$. A darker colour indicates more similar affordances.

Because small robots such as *Lego Mindstorms* tend to move faster than humans, the point-cloud visualisations tend to be denser and have various different characteristic patterns, in addition to those seen in the human motion plots. A number of important characteristic patterns are identified in the sample diagram in Figure 2. Their meanings are discussed in the following sections.

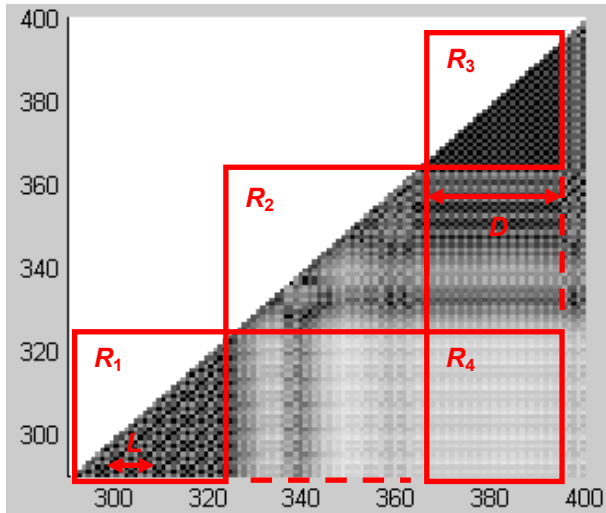


Figure 2. Characteristic patterns on point-cloud diagrams of robotic motion. See text for detailed description.

3.1 Cyclic Behaviour in Robots

In Figure 2, the dark triangles of parallel diagonals in the highlighted regions R_1 and R_3 are cycles of affordances. The distance L between diagonals within a triangle indicates the cycle length. The length D of the side of a triangle indicates the cycle duration. Duration divided by cycle length gives the number of repetitions of a cycle. R_1 shows a distributed cycle while R_3 shows a continuous cycle.

Continuous Cycles

Continuous cycles repeat the same sequence of affordances at adjacent time periods. For example:

$$\boxed{X_1, X_2, X_3}, \boxed{X_1, X_2, X_3}, \boxed{X_1, X_2, X_3}, \boxed{X_1, X_2, X_3} \dots$$

Continuous cycles appear as parallel diagonals on a point-cloud matrix, such as that shown in R_3 .

Distributed Cycles

Distributed cycles repeat two or more sequence of affordances at intermittent time periods. The following example interleaves two sequences:

$$\boxed{X_1, X_2, X_3}, \boxed{X_3, X_2, X_1}, \boxed{X_1, X_2, X_3}, \boxed{X_3, X_2, X_1} \dots$$

Distributed cycles appear as cross patterns on a point-cloud matrix, such as that shown in R_1 .

3.2 Exploration versus Exploitation

Intrinsically motivated systems, both natural and artificial, must exhibit periods of both explorative and exploitative behaviour. Exploration is required to find new, motivating things to learn about. Exploitation is required to carry out learned behaviours. In Figure 2, the structured patterns in regions R_1 and R_3 indicate that the robot is exploiting a learned cycle. In contrast, the unstructured, random pattern in R_2 indicates that the robot is exploring to find something new to learn about.

3.3 Attention Focus

As a robot explores, its focus of attention shifts. The robot may focus on exploiting entirely new behaviours or return its focus to a previous behaviour. The colour of the rectangular regions linking dark triangles can be used to identify these different types of shifts in attention focus. For example, the light rectangular region R_4 in Figure 2 indicates that the affordances in R_1 are generally dissimilar to those in R_3 . If R_4 were dark in colour it would indicate that the affordances in R_1 and R_3 were similar.

4. Demonstration

This section demonstrates the evaluation technique by evaluating two motivated reinforcement learning (MRL) algorithms on four *Lego Mindstorms NXT* critter-bots, shown in Figure 3.

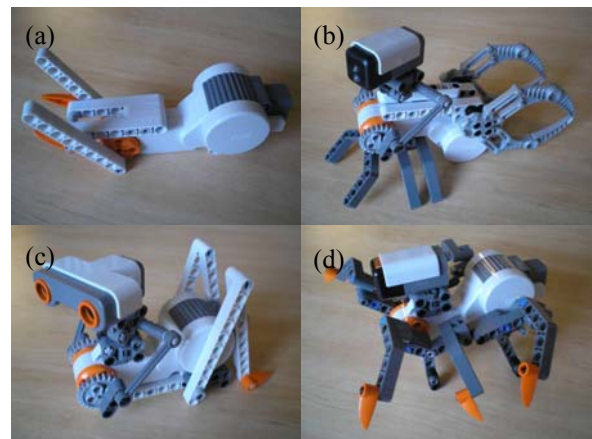


Figure 3. Four critter-bots: (a) a snail with a single motor; (b) a bee with a motor and colour sensor; (c) a cricket with a motor and ultrasonic sensor; (d) an ant with a motor and accelerometer.

The first algorithm, called MRL, is a table-based approach (Merrick and Maher, 2009). The second algorithm, called SART-MRL uses a function approximation technique based on simplified adaptive

resonance theory (SART) networks (Baraldi and Alpaydin, 1998) to generalise over the robot’s state space. It is hypothesised that the latter approach will be able to learn more effectively and exhibit more structured behaviour on the robots, because of its ability to generalise over the noisy state space of the robots. The evaluation technique should thus reflect this hypothesis by revealing the presence of structured behaviour cycles by the robots using SART-MRL.

The two algorithms – MRL and SART-MRL – were each run for 1,200 time-steps (approximately six minutes) on each critter-bot. The following paragraphs describe the state and action spaces of each robot and some of the emergent behaviours with reference to point-cloud diagrams for each run.

4.1 The Snail

The first critter-bot, shown in Figure 3(a), is a snail with a single motor controlling the height of its antennae. The snail can sense the rotation of the motor from its built-in tachometer, and whether the motor is moving or not. The tachometer reading is an angle between -360° and 360° from the initial position of the motor. The movement reading is enumerated such that 0 means the motor is stopped, 200 means the motor is moving forwards and 100 means it is moving backwards.

Every state encountered by the snail affords three actions: A_1 – move the motor forward at a fixed speed; A_2 – move the motor backwards at a fixed speed; A_3 – stop the motor. The control algorithms respond to an intrinsic motivation function to learn which actions to select in each state.

Figure 4 visualises the behaviour of the snails using each algorithm. Figure 4(a) shows the MRL algorithm and Figure 4(b) the SART-MRL algorithm. The white rectangular regions in Figure 4(a) indicate that this robot is focusing attention on different affordances at different times. Inspection of the log file for this critter-bot shows that it is focusing on affordances in states with positive tachometer readings until approximately $t = 550$, and states with negative tachometer readings from $t = 550$ -850. However, while some ability to focus attention is evident in the snail using MRL, zooming in on the darker triangular region (Figure 5) reveals an absence of diagonals. This indicates that structured, cyclic behaviour is not occurring. Rather the robot is continually exploring in an effort to find a region of the environment in which it can learn.

In contrast, zooming in on some of the dark triangles for the snail using SART-MRL shows a number of different diagonal patterns. Figure 6 shows two such patterns. The first pattern in R_1 is a continuous cycle of length $L=5$ and duration $D=30$. Inspection of the log

file for this robot shows that the cycle is repeating actions: $A_1 A_1 A_2 A_3 A_2 \dots$. This corresponds to the snail raising its antennae twice then lowering them twice. The second pattern in R_2 has $L=2$ and $D=49$ and repeats actions: $A_1 A_2$. This corresponds to the snail raising its antennae once then lowering them once. Periods of exploration are visible before R_1 , between the exploitative behaviour in R_1 and R_2 and after R_2 .

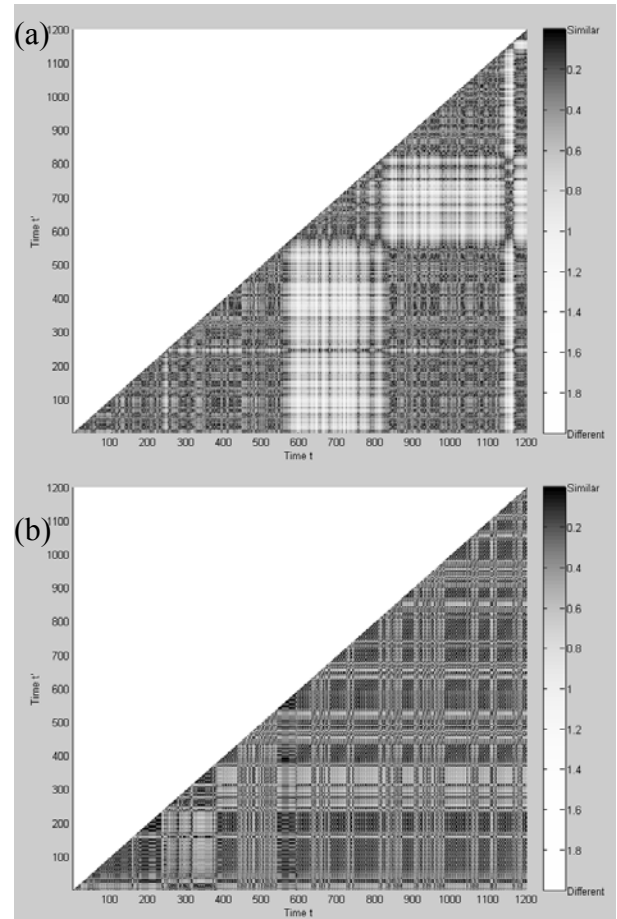


Figure 4. Point cloud visualisations for the snails using (a) MRL and (b) SART-MRL

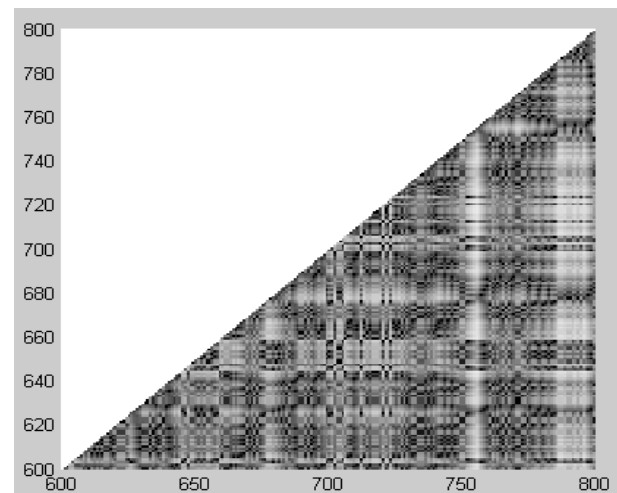


Figure 5. Zoomed region of Figure 4(a). No structured behaviour cycles (dark diagonals) are evident.

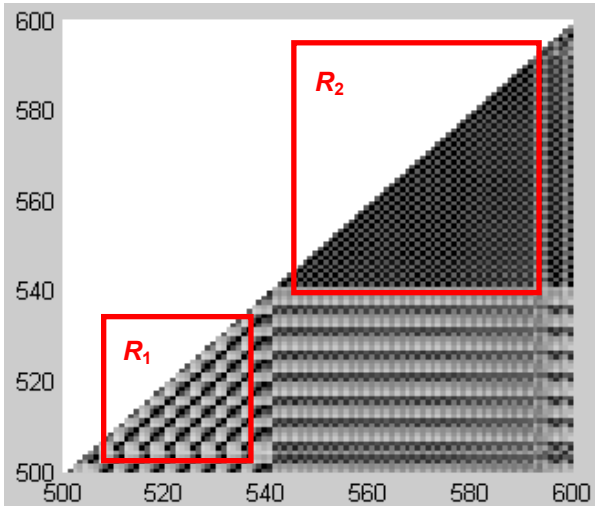


Figure 6. Zoomed region of Figure 4(b). Two structured behaviour cycles are evident as diagonal patterns in regions R_1 and R_2 . Exploratory behaviour is evident before, between and after these regions.

One of the weaknesses of using point-cloud diagrams to analyse robots is that only short time periods can reasonably be displayed on a screen or page. This is addressed in this paper by zooming in on regions of interest. One direction for future work is automated analysis of the point-cloud diagrams to identify such regions of interest and to generate statistical data about the long-term characteristic behaviour of the robot. This is discussed further in Section 5.

4.2 The Bee

The second critter-bot, shown in Figure 3(b), is a bee with a motor and colour sensor. The motor allows the bee to turn its colour sensor ‘head’ through 45° to both the left and right. The bee can sense the rotation of the motor and whether the motor is moving or not. The colour sensor provides data describing red, blue and green intensities of the critter’s environment in the direction the colour sensor is pointing. These readings range between 0 and 255. The bee was placed between two colour panels, one red and the other green.

As for the snail, every state encountered by the bee affords three actions: A_1 – move the motor forward at a fixed speed; A_2 – move the motor backwards at a fixed speed; A_3 – stop the motor.

Figure 7 shows the point-cloud diagrams for the bees. As with the snail using MRL, Figure 7(a) again shows the characteristic patterns of shifting attention focus. Also like the snail, however, this plot shows little structured, cyclic behaviour emerging in the bee using MRL. This is apparent from the light overall colour of the point-cloud diagram, which indicates fewer matching or similar affordances were executed. The reason for the reduction in emergent structured behaviour is the bee’s colour sensor. The colour sensor

returns particularly noisy readings depending on the distance of the robot to the coloured object being sensed and other factors such as ambient light.

The much darker triangles in Figure 7(b) indicate that some structured behaviour is occurring in the bee running SART-MRL. Figure 8 zooms in on two of these triangles, which show clear diagonal patterns. The first in R_1 represents a distributed cycle in which the bee repeatedly turns its head between the red panel and the neutral region between the panels. This cycle has $L=6$ and $D=26$ and repeats actions: $A_3 A_1 A_2 A_1 A_3 A_2 \dots$. This represents the bee experimenting with its colour sensor as it alternates between high and low red intensity readings.

Between R_1 and R_2 is a period of exploration as the robot seeks different motivating stimuli. The mid-range grey colour of the linking square region indicates that some of the affordances in R_2 are similar to those in R_1 .

The cycle in R_2 is a continuous cycle in which the bee is experimenting with its motor in the neutral space between the colour panels. In this space the red and green intensities are both low or zero. This cycle has length $L=2$, duration $D=30$ and repeats actions: $A_1 A_2$.

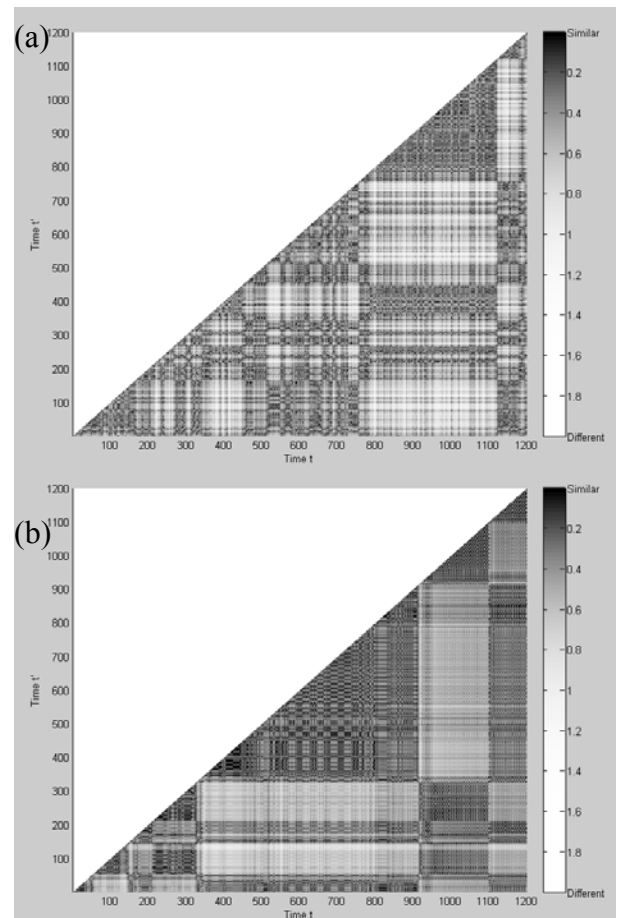


Figure 7. Point cloud visualisations for the bees using (a) MRL and (b) SART-MRL

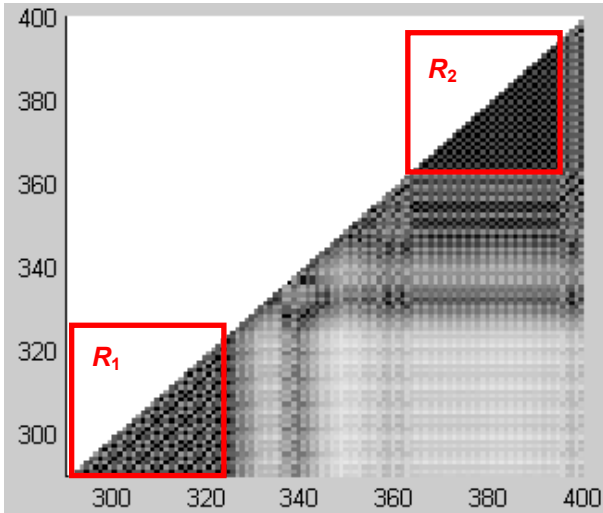


Figure 8. Zoomed region of Figure 7(b). Two periods of exploitation are evident, separated by exploration.

4.3 The Cricket

Figure 3(c) shows the third critter-bot, a cricket, with a motor and ultrasonic (distance) sensor. As with the bee, the motor allows the cricket to turn its ultrasonic sensor ‘head’ through 45° to both the left and right. The cricket can sense the rotation of the motor and whether the motor is moving or not. The ultrasonic sensor provides eight ping values describing the distance of any object in the direction the ultrasonic sensor is pointing. The cricket was placed in a corner such that it was further from one wall than from the other.

Once again, every state encountered by the cricket affords three actions: A_1 – move the motor forward at a fixed speed; A_2 – move the motor backwards at a fixed speed; A_3 – stop the motor.

Figure 9 shows the point-cloud diagrams for the two crickets. Once again, the cricket using MRL (Figure 9(a)) shows little structured behaviour. This is evident from the light overall colour of the diagram. In addition, there is no clear shift in attention focus over the course of the cricket’s life. This is indicated by the absence of light coloured rectangular regions or darker triangular regions.

In contrast to Figure 9(a), the point-cloud diagram for the cricket using SART-MRL (Figure 9(b)) does have the characteristics of shifting attention focus, with a number of light-coloured square regions evident. In addition, zooming in on dark triangular regions, such as that in Figure 10, shows the characteristic diagonal patterns of structured behaviour cycles. Figure 10 shows a continuous cycle with $L=3$ and $D=153$, using actions: $A_1 A_2 A_3 \dots$. This represents the cricket experimenting with its motor settings. Around $t=590$ the robot begins to explore once more.

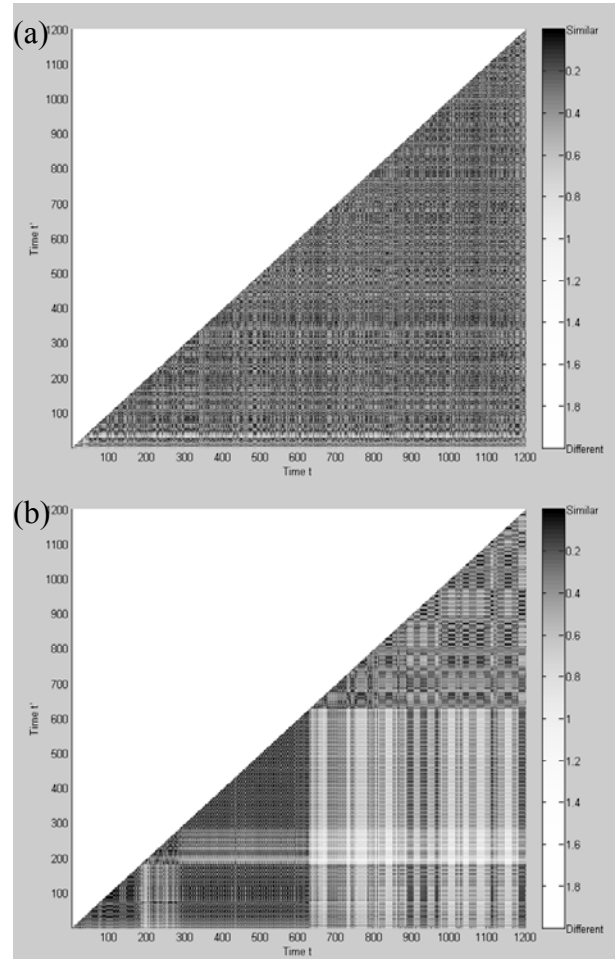


Figure 9. Point cloud visualisations for the crickets using (a) MRL and (b) SART-MRL

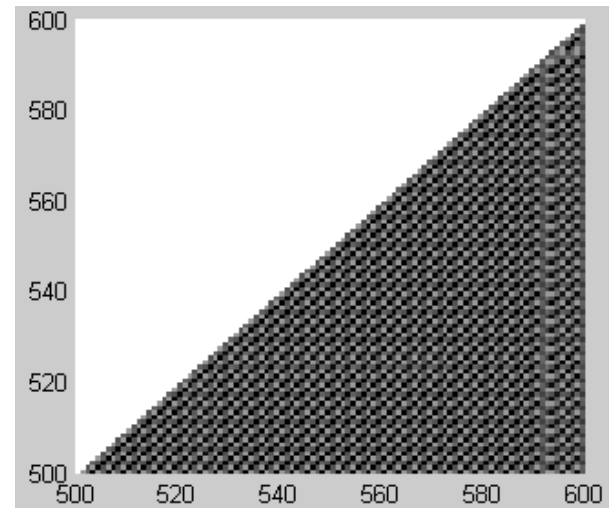


Figure 10. Zoomed region of Figure 9(b). One cycle is evident, followed by exploration from $t=590$.

4.4 The Ant

Finally, the fourth critter-bot in Figure 3(d) is an ant with a motor and accelerometer. The motor moves the ant’s legs, which can grip the surface it is on and propel

the robot forwards or backwards. The ant can sense whether the motor is moving or not. In addition it can sense six values from the accelerometer. Three of these describe its acceleration in three dimensions. These values range between 0 and 981. The other three values describe the bot's tilt from the horizontal in the same dimensions. These values range from 0 to 254. Every state encountered by the ant affords three actions: A_1 – move the motor forward at a fixed speed; A_2 – move the motor backwards at a fixed speed; A_3 – stop the motor.

Figure 11 shows the point-cloud diagrams for the ants using MRL and SART-MRL. This is the noisiest of the applications as accelerometer readings are affected by the rocking motion of the ant as it moves. This is influenced by gravity and, to a lesser extent, the wires attaching the robot to the intelligent brick. Like the cricket using MRL, Figure 11(a) shows that the ant using MRL exhibits little or no structured behaviour cycles and has little change in attention focus. This is evidenced by the light, even colours on the point-cloud diagram.

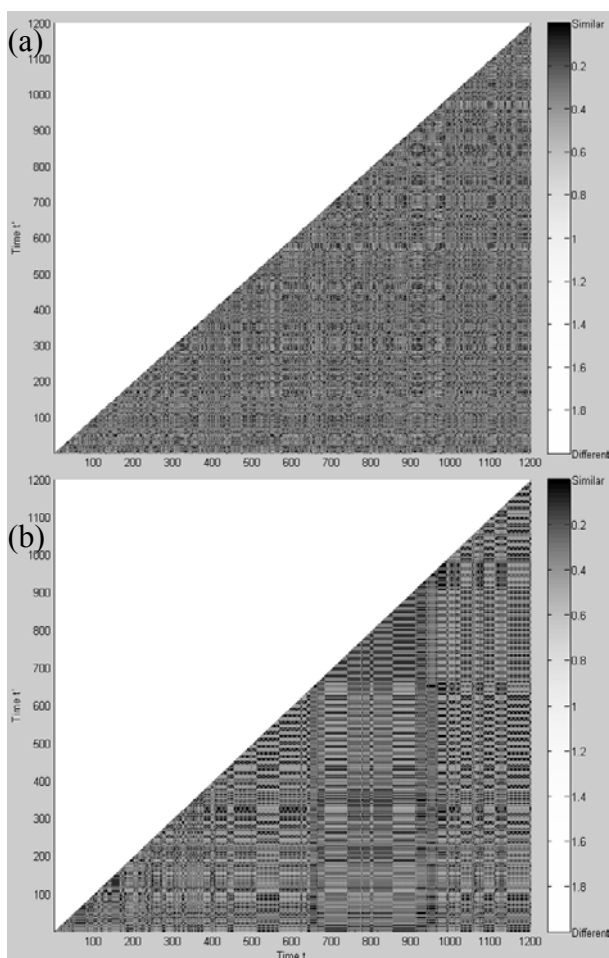


Figure 11. Point cloud visualisations for the ants using (a) MRL and (b) SART-MRL

Figure 11(b) for the ant using SART-MRL also shows relatively mid-range greys, although more triangle patterns are evident in the diagram. Zooming in on parts of the plot, such as in Figure 12, shows that structured behaviour is evident, but the characteristic diagonal patterns are much noisier. This mirrors the fact that the state space for this robot is also much noisier. Figure 12 in fact shows a ‘walking’ behaviour learned by the ant. The walk was somewhat jerky, with the ant learning to combine a sequence of ‘move-forward’ and ‘stop-motor’ actions. Despite this, the structured behaviour was evident both visually when the robot was learning and in the point-cloud diagram. One of the strengths of the point-cloud visualisations is that they can reveal quite noisy, yet still structured, behaviour that would be difficult to identify by analysing the data numerically.

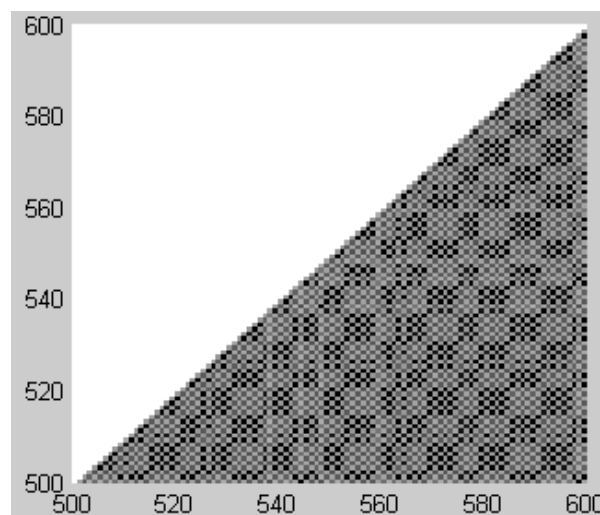


Figure 12. Zoomed region of Figure 11(b). A noisy yet still structured behaviour cycle is evident. This behaviour cycle was the robot ‘walking’.

5. Conclusion and Future Work

This paper has presented a novel use of point-cloud matrices and affordances for evaluating intrinsically motivated robots. A demonstration was presented of the evaluation model on two motivated reinforcement learning approaches on four critter-bots using the *Lego Mindstorms NXT* platform. Results show that the evaluation technique can distinguish:

- Changing attention focus by a robot – visible as light coloured, rectangular linking regions;
- Periods of exploration – visible as random patterns;
- Periods of exploitative cyclic behaviour – visible as dark, triangular patterns of diagonals.

In addition the length and duration of cycles can be computed from the diagrams. These results qualitatively confirmed the hypothesis that the SART-MRL control algorithm would exhibit more structured

behaviour and greater ability to focus attention than the MRL control algorithm.

While the model in this paper does not seek to evaluate the ‘intelligence’, ‘usefulness’ or ‘correctness’ of a robot’s behaviour, it provides an approach that can be used in conjunction with domain specific case studies or other metrics to identify the emergence of structured, cyclic patterns characteristic of learning.

The next phase of this work will focus on developing an automated, numerical analysis of the point-cloud diagrams to permit a quantitative evaluation of the behaviour of a robot. This will complement the visualisations to assist with identifying regions of interest and provide a way to compare the behaviour of different robots numerically. The numerical analysis might include automatically identifying properties such as the number, length and duration of behaviour cycles. This work will further permit the design and analysis of more complex motivated robots running for longer time periods in complex environments.

References

- A. Ahlgren and F. Halberg. *Cycles of nature: an introduction to biological rhythms*. Washington DC: National teachers association, 1990.
- A Baraldi and E. Alpaydin. Simplified ART: a new class of ART algorithms. *International Computer Science Institute*, Technical Report TR 98-004, Berkley, CA, 1998
- J. Dunlap, J. Loros and P. DeCoursey. *Chronobiology: biological timekeeping*. Sinauer Associates, 2003.
- K. Forbes and E. Fiume. An efficient search algorithm for motion data using weighted PCA. *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. pp 67-76, CA, 2005.
- J. Gibson. *The ecological approach to visual perception*. Boston, MA, Houghton Mifflin, 1979.
- V. Hafner and F. Kaplan. Interpersonal maps: how to map affordances for interaction behaviour. *Towards affordance-based robot control*. In *Lecture Notes in Computer Science*, J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008.
- J. Hertzbert, K. Lingemann, C. Lorken, A. Nuchter and S. Stiene. Does it help a robot navigate to call navigability an affordance? *Towards affordance-based robot control*. In *Lecture Notes in Computer Science*, J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008.
- X. Huang and J. Weng. Inherent value systems for autonomous mental development. *International Journal of Humanoid Robotics*, 4(2):407-433, 2007.
- D. A. Kolb, I.M. Rubin, and J.M. McIntyre, (Eds) *Organizational Psychology: Readings on Human Behaviour in Organizations*. Prentice-Hall, Englewood Cliffs, NJ, 1984.
- L. Kovar and M. Gleicher. Automated extraction and parametrization of motions in large data sets. *Proceedings of ACM SIGGRAPH 2004 Papers*, pp 559-568, Los Angeles, CA, August, 2004.
- B. Li and H. Holstein. Recognition of human periodic motion – a frequency domain approach. *Proceedings of the 16th International Conference on Pattern Recognition*, pp 311-314, Washington DC.
- K. Merrick and E. Huntington. Attention focus in Curious, Reconfigurable Robots. *Proceedings of the 2008 Australian Conference on Robotics and Automation*. ANU, Canberra, Australia, (CD: no page numbers), December 2008.
- K. Merrick, M. L. Maher, *Motivated Reinforcement Learning Agents: Curious Characters for Multiuser Games*, Springer-Verlag, Berlin/Heidelberg, ISBN 978-3-540-89186-4, 2009
- J. Modayil and B. Kuipers. The initial development of object knowledge by a learning robot. *Robotics and Autonomous Systems*. 56:879-890, Elsevier, 2008.
- P.-Y. Oudeyer, F. Kaplan and V. Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*. 11(2):265-286, April 2007.
- L. Paletta and G. Fritz. Reinforcement Learning of Predictive Features in Affordance Perception. *Towards affordance-based robot control*. In *Lecture Notes in Computer Science*, J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008.
- E. Rome, J. Hertzberg and G. Dorffner. *Towards affordance-based robot control*. In *Lecture Notes in Computer Science*, J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008a.
- E. Rome, L. Paletta, E. Sahin, G. Dorffner, J. Herzberg, R. Breithaupt, G. Fritz, J. Irran, F. Kintzler, C. Lorken, S. May and E. Ugur. The MACS project: an approach to affordance-inspired robot control. *Towards affordance-based robot control*. In *Lecture Notes in Computer Science*, J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008b.
- A. Stoytchev. Behaviour-grounded representation of tool affordances. *Proceedings of the IEEE International Conference on Robotics and Automation*, pp 3060-3065, Barcelona, Spain, 2005.
- K.-T. Tang, H. Leung, T. Komura, H. Shum. Finding repetitive patterns in 3D human motion captured data. *Proceedings of the Second International Conference on Ubiquitous Information Management and Communication*. pp 396-403, Suwon, Korea, 2008.

An Unsupervised Model of Infant Acoustic Speech Segmentation

Matthew Miller and Alexander Stoytchev
Developmental Robotics Laboratory
Iowa State University
{mamille|alexs}@iastate.edu

Abstract

There is a long standing hypothesis in Developmental Psychology that children use statistical information to segment acoustic speech streams into words. Additionally, several experiments have demonstrated that infants are able to find word breaks using distributional cues. In this paper we propose an algorithm for the unsupervised segmentation of audio speech, based on the Voting Experts (*VE*) algorithm. We show that this algorithm can reproduce results obtained from segmentation experiments performed with 8-month-old infants.

1. Introduction

Spoken human language contains no analogue to the spaces placed between written words. The pauses that do exist in audio speech appear between phrases, when the speaker takes a breath, or when the airflow is stopped in the pronunciation of certain consonants. The sounds that are separated by these pauses are rarely composed of a single word, and there are no universal markers to indicate where those single words might be (Klatt, 1979). However, when we hear our native language, we hear discrete words. We unconsciously break the stream into its constituents, rendering it comprehensible. This is possible because we know the language, and are familiar with the large lexicon of words we might expect to hear. When confronted with a novel word, we need only segment the words before and after it to identify it as a brand new token.

Infants, however, do not share this luxury. They must learn to segment their mother's tongue from scratch. Every word is a novel word, and their lexicon starts off empty. Fortunately, human beings have an apparently innate ability to use statistical information to segment continuous spoken speech into

words, and that ability is present in infants as young as 8 months old. Apparently, they can perform this task without any feedback or other salient cues as to the locations of word breaks (Saffran et al., 1996) (Saffran et al., 1999).

An accurate characterization of this ability would presumably be theoretically and practically advantageous. Along those lines, this paper proposes a method for the unsupervised segmentation of spoken speech, based on an algorithm designed to segment discrete time series into meaningful episodes. We suggest that our model may capture part of the human process of speech segmentation. To substantiate our claim, we replicate an experiment that was performed on 8-month-old infants, and show that our algorithm performs similarly to the children.

2. Related Work

There are two main fields that are related to this topic. The first is the study of the speech segmentation methods that are used by infants. This constitutes a very broad area of research, with many subfields. This work is most related to the study of statistical learning in developmental psychology, which focuses on infants' ability to use statistical cues to segment language streams. These studies are the direct inspiration for this line of research, but they suggest no practical algorithm for replicating the results they have observed. The second related field of research pertains to algorithms for the segmentation of time series data. These studies suffer from the opposite problem. That is, there are many strategies by which to segment data, but not many that serve as a plausible model of infant segmentation.

2.1 Statistical Learning

The idea that infants use statistical cues to segment speech streams is very old (Harris, 1955). Specifically, the canonical theory is that they use the transitional probabilities between syllables as an indicator of word boundaries. Suppose that α and β are syllables in some language. Then the transitional probability $TP(\alpha \rightarrow \beta)$ is the probability that β follows

α when α appears in the speech stream. It stands to reason that syllables that appear together inside of a word would have a higher transitional probability than those that do not. Therefore, the argument goes, the transitional probabilities between syllables inside of words should be high, but the *TP* between syllables that cross a word boundary should be low. Hence, a child might easily segment a sequence of syllables by noting whenever the transitional probability dips down low.

This is precisely the strategy suggested in a series of experiments performed by Saffran *et. al.* (Saffran et al., 1996) (Saffran et al., 1997) (Saffran et al., 1999). The first of these experiments demonstrated that 8-month-old infants can, in fact, segment words based solely on statistical information. The children were played an artificially generated acoustic stream composed of the words *tupiro*, *golabu*, *bidaku* and *padoti* repeated in random order. After two minutes they were played a second stream consisting of a single word repeated over and over. Half of the time the word was from the original language, and the other half of the time it was a novel word, generated from the same syllables. The stimulus streams had no audible breaks between the words, no variation in pitch or meter, and no other cues as to the word breaks. The only clue was the transitional probability between the syllables. Inside of words it was always 100%, but between words it dropped to 25%. The stimulus stream was constructed specifically to be segmentable by the *TP* strategy. And the amazing result was that, after only two minutes, the infants were able to tell the novel words from the old.

The results of these experiments were taken as evidence that human infants really do pay attention to the transitional probabilities between syllables, and that they use them to segment audio speech. However, that's not really what these experiments showed. They showed that infants can segment audio speech using *some kind* of statistical model, and that it is powerful enough to work on the stimulus stream they were presented. Dips in inter-syllable transition probability were the simplest cue that they could have used to segment the sequence, but virtually any sophisticated model should have picked up this very simple pattern. And there is significant evidence to suggest that infants, in fact, are not using *TPs* to do this.

Most dramatically, multiple studies have showed that the direct application of the *TP* strategy performs poorly when used to segment phonetic transcripts of speech (Cairns and Shillcock, 1997) (Gambell and Yang, 2008). This exposes several of the weaknesses of the traditional statistical learning approach. First of all, a very high percentage of common words contain only one syllable. It is therefore

impossible for there to be a *TP* valley on both sides of the word. Moreover, the original conclusion that word-internal transitions should have higher probabilities than word-external ones is not always true in practice. Often, the last syllable of one word and the first syllable of the next happen to form a perfectly common pair. Similarly, many words contain syllable combinations that are, in general, rare (perhaps only appearing in a handful of words). The difference in single-syllable *TP* inside of and between words is more of a trend than a reliable rule.

2.2 Segmentation Algorithms

Most of the algorithms mentioned in this section are used to segment discrete token sequences (*i.e.*, they segment text - or text based phonemic transcripts of speech). This paper describes an algorithm that runs on real audio, and is able to perform the unsupervised segmentation of individual words from acoustic speech streams. So, in some sense, we are comparing apples and oranges. However, this previous work is certainly related, since it is also inspired by developmental psychology, and intends to accomplish roughly the same task.

There exist a wide variety of algorithms capable of segmenting discrete time series into meaningful "chunks." For instance, compression algorithms that find minimum description lengths can often be coerced into segmentation by using whatever encoding they perform (Nevill-Manning and Witten, 1997) (Cohen et al., 2007). Several studies have attempted to train Neural Nets to predict the subsequent phoneme given the last few, and induce breaks whenever the prediction is uncertain (Elman, 1990) (Cairns and Shillcock, 1997). Gambell and Yang suggested a method of segmenting speech by assuming that every word contains a single stressed syllable (Gambell and Yang, 2008). They reported very good results on the CHILDES dataset, transcribed to phonemes and then concatenated into syllables. Michael Brent published a thorough survey of many different strategies for attacking this problem (Brent, 1999b). In fact, his own algorithm has set the bar for the unsupervised segmentation of phonemic transcripts of infant directed speech (Brent, 1999a). It incrementally builds a lexicon and induces maximum likelihood parses in short phrases. Using this strategy, Brent was able to segment phonemic transcripts of child directed speech with precision and recall above 80%. This remains the best performing algorithm on this type of data.

However, Brent's algorithm pays no attention to statistical regularities in phoneme sequences, and typically builds very large lexicons with many wrong words. For instance, this algorithm would be incapable of segmenting the stimulus streams used in the statistical learning experiments, since they contained

no phrase boundaries. This demonstrates that, while some kind of bootstrapping, lexicon-based segmentation method might be useful, it does not completely model the human system. Perhaps infants use a similar process as part of their strategy, but they are also sensitive to statistical cues.

Recently the ACORNS project has been created to investigate human language acquisition (Boves et al., 2007). This research is unique, in that it attempts to learn the grounded meaning of words in an unsupervised way. However, the automatic segmentation of speech into words is a secondary goal to the extraction of semantic meaning. These two strategies are certainly complimentary, and children must perform both of these tasks in order to acquire language. In this paper, we do not address word learning, but instead focus entirely on unsupervised segmentation. Our goal is to introduce a unique unsupervised method for segmenting continuous data streams, to apply the method to speech, and suggest that such a model might characterize the statistical segmentation abilities of human infants.

2.3 Voting Experts

Voting Experts (*VE*) is an algorithm for the unsupervised segmentation of discrete time series into meaningful episodes (Cohen et al., 2007). It is a purely distributional algorithm, in that it relies solely on statistics calculated from the time series itself. *VE* has demonstrated an ability to accurately segment text, phonetic transcripts, vertical pixel columns scanned from text, discrete robot sensor data and even a text transcript of the acoustic streams used in this paper (Miller and Stoytchev, 2008a) (Cohen et al., 2007). It's based on the hypothesis that natural breaks in a sequence are usually accompanied by two information theoretic signatures. These are low *internal entropy* of chunks, and high *boundary entropy* between chunks.

In this context, the internal entropy of a chunk is simply its Shannon information, or the negative log of its probability (Shannon, 1951). So the higher the probability of a chunk, the lower its internal entropy. We can calculate the probability of a short sequence of tokens by observing how often that sequence appears in a longer time series. So, essentially, this marker picks out short sequences of tokens that appear often.

Boundary entropy is the uncertainty at the boundary of a chunk. Given a sequence of tokens, the boundary entropy is the expected information gain of being told the next token in the time series. This is calculated as

$$H_B(c) = - \sum_{h=1}^m P(h | c) \log(P(h | c))$$

where c is this given sequence of tokens, $P(h | c)$ is the conditional probability of symbol h following

c , and m is the number of tokens in the alphabet. Well formed chunks are groups of tokens that are found together in many different circumstances, so they are somewhat unrelated to the surrounding elements. If the boundary entropy of a subsequence is high it means that there is no particular token that is very likely to follow that subsequence. In other words, the next token is unpredictable.

In order to segment a discrete time series, *VE* preprocesses the series to build an n -gram trie, which represents all its subsequences of length less than or equal to n . It then passes a sliding window of length n over the series. At each window location, two "experts" use the trie to vote on how they would break the contents of the window. One expert votes to minimize the internal entropy of the induced chunks, and the other votes to maximize the entropy at the break. After all the votes have been cast, the sequence is broken at the "peaks" - locations that received more votes than their neighbors, so long as the total votes at the location exceeded a threshold V_t . For all of our experiments we chose $n = 7$, and we varied V_t over a range of values. The effect of this variation will be discussed later, and evident in the results of our experiments. The choice of n roughly approximates the expected length of an individual "chunk." This algorithm runs in linear time with respect to the length of the sequence, and can therefore be used to segment very long sequences. For further technical details of *VE*, or a more in-depth discussion of the roles of V_t and n , see (Cohen et al., 2007).

This model bears a strong resemblance to the statistical learning approach mentioned before. If the conditional probability between each syllable within a word is high, then by definition the internal entropy of the word is low. But instead of evaluating each transitional probability in isolation, *VE* looks for short sequences of tokens where all of the *T**P*s are high. Similarly, the boundary entropy of a sequence is high precisely when there is no particular token that is very likely to come next. However, instead of focusing on the transition probability between two syllables that happened to be adjacent, *VE* looks at whether the *TP* is *expected* to be low. This is an important difference, and it solves one of the major problems with the transitional probability approach. When the last syllable of one word and the first syllable of the next happen to form a likely pair, the *TP* based approach fails. But *VE* isn't affected when the *TP* at the word boundary is high, as long as the next token is unpredictable based on several previous tokens. This extra power is afforded by the use of the more sophisticated entropy metrics. Moreover, the model should still be extremely sensitive to the transitional probability cues, since the entropy cues must be present wherever the *TP* cues are.

In this paper we extend *VE* to work on audio data. We then use this algorithm to reproduce Saffran *et al.*'s original experiments. *VE* might not be the best possible unsupervised distributional segmentation algorithm, but it is certainly a powerful one. Additionally, the complexity of its metrics seems close to the horizon of biological plausibility. It is not unrealistic to think that humans naturally pick out commonly recurring sequences of sounds, and tend to place breaks at moments of unpredictability. Accordingly, we suggest that *VE* is a strong candidate for a usable model of the human distributional segmentation mechanism.

3. Datasets

We obtained two stimulus streams from the original infant speech segmentation experiments (Saffran *et al.*, 1996). Each audio stream is about 60 seconds long and contains roughly 90 "words." The first stream (stream A) was composed, as described above, of randomly ordered instances of the four words *tupiro*, *golabu*, *bidaku* and *padoti*. The second stream (stream B) was composed of random instances of the words *tilado*, *dapiku*, *pagotu* and *burobi*. The second language is composed of the same syllables as the first, but arranged so that the concatenation of words in either language cannot produce a word from the other. So in some sense these two audio streams are disjoint.

In the original experiment, the infants were played a stream created in the same way as stream A, and then tested on a single word repeated over and over. This method is useful when evaluating infants because it is simple. However, we can perform a more thorough evaluation of our model since it produces explicit break locations. We found it more informative to test our model by training it on one stimulus stream and then testing it on the other. This provides more information on the performance of the model, but the results can clearly be compared to those of the infant experiments.

In order to evaluate the segmentations induced by our algorithm, we manually recorded the timestamps of all of the word boundaries in the two stimulus streams. It is impossible for this process to be absolutely precise, since spoken audio is not actually composed of distinct phonemes, and word breaks are not always marked by silence. The sound morphs from one allophone to the next, providing few clear boundaries. However, the speech in the streams used by Saffran *et al.* is very regular, which allowed us to consistently place breaks at the same location in each word. The waveform in between each word pair was identical every time it appeared, since it was generated artificially. The beginning and ending of each word was verified acoustically once, and then the boundaries could be placed in exactly the same

location for each instance of each word. The resulting "answer keys" were therefore consistent, and as close to the ground truth as possible.

4. Audio Segmentation Algorithm

The raw audio of both stimulus streams was converted into a sequence of Mel-cepstral feature vectors, along with their first and second order time derivatives and their log energy (Davis and Mermelstein, 1980). The standard 13 cepstral features were used, so that each time slice of the audio was represented by a 42-dimensional real valued feature vector. That's 13 cepstral features, 13 first order and 13 second order time derivatives, and the log energy of each. This is a standard method of feature extraction for speech processing, and it was performed using the Matlab package "Voicebox."

Since *VE* is designed to work on a sequence of tokens, these feature vectors must be quantized into a manageable alphabet. A common technique in automatic speech recognition is to use Hidden Markov Models with continuous observation densities to recognize phonemes (Rabiner, 1990). We will draw inspiration from these models, however we cannot apply the techniques exactly. In the infant experiments the children learned to segment novel language streams in a completely unsupervised way. Therefore, any model of this process must also be entirely unsupervised. These HMMs are typically trained on labeled data, disqualifying them as plausible models. Specifically, a separate Markov chain is typically trained to represent each phoneme in the language. The models are built using a large set of hand-labeled instances of each phoneme, and then their parameters are improved by bootstrapping over a large audio corpus. Instead, we will suggest an unsupervised model that can convert an audio stream into a state sequence suitable for segmentation, but one that does not necessarily correspond to the phoneme sequence as a human would label it.

4.1 Unsupervised Acoustic Model

The critical observation is that we don't necessarily need a sequence that corresponds to the true phonemes of the language. All that's needed is a model that decomposes an audio stream into a sequence of its most salient acoustic features. These may or may not correspond to the "phonemes" as a human might label them. But that is irrelevant, at least as far as *VE* is concerned.

Just such a model was suggested by (Iwahashi, 2006), and implemented by (Brandl *et al.*, 2008). We used a version of that model in this work. Each phoneme was represented using a 3-node Markov chain with Bakis-topology, with the observation probability density of each state represented by a mixture of Gaussian functions

(Rabiner, 1990). In order to train these models without labeled data, we first trained a completely connected Markov network containing 10 Gaussian mixture states on the acoustic stream. The parameters of the network were initialized using k-means, and then optimized using EM, so no labeled data was required. Then, we stochastically sampled paths of length 3 through that network based on the learned transition probabilities. The m most common paths were used to initialize m 3-node Markov chains. The last state of each chain was connected to the first state of every other chain, including itself, initialized with uniform transition probability. The parameters of this larger Markov model were then optimized over the corpus using EM.

In one implementation, m was set using the Akaike information criterion (Brandl et al., 2008). Instead we used $m = 10$ to build the models used in this paper. We varied this parameter, and found that it did not have a strong effect on the performance of the model on this task. The results of that evaluation are not included for space considerations. However, if this model were to be applied to a larger or more complex dataset, such an evaluation would certainly be necessary.

4.2 Segmentation

Given a model as described above and an acoustic stream for segmentation, we converted the stream into a state sequence using Viterbi decoding. The state sequence was simplified by assuming that all nodes from the same Markov chain were equivalent. So instead of a sequence of nodes in the HMM, the stream was represented as a sequence of 3-node Markov chain labels. However, this created sequences with long stretches of the same label repeated over and over. These repeated labels were collapsed into a single token. So the final token sequence represented the order in which these chains were visited in the decoding of the stimulus stream, with no information about how long the sound stayed in the same chain. If the chains corresponded to the phonemes of the language, as they do in more typical acoustic models, the result would be a transcription of the spoken phonemes of the stream. The idea is that the unsupervised model approximates the phoneme sequence, but perhaps extracts a slightly different set of fundamental sounds.

We ran *VE* on the resulting label sequence. *VE* placed breaks at locations of low internal entropy and high boundary entropy. Then, after accounting for the collapsed (*i.e.*, repeated) states, it produced the time stamps of all of the induced break locations in each audio stream. These breaks were then checked against the answer keys that had been manually created for each stimulus stream (See Figure 1).

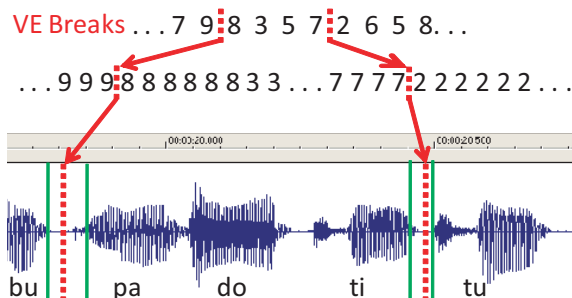


Figure 1: Evaluation of the breaks induced by *VE*. Each break is mapped to its location in the expanded state sequence, which corresponds to a timestamp in the audio stream. The break counts as correct if it falls within the marked boundary between two words. The states are represented by their numeric index in the Markov model.

5. Evaluation Methodology

In order for an induced break to count as a correct break, it had to be placed between the specified end of the previous word and the beginning of the next one, within an error of one time slice. The feature vectors that composed the audio stream were calculated using a window that was 0.016 seconds wide with a 50% overlap. This means that the additional time slice allowed at each boundary increased the break window by 0.008 seconds. This leeway was provided to compensate for labeling errors or other boundary conditions.

An induced break was counted as breaking two words if it was placed anywhere in the window between them. Both stimulus streams were 61.2 seconds long. Stimulus stream A contained approximately 7.7 seconds of “break” time, and stream B contained 7.2 seconds. The reason for the discrepancy is that the different pronunciations of the first and last syllables of the words in each stream led to slightly different amounts of time between them. It should be noted that these “breaks” are not perceivable when listening to the stream, and are no longer than the space between the phonemes within words.

Unfortunately these boundaries make it easier for the algorithm to accidentally induce a break between two words. Thus, even random breaks will be counted as correct some of the time. Accordingly, we used a Monte Carlo method to simulate random segmentations for each experiment. Each reported result is accompanied by the results of inducing a large number of random segmentations, each one having the same number of induced breaks as the algorithm produced. The random breaks were induced in the same compressed state sequence used by *VE*, and were evaluated in the same manner. These random trials are averaged and provide a baseline from which to evaluate the algorithm.

The quality of the segmentation is evaluated based

on the accuracy, hit-rate and f-measure of the induced breaks. In this case, accuracy is the percentage of induced breaks that are correct, hit-rate is the percentage of true breaks found by the algorithm, and the f-measure is the harmonic mean of the two, given by

$$\text{f-measure} = \frac{2 * \text{accuracy} * \text{hitrate}}{\text{accuracy} + \text{hitrate}}$$

The f-measure is treated as most important, since it strikes a balance between the other two. It's possible to increase the accuracy of the segmentation by inducing fewer breaks, but being more confident about those that are induced. However, this will lower the hit-rate. Similarly we can raise the hit-rate by inducing more breaks, but this will lower the accuracy. The Voting Experts algorithm lets us explicitly make this tradeoff by adjusting the threshold V_t for the minimum number of votes required to induce a break at a location. All three of these metrics will be reported for each of our experiments. Additionally, the experiments will be repeated for a range of thresholds V_t , and the sensitivity of these metrics to variation in that threshold will be demonstrated.

6. Experimental Results

We have outlined a general, unsupervised algorithm for the segmentation of an audio stream. First, convert the stream into an appropriate sequence of feature vectors - in our case the Mel-cepstrum. Then train an unsupervised Gaussian Mixture HMM (GMHMM) on the sequence as described above. Use this model to produce a sequence of Markov chain labels based on the audio stream. Finally, collapse the repeated labels in this sequence and run *VE* on the result.

This algorithm constitutes a very basic application of the *VE* model to a real audio stream. The first question is whether this can induce an accurate segmentation. The second question is whether we can use this system to model the human segmentation mechanism. The following experiments were designed to answer both of these questions.

Experiment 1: We ran the segmentation process described above separately on each stimulus stream (A and B). We then compared the induced breaks to the true breaks for each stimulus stream. The results are shown in Figure 2.

The segmentation induced on both audio streams was significantly more accurate than chance. Clearly this model is capable of segmenting the given stimulus streams. These results are even more surprising when considering that these models were each trained on only one minute of audio. Presumably infants might be better equipped to perform this task since they have the advantage of a previously trained acoustic model. They do not have to learn it from scratch in just one minute as we have done here. But even with that limitation, *VE* performs very well.

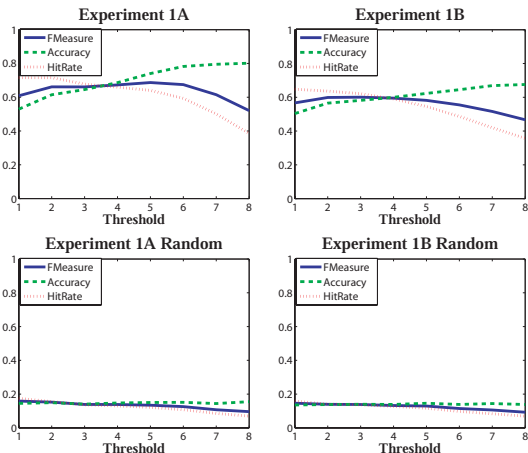


Figure 2: The F-measure, accuracy and hit-rate of the segmentation of both stimulus streams in Experiment 1, along with the performance of random segmentations on both datasets.

It should be noted that the initialization of the acoustic models is a stochastic process, and leads to a unique model every time. The EM algorithm does not necessarily find a global optimum for the model parameters, but only a local maximum. Therefore, the model should not be evaluated based on a single instantiation, but rather based on several trials. Accordingly, we trained 10 different acoustic models on each of the two stimulus streams. All three experiments were performed 10 different times with 10 different pairs of models. The results were averaged to produce the results reported.

Additionally, the segmentation step, where *VE* was run on the token sequence, was repeated for different threshold values ranging from 1 to 8 for each experiment. Notice the tradeoff between accuracy and hit-rate as V_t varies. The f-measure, accuracy and hit-rate are reported both for the aggregate over all 10 models, as well as for the random trials over the same data. For each trial that was done with a single model, 10 random trials were performed. So, overall, 100 random trials were performed in each experiment for each stimulus stream.

Experiment 2: The point of this experiment is to demonstrate that an acoustic model trained on stimulus stream A can still be used to segment the audio from stream B, and vice versa. The two streams are composed of the same set of syllables. The only difference is the order in which the syllables are spoken, which may produce some interaction effects that the GMHMM cannot model. However, most of the sounds are the same. So, for instance, the tokenization of stream B by an acoustic model trained on stream A should still be useful for inducing a segmentation on B.

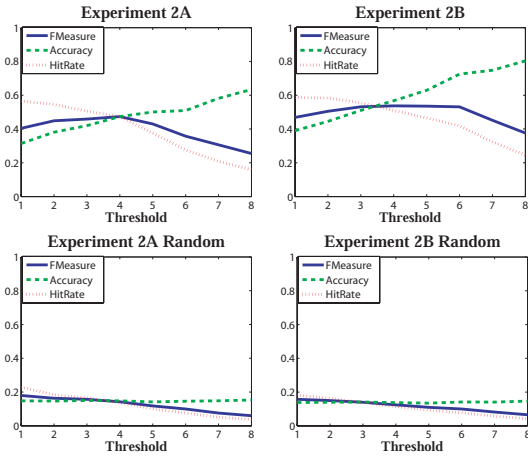


Figure 3: The F-measure, accuracy and hit-rate of the segmentation of both stimulus streams in Experiment 2. Once again, the performance of random segmentation is also shown.

To demonstrate this, we trained an acoustic model on each stream to obtain $GMHMM_A$ and $GMHMM_B$. Then we used $GMHMM_A$ to tokenize the feature vectors from stimulus stream B and $GMHMM_B$ to tokenize stream A. Then we trained a VE model on each of the token sequences and induced a segmentation. Once again we used the true breaks to evaluate the results (see Figure 3).

There is a slight drop in both the accuracy and hit rate of each segmentation in this experiment. However, in each case the algorithm still performed much better than chance. There is not a tremendous loss due to the unmodeled interaction of the diphones in the stimulus streams. This fact is important in understanding the results of experiment 3.

Experiment 3: This experiment is intended to replicate the results of the infant studies. In those experiments, the children listened to one stimulus stream, and were then presented a novel token from the second stream. Similarly, in this experiment, our model is trained on one stimulus stream, and then used to segment the other. That is, the $GMHMM$ and the statistical model of VE (the experts) are trained on stream A, and then that model is used to segment stream B and vice versa.

Figure 4 shows that the algorithm is almost completely unable to induce a segmentation. It performs only slightly better than chance, and this is most likely due to its ability to pick out syllables. From the results of experiment 2 we can conclude that the poor performance is not the fault of the acoustic model. Instead, the language model trained on one language is insufficient to induce a segmentation in another.

As the threshold increases, the algorithm induces very few breaks. When V_t is higher than 5, almost no breaks are induced (*e.g.*, no breaks were induced at all when $V_t = 8$). This explains why the accuracy

becomes erratic at higher threshold levels, and the hit-rate drops very low. The random segmentations only contained as many breaks as the algorithm induced, so the random hit-rate drops as well. The fact that not very many breaks were induced indicates that the experts did not vote for the same break locations very often. They could not agree on suitable breaking points, and therefore did not create many breaks. Essentially, the algorithm was confused.

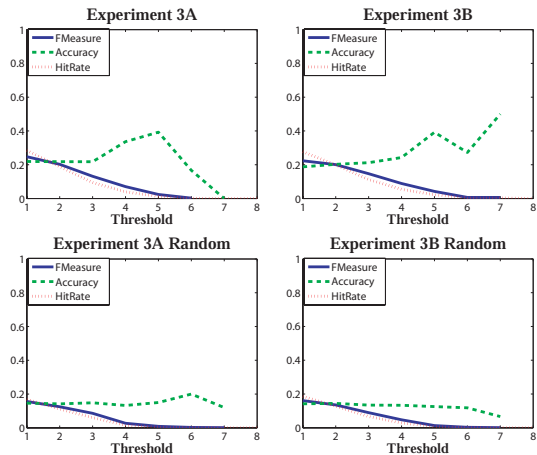


Figure 4: The F-measure, accuracy and hit-rate of the segmentation of both stimulus streams in Experiment 3, along with the results of the random segmentation.

This corresponds precisely with the situation of the 8-month-old who listens to stimulus stream A, and then hears a novel word from stream B. The child has learned the sounds present in the stream, and has learned a statistical model that characterizes it. Then, suddenly, that model is violated. The child is initially unable to use the old model to “understand” the novel word, and therefore becomes confused.

7. Conclusions and Future Work

We have described an unsupervised technique for transforming spoken audio into a discrete sequence of tokens suitable for segmentation by the Voting Experts algorithm. This algorithm is novel in its application to real audio, and its reliance on simple but powerful information theoretic cues. We have shown that the VE model is capable of inducing an accurate segmentation on an audio stimulus stream with very limited training data. Finally, we have shown that the behavior of this model mimics the behavior of 8-month-old infants. This should be counted as a small victory for VE as a model of human segmentation. It also demonstrates that distributional cues can be used to segment audio streams. Specifically, the low internal entropy and high boundary entropy of chunks provide sufficient markers to do so.

The psychological studies that have explored infant statistical learning have used stimulus streams

that could be segmented using transitional probabilities. Infants can segment these simple streams, but the full extent of their capabilities remains unknown. *VE* can segment the same stimulus streams, and therefore is not disqualified as a possible model of the human distributional speech segmentation mechanism. If an algorithm can pass that test, it's at least a plausible candidate. However, this may be an easier task than children face with natural language.

It is simply unknown how important a role distributional segmentation really plays in the acquisition of language, and how sophisticated that mechanism is. Presumably it is significantly useful, or else children wouldn't demonstrate this ability at such a young age. Since some studies have shown that the simple statistical learning approaches are not sufficient to segment natural language, we should conclude that infants have a more sophisticated strategy. *VE* has the advantage of being able to segment many different kinds of speech, including natural language phoneme sequences (Miller and Stoytchev, 2008a). This makes it a much more attractive candidate for modeling human segmentation, since the approaches based on transitional probabilities have not done the same. The next logical step is to use this model on a natural language corpus to see how effective it can really be.

Acknowledgments

An earlier version of this paper, with a much simpler acoustic model and less detailed analysis, was accepted into the NIPS 2008 Workshop on Speech and Language (Miller and Stoytchev, 2008b). We would like to thank the organizers and participants for the suggestions and feedback that helped improve our work. We would also like to thank Richard Aslin from the University of Rochester for providing us with the stimulus streams used in the original Saffran *et al.* experiments. Finally, we would like to thank Paul Cohen from the University of Arizona for generously providing the source code for the original Voting Experts algorithm.

References

- Boves, L., ten Bosch, L., and Moore, R. (2007). ACORNS – towards computational modeling of communication and recognition skills. In *Proceedings of ICCI*.
- Brandl, H., Wrede, B., Joublina, F., and Goerick, C. (2008). A self-referential childlike model to acquire phones, syllables and words from acoustic speech. In *Proceedings of the 7th IEEE International Conference on Development and Learning*, pages 31–36.
- Brent, M. R. (1999a). An efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learning*, 34(1-3):71–105.
- Brent, M. R. (1999b). Speech segmentation and word discovery: A computational perspective. *Trends in Cognitive Science*, 8(3):294–301.
- Cairns, P. and Shillcock, R. (1997). Bootstrapping word boundaries: A bottom-up corpus-based approach to speech segmentation. *Cognitive Psychology*, 33:111–153.
- Cohen, P., Adams, N., and Heeringa, B. (2007). Voting Experts: An unsupervised algorithm for segmenting sequences. *Journal of Intelligent Data Analysis*, 11(6):607–625.
- Davis, S. and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 24(4):357–366.
- Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14:179–211.
- Gambell, T. and Yang, C. (2008). Mechanisms and constraints in word segmentation. Manuscript, Yale University.
- Harris, Z. S. (1955). From phoneme to morpheme. *Language*, 31.
- Iwahashi, N. (2006). *Symbol Grounding and Beyond*, volume 4211/2006 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg.
- Klatt, D. H. (1979). Speech perception: A model of acoustic-phonetic analysis and lexical access. *Journal of Phonetics*, 7:279–285.
- Miller, M. and Stoytchev, A. (2008a). Hierarchical Voting Experts: An unsupervised algorithm for hierarchical sequence segmentation. In *Proceedings of the 7th IEEE International Conference on Development and Learning (ICDL)*.
- Miller, M. and Stoytchev, A. (2008b). Unsupervised audio speech segmentation using the Voting Experts algorithm. In *NIPS Workshop on Speech and Language: Learning-based Methods and Systems*.
- Nevill-Manning, C. and Witten, I. (1997). Identifying hierarchical structure in sequences: A linear-time algorithm. *Journal of Artificial Intelligence Research*, pages 7:67–82.
- Rabiner, L. R. (1990). A tutorial on hidden markov models and selected applications in speech recognition. *Readings in speech recognition*, pages 267–296.
- Saffran, J. R., Aslin, R. N., and Newport, E. L. (1996). Statistical learning by 8-month-old infants. *Science*, 274(5294):1926–1928.
- Saffran, J. R., Johnson, E. K., Aslin, R. N., and Newport, E. L. (1999). Statistical learning of tone sequences by human infants and adults. *Cognition*, 70:27–52.
- Saffran, J. R., Newport, E. L., Aslin, R. N., and Tunick, R. A. (1997). Incidental language learning: Listening (and learning) out of the corner of your ear. *Psychological Science*, 8(2):101–105.
- Shannon, C. (1951). Prediction and the entropy of printed english. Technical report, Bell System Technical Journal.

A Comparison of Strategies for Developmental Action Acquisition in QLAP

Jonathan Mugan

Department of Computer Science
The University of Texas at Austin
Austin, TX 78712 USA
jmugan@cs.utexas.edu

Benjamin Kuipers

Computer Science and Engineering
University of Michigan
Ann Arbor, MI 48109 USA
kuipers@umich.edu

Abstract

An important part of development is acquiring actions to interact with the environment. We have developed a computational model of autonomous action acquisition, called QLAP. In this paper we investigate different strategies for developmental action acquisition within this model. In particular, we introduce a way to actively learn actions and we compare this active action acquisition with passive learning of actions. We also compare curiosity based exploration with random exploration. And finally, we examine the effects of resource restrictions on the agent's ability to learn actions.

1. Introduction

We seek to understand how an agent (human or otherwise) can learn to adapt to its environment through the process of development. Gibson (1988) proposed that human children are endowed with systems to allow them to explore and learn about the world. She emphasized that it was this exploration that enabled cognitive development. One such system appears to be that for learning contingencies. It has been proposed that humans have an innate contingency detection module (Gergely and Watson, 1999). Human infants can detect contingencies in their environment shortly after birth (DeCasper and Carstens, 1981), and they can link these contingencies with observable effects (Adolph and Joh, 2007).

Inspired by this idea that learning can take place through the acquisition of contingencies, we created the Qualitative Learner of Action and Perception (QLAP). QLAP is constructivist in the tradition of Piaget (1952) because the agent constructs representations of the environment. QLAP learns contingencies and actions through autonomous exploration. QLAP learns contingencies by observing events in the environment and looking for correlations (Mugan and Kuipers, 2008,

Mugan and Kuipers, 2007). Once a contingency is found that is sufficiently deterministic, QLAP creates a plan to perform an action based on that contingency (Mugan and Kuipers, 2009).

Adolf and Joh (2007) note the importance of action learning in the role of providing agent-centered input to the perceptual systems. Generating agent-centered experience by learning actions requires that the agent autonomously explore its environment. This type of exploration has been characterized as intrinsically motivated learning (Berlyne, 1965) and is essential for autonomous development (Ryan and Deci, 2000). The problem of picking which action to choose has been studied extensively, for example see (Schmidhuber, 1991, Huang and Weng, 2002, Marshall et al., 2004). One promising approach is picking actions that maximize the learning gradient (Oudeyer et al., 2007). However, exploration for learning actions is more than picking which action to choose. The agent must first form the actions.

QLAP assumes that the agent has motor primitives but no initial complex actions. From these motor primitives, QLAP learns actions such as reaching out to hit a block. However, some more complex actions may have to be learned using *active action acquisition*. Active action acquisition involves two steps. First, the agent tunes its search for contingencies related to a desired action to be more sensitive, so that it finds contingencies that it might otherwise overlook. And second, the agent makes it more likely that a found contingency will become a plan to perform the action by lowering the required reliability of the contingency.

The contribution of this paper is to provide an evaluation of exploration strategies for learning actions. We evaluate different exploration strategies in an environment inspired by the sticky mittens experiments (Needham et al., 2002). In these experiments, children wore mittens covered with Velcro that allowed them to more easily grasp objects. They found that infants trained with the sticky mittens exhibited

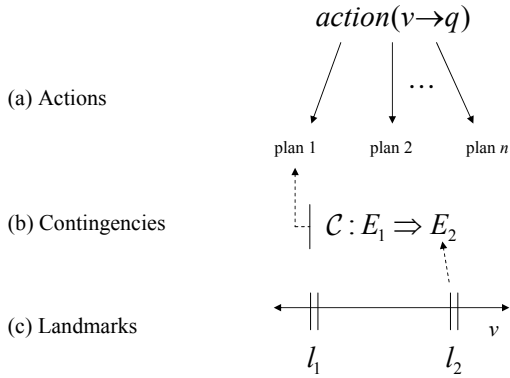


Figure 1: (a) An action brings a qualitative variable v to a desired value q . Each action can have one or more plans. Each plan is a different way to perform the action. (b) Each plan is learned by first learning a contingency. A contingency links an antecedent event E_1 with a consequent event E_2 . Associated with each contingency is a probability table that gives the probability of event E_2 following event E_1 for each value of the variables in context \mathcal{C} . (c) Each event is able to be perceived because of the discretization created by the landmarks.

more object engagement and more sophisticated object exploration strategies.

We evaluated the effect of using active action acquisition. We found that active acquisition improved the agent’s performance on the task of picking up the block with the sticky mitten, but hurt the agent’s performance on the easier task of moving the block. We found that using active action acquisition in combination with the exploration method of Intelligent Adaptive Curiosity (IAC) (Oudeyer et al., 2007) worked best in this continuous domain for enabling the agent to develop so that it could learn to pick up the block using the sticky mitten. We also evaluated the use of developmental restrictions and we found that certain developmental restrictions allowed the agent to reduce the number of learned contingencies without hindering learning. And finally, we found that the developmental trajectory allows that agent to progress from actions being used mostly as exploration to actions being used as subactions for other actions.

2. The Qualitative Learner of Action and Perception, QLAP

The Qualitative Learner of Action and Perception (QLAP) is a computational model for learning both important perceptual distinctions and actions (see Figure 1). QLAP assumes that the agent can distinguish objects from the background and track them. QLAP also assumes that the agent can measure distances between objects and that the agent has motor variables for output. The result of these assumptions

is that the agent interacts with the world using a set of real-valued variables.

2.1 Qualitative Representation

The distinctions that QLAP learns allows it to represent the state of the world qualitatively. It does this by converting the continuous input and motor variables to qualitative variables (Kuipers, 1994). A qualitative representation allows the agent to focus on important distinctions while ignoring others. The qualitative variables are created by discretizing the continuous variables using *landmarks*. A landmark is a symbolic name for a point on a number line. A variable v with two landmarks l_1 and l_2 would have a set of five possible qualitative (discrete) values $\{(-\infty, l_1), l_1, (l_1, l_2), l_2, (l_2, +\infty)\}$. QLAP must learn these landmarks. For example, QLAP learns a landmark that a force of at least 300 is needed to move the hand to the right. It also learns a landmark that a distance of 0 between the right side of the hand and the left side of the block is important to move the block to the right.

2.2 Landmarks to Events

Landmarks allow the agent to perceive *events*. An event is the change in qualitative value of a variable. We use the notation $E = X_t \rightarrow x$ to denote event E where the value of qualitative variable X changes to x at time t (although the t may be omitted for brevity.) For example, when the distance between the right side of the hand and the left side of the block goes to 0.

2.3 Events to Contingencies

The perception of events allows the agent to learn *contingencies*. Contingencies link an *antecedent event* $E_1 = X \rightarrow x$ with a *consequent event* $E_2 = Y \rightarrow y$ together in time. For each contingency, QLAP learns a context \mathcal{C} that gives the probability of the consequent event following the antecedent event for each value of the variables in \mathcal{C} . We call the highest probability of event E_1 leading to event E_2 the *best reliability* of the contingency. Once the best reliability of a contingency exceeds 0.75 the contingency is labeled *sufficiently deterministic*.

New landmarks can be learned by finding new distinctions that make contingencies more reliable. For example, the agent may learn a contingency that states that the event of a positive force on the hand will cause the event of the hand moving to the right. Once this contingency is learned, QLAP can examine the real values of the variables and determine if there is a new distinction that will make this contingency more reliable. In this case, it takes a force of 300 units to move the hand to the right. The agent

can then update the contingency to reflect this new distinction by introducing a landmark. The agent can also learn that the hand will not move to the right if it is already all the way to the right. It can then learn a landmark on the location of the hand to indicate when it is in its rightmost position.

2.4 Contingencies to Plans for Actions

In QLAP, the agent learns *actions* to achieve the qualitative values of variables. Each action sets the qualitative value of a variable to a desired value. In QLAP, actions may be performed in more than one way. Each way to perform the action is called a plan. Each plan is represented as an option (Sutton et al., 1999). Once a contingency is sufficiently deterministic it is converted into a plan. These plans are learned using reinforcement learning (Sutton and Barto, 1998), see (Mugan and Kuipers, 2009) for details.

3. Developmental Learning in QLAP

QLAP is not given a learning objective but learns in a developmental progression. This developmental progression comes from incrementally learning contingencies, actions, and landmarks. In addition, it comes from developmental restrictions that take three forms:

1. restrictions on learning contingencies

In QLAP, a contingency can only be learned if its antecedent event can be reliably predicted by a previously learned contingency.

2. restrictions on learning plans

A contingency can only be converted to a plan if the antecedent event can be reliably achieved using an existing action.

3. restrictions on cognitive load

An agent has limited cognitive resources and an important part of development is freeing up resources. QLAP designates an action as *open*, *full*, or *closed*. An action is closed if it can be achieved 75% of the time; otherwise, it is full if it has 5 plans; and it is open otherwise. Actions that are closed or full do not accept additional plans. When an action is closed, it also affects the learning of contingencies. QLAP does not add a contingency if the action to bring about the consequent event is closed.

Contingencies can also be deleted. If the contingency does not become a plan after 100,000 timesteps, it is deleted. When an action is closed, all of the related contingencies that are not part of plans for that action are deleted.

Plans can also be deleted. A plan and its associated contingency are deleted if its associated

action is still not closed and the reliability of the plan is less than 5%.

3.1 Choices Made During Exploration

The agent continually makes three types of choices during its exploration. These choices vary in time scale from coarse to fine.

1. The agent chooses an *exploration action*, which is a previously learned action that it can practice. This can be done randomly or by using a version of Intelligent Adaptive Curiosity (IAC) (Oudeyer et al., 2007) which first measures the change in the agent’s ability to perform the action over time and then chooses actions where that ability is increasing. For IAC, we use a time window $\tau = 25$ and a smoothing parameter $\theta = 25$ (before the time window of $\tau = 25$ is full, actions are chosen based on the product of probability of success in the current state and the entropy of their overall reliability).
2. The agent chooses the best plan for performing the action. The agent chooses the plan most likely to succeed in the current state with probability 0.95 and chooses a random plan otherwise.
3. The agent chooses the subaction within the plan. This is done using the standard reinforcement learning technique ϵ -greedy that balances exploration with exploitation (Sutton and Barto, 1998).

3.2 Execution

An outline of the execution of QLAP is shown in Algorithm 1. Note that for the first 20,000 timesteps the agent chooses random motor babbling exploration actions. After that point it chooses a motor babbling action with probability 0.1, otherwise it chooses an exploration action and action plan according to (Mugan and Kuipers, 2009).

4. Active Action Acquisition

A plan to perform an action is formed when the contingency is sufficiently deterministic. In the developmental progression just described, the agent learns these plans without paying special attention to what the goal of the associated action is. We call this approach *passive action acquisition*. This method of passive learning may not be sufficient to learn difficult actions. To learn difficult actions, the agent may have to employ active action acquisition. To learn a plan for an action chosen for active action acquisition, QLAP

1. **lowers the threshold needed to learn a contingency.** QLAP learns a contingency linking an event E_1 and an event E_2 , if E_2 is more likely to

Algorithm 1 The Qualitative Learning of Action and Perception (QLAP)

```
1: for  $t = 1 : \infty$  do
2:   Sense environment
3:   Convert input to qualitative values using cur-
   rent landmarks
4:   Update statistics to learn new contingencies
5:   Update statistics for each contingency
6:   if  $\text{mod}(t, 2000) == 0$  then
7:     Learn new contingencies
8:     Delete unneeded contingencies and plans
9:     Learn new landmarks to change qualitative
   representation
10:    Learn new actions
11:  end if
12:  if current exploration action is completed
   then
13:    Choose new exploration action and action
   plan
14:  end if
15:  Get low-level motor command based on plan
   of current exploration action
16:  Pass motor command to robot
17: end for
```

soon occur given that E_1 has occurred than otherwise. More formally, if we define a time window with the predicate $\text{soon}(t, E)$ that is true if event E occurs within a window of $k = 5$ timesteps starting at time t , then we can say that the contingency is formed if

$$Pr(\text{soon}(t, E_2) | E_1(t)) - Pr(\text{soon}(t, E_2)) > \theta_p$$

where $\theta_p = 0.05$. If event E_2 is chosen to be the goal of an actively acquired action, we make it more likely that a contingency will be learned by using $\theta_a = 0.02$ instead of $\theta_p = 0.05$.

- lowers the threshold needed to learn a plan.** A contingency becomes a plan if its best reliability is greater than 0.75. For a contingency with a consequent event that is chosen to be the goal of an actively acquired action, this threshold is reduced to 0.25.

This leaves the question of when to specify events as goals of actively acquired actions. An event is chosen as a goal for active action acquisition if the probability of being in a state where the event is satisfied is less than 0.05; we call such an event *sufficiently rare*. This is reminiscent of Bonarini et al. (2006). They consider desirable states to be those that are rarely reached or are easily left once reached.

5. Evaluation

We run experiments using the environment shown in Figure 2. The environment is implemented in

Breve (Klein, 2003) and has realistic physics. The simulation consists of a robot at a table with a block. The robot has an orthogonal arm that can move in the x , y , and z directions. During learning, the agent chooses exploration actions autonomously. Each time the agent knocks the block out of reach, the block is replaced with a different block and put on the table. The block size varies randomly in length from 1.0 to 3.0 units.

For each experiment we trained 40 agents. We trained each for 250,000 timesteps, which corresponds to about 3.5 hours of physical experience. The robot has a “sticky mitten.” If the center of the block touches the bottom of the hand, then the block is “grabbed.” For simplicity, there is no ungrab action. Instead, the block has a probability of 0.1 of becoming ungrabbed at each timestep. Then when the block becomes ungrabbed, it falls to the table with probability 0.5 or gets moved to another place on the table with probability 0.5. To make the environment more realistic, there are two distractor objects that float in front of the agent. The agent can perceive the distractor objects and learn contingencies about them, but cannot interact with them.

5.1 Evaluation Tasks

We measure the performance on two tasks. The first task is that of moving the block in a specified direction. The agent is told to move the block either left, right, or forward. The second task is picking up the block using the sticky mitten.

QLAP autonomously learns without being specified a task. We can be confident that it will learn the specified tasks because the number of variables in the environment is small. However, during learning, the agent does not know that it will be evaluated on these tasks.

Every 10,000 timesteps (about every 8 minutes of physical experience) we save the state of the agent. We then test how well each can do that task starting from this stored learned state. Each evaluation consisted of 100 episodes. Each episode lasted for 300 timesteps or until the block was moved. The agent received a penalty of -0.01 for each timestep, and it received a reward of 10.0 if it completed the task.

5.2 Experimental Conditions

active random This case used active action acquisition with exploration actions chosen randomly from a uniform distribution.

active IAC This case used active action acquisition with exploration actions chosen using Intelligent Adaptive Curiosity.

passive random This case used passive action acquisition with exploration actions chosen randomly.

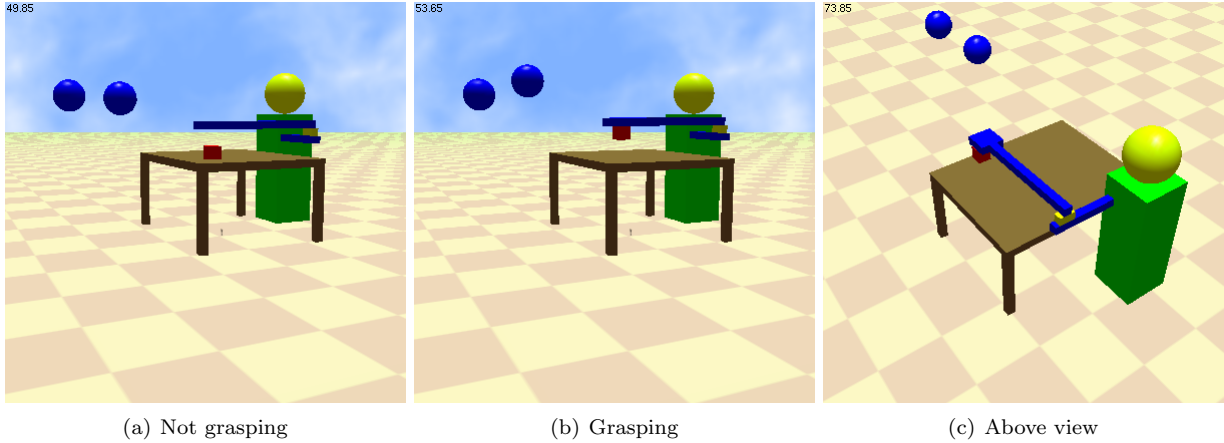


Figure 2: The robot is implemented in Breve; a simulator with realistic physics. The robot has a torso with a 3-dof orthogonal arm and is sitting in front of a table with a block and two floating distractor objects. The robot has three motor variables \tilde{u}_x , \tilde{u}_y and \tilde{u}_z that move the hand in the x , y , and z directions, respectively. The location of the hand is given by three time-varying continuous proprioceptive variables \tilde{h}_x , \tilde{h}_y , \tilde{h}_z that represent the location of the hand in the x , y , and z directions, respectively. The relationship between the hand and the block is represented by the continuous variables \tilde{x}_{rl} , \tilde{x}_{lr} , \tilde{y}_{tb} , \tilde{y}_{bt} , and \tilde{z}_{du} . The variable \tilde{x}_{rl} is the x value of the location of the right side of the hand in a coordinate system whose origin is centered on the left side of the block (variable \tilde{x}_{lr} is analogous). The variable \tilde{y}_{tb} is the y value of the location of the far (top) side of the hand in a coordinate system whose origin is centered on the bottom (near) side of the block (variable \tilde{y}_{bt} is analogous). And variable \tilde{z}_{du} is the z value of the location of the down side of the hand in a coordinate system whose origin is centered on the up side of the block. Additionally, the variables \tilde{c}_x and \tilde{c}_y represent the two-dimensional coordinates of the center of the hand in the frame of reference of the center of the block. There is also a Boolean touch variable T that is true if the block is colliding with the hand and the center of the top of the block is underneath the bottom of the hand. There are also two distractor floating objects f^1 and f^2 . The variables for f^1 are \tilde{f}_x^1 , \tilde{f}_y^1 , and \tilde{f}_z^1 and the variables for f^2 are analogous. Including the direction of change variables, there are 32 variables total.

passive IAC This case used passive action acquisition with exploration actions chosen using Intelligent Adaptive Curiosity.

active random NDRC This case used active action acquisition with exploration actions chosen randomly, but with no developmental restriction on learning contingencies. This means the antecedent event of a contingency does not have to be sufficiently reliably predicted by another contingency for the contingency to be learned.

active random NDRA This case used active action acquisition with exploration actions chosen randomly, but with no developmental restriction on learning plans for actions. Thus, the agent does not have to be able to achieve the antecedent event of a contingency with sufficient capability before it can become a plan for an action.

all active random This case used active action acquisition with exploration actions chosen randomly with the change that all actions are acquired using active action acquisition.

To make the evaluation fair between active and passive action learning, during evaluation a contingency must be deterministic to be used as a plan.

6. Results

6.1 Ability to Perform Tasks

The results of the move task are shown in Figure 3. On this task passive action acquisition did better. This is likely because moving the block was sufficiently rare and using active acquisition the maximum number of plans was filled up with plans from inferior contingencies.

The results of the pickup task are shown in Figure 4. How the agent was able to do on this task largely depended on its ability to learn a sufficiently deterministic contingency. The method of **active IAC** did the best. It also had the most experience picking up the block (see Figure 7). The method of **all active random** did poorly, most likely because it spent too much time trying to move the distractor objects (see Figure 8).

6.2 Exploration Using Various Actions

We evaluated how often various exploration techniques explored different actions. Figures 5-7 show the cumulative exploratory calls to various types of actions. Figure 5 shows that Intelligent Adaptive Curiosity has the nice property of not continually ex-

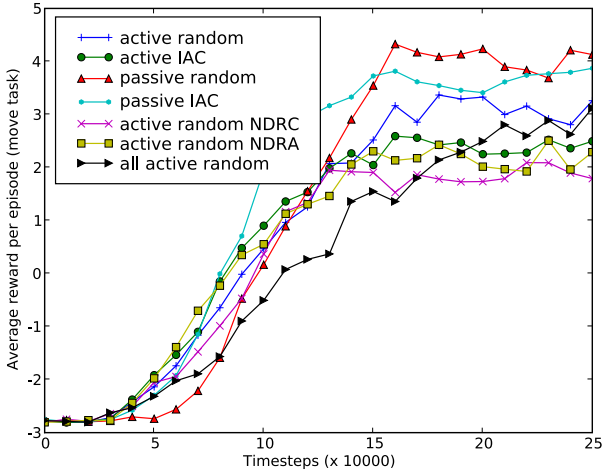


Figure 3: The agent’s ability to move the block increases as it develops. Passive action acquisition outperforms active action acquisition.

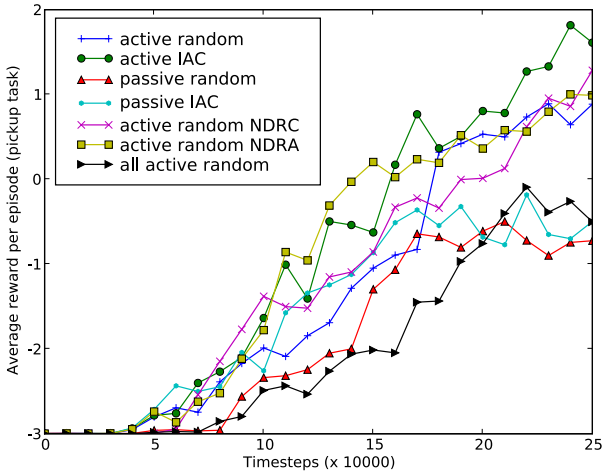


Figure 4: The agent’s ability to pickup the block increases as it develops. In this case active acquisition using curiosity-based exploration performs the best.

ploring actions that the agent has already mastered. Figure 6 shows that Intelligent Adaptive Curiosity causes the agent to explore the relatively difficult action of moving the block. We see this behavior as well with Figure 7 for the case of **active IAC**. Figure 8 shows that the agent should not pursue all actions actively. In this case, **all active random** spends time trying to manipulate the distractor objects.

6.3 Developmental Restrictions

We see in Figures 3 and 4 that **active random** does about as well as **active random NDRC**, which has no developmental restriction on learning contingencies, and **active random NDRA**, which has no developmental restrictions on learning plans for actions. However, we see in Figure 9 that during the early course of the agent’s development that **active**

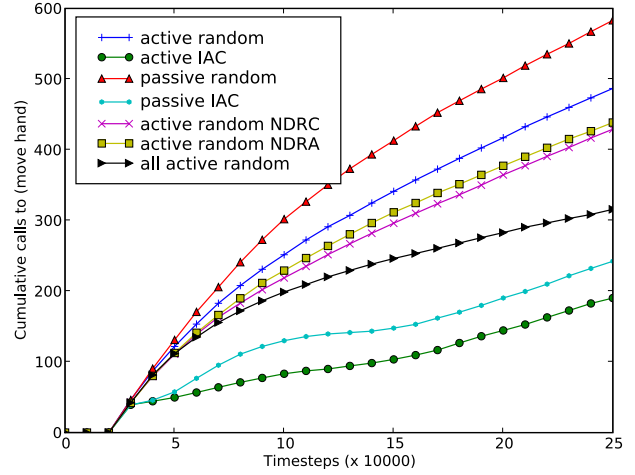


Figure 5: Exploration calls to moving the hand. The curiosity based exploration methods (**active IAC** and **passive IAC**) efficiently use exploration time by making fewer calls to this relatively easy action.

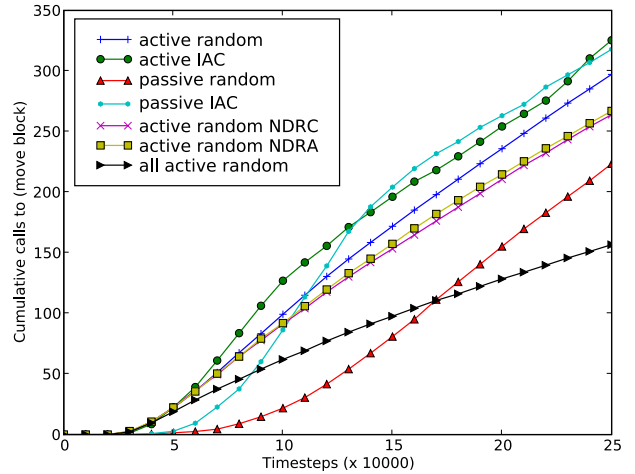


Figure 6: Cumulative exploratory calls to hit the block left, right, or forward.

random has fewer open contingencies, and thus uses fewer resources for those contingencies.

6.4 Exploration Action to Subaction

When the agent first learns an action it is often called as part of exploration. An interesting part of the developmental progression is that these actions are often later called more often as subactions of other actions. We show graphs from the method **active IAC** that compare exploration calls to subaction calls. Figure 10 shows the calls for moving the hand relative to the block (c_x and c_y in Figure 2). These actions are first used more as exploration actions and then later more as subactions.

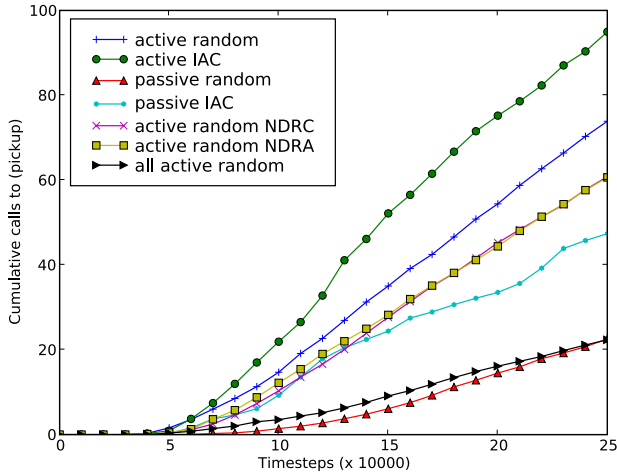


Figure 7: Cumulative exploratory calls to pickup the block. Active acquisition using curiosity-based exploration has the most calls to this complex action.

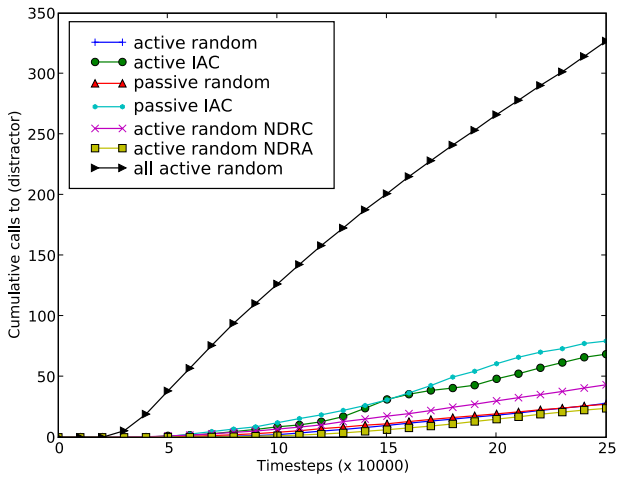


Figure 8: Cumulative exploratory calls to manipulate the floating objects. The method **all active random** has the most calls to this distractor task.

7. Conclusion

In this paper we have presented an evaluation of exploration strategies for learning actions. We found that a combination of active action acquisition and curiosity-based exploration worked best to enable an agent to develop so that it could pick up a block with a sticky mitten. However, we found that active action acquisition was detrimental to the simpler task of moving the block. This is an interesting result that warrants further investigation.

The results indicated that curiosity-based exploration enabled the agent to spend more time exploring the relatively more advanced tasks of moving the block and picking up the block, and enabled the agent to spend less time on the easily mastered task of moving the hand. The results also indicated that we could add restrictions on resources without hindering

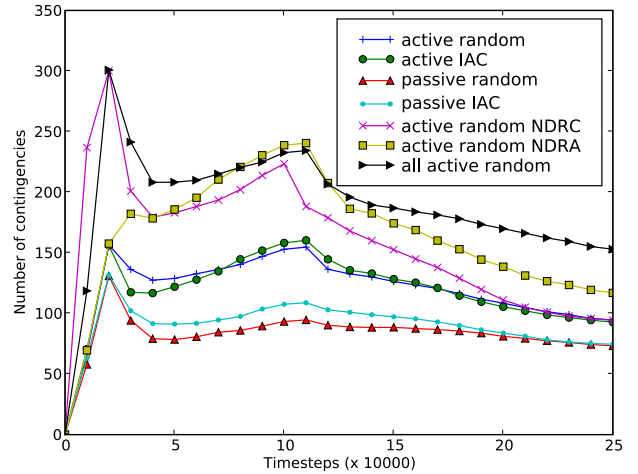


Figure 9: This graph shows that the number of contingencies does not increase without bound. We see two drops in the number of contingencies. The first drop corresponds to learning to move the hand and those actions becoming closed. The second drop corresponds to contingencies being deleted after 100,000 timesteps because they did not become plans to perform actions.

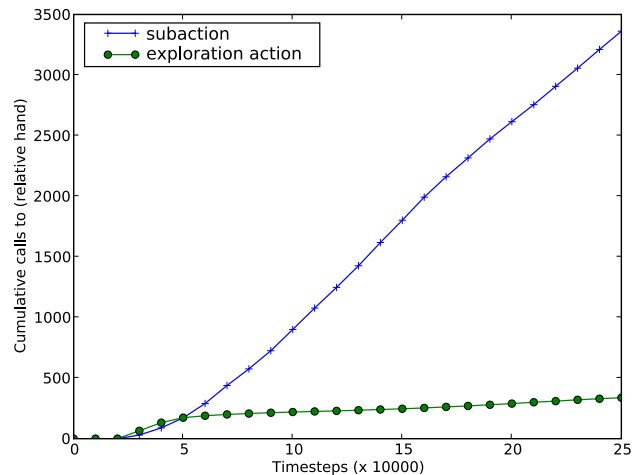


Figure 10: Action calls to moving the hand relative to the block for the method **active IAC**. This task is first called mostly as exploration and then later more as a subaction.

learning.

There are, of course, other approaches that enable agents to learn actions. For example, Metta and Fitzpatrick (2003) focus on learning affordances (Gibson, 1979). However, the focus of QLAP is on enabling an agent to autonomously learn actions from motor primitives. The results presented here will most closely apply to models where the agent picks which action to learn during the process of autonomous development.

Acknowledgements

This work has taken place in the Intelligent Robotics Lab at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Intelligent Robotics lab is supported in part by grants from the Texas Advanced Research Program (3658-0170-2007), and from the National Science Foundation (IIS-0413257, IIS-0713150, and IIS-0750011). The authors would also like to thank Lewis Fishgold and the anonymous reviewers for helpful comments and suggestions.

References

- Adolph, K. E. and Joh, A. S. (2007). Motor development: How infants get into the act. In Slater, A. and Lewis, M., (Eds.), *Introduction to infant development*. Oxford University Press.
- Berlyne, D. (1965). *Structure and Direction in Thinking*. John Wiley and Sons, Inc., New York.
- Bonarini, A., Lazaric, A., and Restelli, M. (2006). Incremental Skill Acquisition for Self-Motivated Learning Animats. *Lecture Notes in Computer Science*, 4095:357.
- DeCasper, A. J. and Carstens, A. (1981). Contingencies of stimulation: Effects of learning and emotions in neonates. *Infant Behavior and Development*, 4:19–35.
- Gergely, G. and Watson, J. (1999). Early socio-emotional development: Contingency perception and the social-biofeedback model. *Early social cognition: Understanding others in the first months of life*, pages 101–136.
- Gibson, E. (1988). Exploratory behavior in the development of perceiving, acting, and the acquiring of knowledge. *Annual review of psychology*, 39(1):1–42.
- Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Lawrence Erlbaum Associates, New Jersey, USA.
- Huang, X. and Weng, J. (2002). Novelty and Reinforcement Learning in the Value System of Developmental Robots. *Proc. 2nd Inter. Workshop on Epigenetic Robotics*.
- Klein, J. (2003). Breve: a 3d environment for the simulation of decentralized systems and artificial life. In *Proc. of the Int. Conf. on Artificial Life*.
- Kuipers, B. (1994). *Qualitative Reasoning*. The MIT Press, Cambridge, Massachusetts.
- Marshall, J., Blank, D., and Meeden, L. (2004). An emergent framework for self-motivation in developmental robotics. *Proc. of the 3rd Int. Conf. on Development and Learning (ICDL 2004)*.
- Metta, G. and Fitzpatrick, P. (2003). Early integration of vision and manipulation. *Adaptive Behavior*, 11(2):109–128.
- Mugan, J. and Kuipers, B. (2007). Learning to predict the effects of actions: Synergy between rules and landmarks. In *Proc. of the Int. Conf. on Development and Learning*.
- Mugan, J. and Kuipers, B. (2008). Towards the application of reinforcement learning to undirected developmental learning. In *Proc. of the Int. Conf. on Epigenetic Robotics*.
- Mugan, J. and Kuipers, B. (2009). Autonomously learning an action hierarchy using a learned qualitative state representation. In *Proc. of the Int. Joint Conf. on Artificial Intelligence*.
- Needham, A., Barrett, T., and Peterman, K. (2002). A pick-me-up for infants’ exploratory skills: Early simulated experiences reaching for objects using ‘sticky mittens’ enhances young infants’ object exploration skills. *Infant Behavior and Development*, 25(3):279–295.
- Oudeyer, P., Kaplan, F., and Hafner, V. (2007). Intrinsic Motivation Systems for Autonomous Mental Development. *Evolutionary Computation, IEEE Transactions on*, 11(2):265–286.
- Piaget, J. (1952). *The Origins of Intelligence in Children*. Norton, New York.
- Ryan, R. M. and Deci, E. L. (2000). Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary Educational Psychology*, 25:54–67.
- Schmidhuber, J. (1991). Curious model-building control systems. In *Proc. Int. Joint Conf. on Neural Networks*.
- Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning*. MIT Press, Cambridge MA.
- Sutton, R. S., Precup, D., and Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1-2):181–211.

Can imprecise internal motor models explain the ataxic hand trajectories during reaching in young infants?

Francesco Nori
francesco.nori@iit.it[†]

Giulio Sandini
giulio.sandini@iit.it[†]

Jürgen Konczak
jkonczak@umn.edu[‡]

[†] Dep. of Robotics, Brain and Cognitive Sciences,
Italian Institute of Technology
Via Morego 30, Genova

[‡] Human Sensorimotor Control Lab.
University of Minnesota, U.S.A.
Minnesota, U.S.A.

Abstract

The first reaching movements of human infants lack limb coordination leading to ataxic-like hand trajectories. Kinematically, these early trajectories are characterized by multiple peaks in the hand velocity profile which gradually decrease in frequency during development. In this paper we explore the hypothesis that the jerky hand trajectories seen in early infancy can be the result of imprecise internal motor models. Results from our simulation suggest that imprecise estimations of multi-joint inter-segmental torques (e.g., Coriolis forces) by the controller may induce multi-peak hand velocity profiles. When the system was allowed to use delayed peripheral feedback (300 ms after reaching onset), the resulting kinematics began to resemble those seen in early infancy. This suggests that the output of an imprecise internal model of limb dynamics coupled with delayed feedback may be sufficient to explain early human hand trajectories. Our data provide an alternative to previous hypotheses theorising jerky trajectories as the result of concatenated mini ballistic movements.

1. Introduction

The first goal-directed movements of young infants at the age of 4-5 months lack coordination which gives them an ataxic appearance. The lack of proximal joint coordination leads to multiple movement units of the hand during early attempts to reach for objects (von Hofsten, 1979, Konczak et al., 1995, Berthier, 1999). That is, the hand is not moved

in a smooth, stereotypical fashion, but its trajectory is jerky showing numerous changes in direction. Previous research on motor control in early infancy tried to explain this phenomenon on the basis of a faulty planning mechanism or as a compensatory motor strategy trying to overcome the lack of control (von Hofsten, 1992, Berthier, 1999). In this view, the observed segmented trajectories are a series of concatenated mini ballistic trajectories. At the end of each movement segment, the control system uses either afferent information to update the initial plan and to correct the chosen joint paths (on-line feedback control) or based on the experience from previous failures it tries not to perform a single large-amplitude reach, but executes a series of planned sub-movements. Each sub-movement is viewed as a perfect trajectory following the minimum-jerk principle (Berthier, 1999).

While such view could explain the appearance of multiple hand velocity profiles seen in infant reaching, it relies on a set of assumptions. First, the infant's motor system is seen as not being ready to deal with the peripheral bio-mechanics, but it is capable of using peripheral feedback very fast and effectively. Second, it assumes that higher cognitive structures "know" about this control predicament and induce the motor system to adapt a compensatory strategy by which a sequence of small amplitude movements are performed in order to approach a desired objects.

We here present an alternative view that may explain the phenomenon of dyscoordination in early infancy without recurrence to a cognitive mechanism. First, the assumption is made voluntary sensorimotor control is based on movement planning. Second, limb mechanics are controlled by the central nervous system. We make no specific assumption about

the exact neuroanatomical location of these control structures, although it is known that in humans they involve the motor cortices, the cerebellum, basal ganglia and the spinal cord. One way to control limb mechanics is that the neural controller has acquired an internal model of the peripheral mechanics which implies that it operates like an inverse model of the body. There is increasing empirical evidence that is consistent with the view that human motor systems uses inverse models for the multi-joint limb control (Gandolfo et al., 1996, Wolpert et al., 1998) and that these models become more precise during development (Jansen-Osmann et al., 1997). The question arises of how the infant's brain acquires an inverse model? In theory, it could be genetically determined and be operational at birth. This is unlikely knowing that early reaches show clear signs of dyscoordination, which implies that infant internal motor models at best contain imprecise estimations of the real limb parameters at birth or that the associated planning agencies are not fully functional in early infancy or both. An alternative view is that internal models are not pre-wired in the brain but are acquired through a process of parallel exploration and calibration (Metta et al., 1999). Assuming that the infant brain has acquired some form of an inverse motor model before the onset of goal-directed behaviour (e.g. through "motor babbling"), the question arises whether motor performance is susceptible to imprecise estimations of specific limb mechanical parameters. For example, given the rapid growth during the first postnatal months, could it be that an over- or underestimation of inertia or mass impacts on hand trajectory formation? The purpose of this paper is to investigate the hypothesis that early human reaching trajectories are the result of imprecise estimations of limb dynamics. We compared the simulated reaching kinematics generated by an artificial neural controller consisting of an incorrect inverse model of the human limb dynamics to the kinematics of human infants observed at the onset of goal-directed reaching (Konczak et al., 1995, Konczak and Dichgans, 1997). An incorrect controller implies that only imprecise estimations of limb mechanical parameters are available to the control system. To test the effects of an incorrect inverse model on trajectory formation, we developed a 4 degrees of freedom arm simulation that received adult-like kinematics as movement plan. Assuming a controller with a correct internal model of the arm dynamics, we show that the generated movement trajectories are identical to the planned ones. We then manipulated limb parameters such as inertia, interaction torques or gravity and compared the resulting kinematics with the planned trajectories.

Though the scope of our considerations within the current paper is limited to infant motion planning,

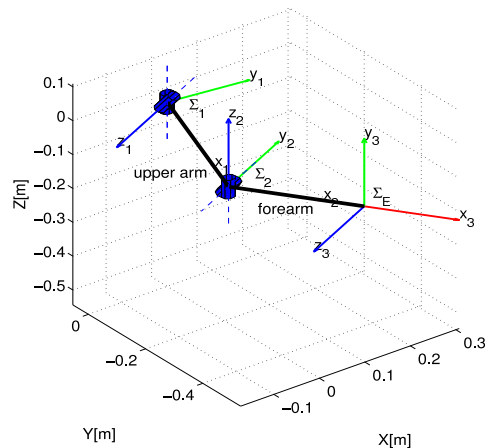


Figure 1: Three dimensional representation (using a MATLAB[®] toolbox (Corke, 1996)) of the arm model kinematics in the configuration $q = [-\frac{\pi}{3}, -\frac{\pi}{3}, \frac{\pi}{3}, 0]^\top$. The shoulder and the elbow are represented by two universal joints (two degrees of freedom each). The picture shows also the link reference frames (Σ_1 , Σ_2) and the end effector reference frame (Σ_E). The root reference frame Σ_o corresponds to the plot axes.

it will be evident that the overall framework have important implications in the field of robotics, with specific concern to developmental approaches. As a matter of fact, in a wide sense, our research considers the relative role of feedback and feedforward motor control with specific attention to various stages of development.

2. Method

This section describes the basic setup of the simulation. The arm model is composed of two segments, nominally the upper arm and the forearm. Each segment has two joints (two at the shoulder and two at the elbow), so that the overall structure has four degrees of freedom (see figure 1 for a sketch of the kinematic structure). The angular position of the i -th degree of freedom will be denoted q_i and the overall arm configuration $q = [q_1, q_2, q_3, q_4]^\top \in \mathbb{R}^4$. According to the Denavit-Hartenberg convention, we define an inertial reference frame Σ_o and associate two reference frames (Σ_1 , Σ_2) to each of the segments (see figure 1). The rigid roto-translation from Σ_i to Σ_o will be denoted oT_i and computed as follows:

$$\begin{aligned} {}^oT_1 &= T_1(q_1)T_2(q_2); \\ {}^oT_2 &= T_1(q_1)T_2(q_2)T_3(q_3)T_4(q_4); \end{aligned}$$

i	α_i	a_i	d_i
1	0	0	0
2	$\frac{\pi}{2}$	0	0
3	0	l_1	0
4	$-\frac{\pi}{2}$	0	0

Table 1: Kinematic parameters that describes the arm forward kinematics as a function of simple anthropometric measurement, i.e. the upper arm length l_1 .

with $T_i(q_i)$ represented by:

$$T_i = \begin{bmatrix} \cos q_i & -\sin q_i & 0 & a_i \\ \cos \alpha_i \sin q_i & \cos \alpha_i \cos q_i & -\sin \alpha_i & -\sin \alpha_i d_i \\ \sin \alpha_i \sin q_i & \sin \alpha_i \cos q_i & \cos \alpha_i & \cos \alpha_i d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (1)$$

with parameters α_i , a_i and d_i given in table 1 and computed from simple anthropometric measurement.

The differential equation used to describe the arm dynamics is the following (Murray et al., 1994):

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau, \quad (2)$$

where $q \in \mathbb{R}^4$ is the vector of generalised coordinates (i.e. angular displacements) describing the arm posture, $\tau \in \mathbb{R}^4$ is the associated vector of generalised forces (i.e. joint torques) describing the muscle activation and M , C and G are the inertia, Coriolis and gravitational component of the dynamical forces acting on the arm. The analytical expressions for all these components have been constructed as proposed in (Yoshikawa, 1990) page 94. Their numerical values have been computed from simple anthropometric measurements following the approach proposed by (Schneider and Zernicke, 1992). The interested reader can find the complete analytical derivation in Appendix A.

2.1 Control strategy

Given the applied control strategy $\tau(t, q, \dot{q})$, $t \in [0, T]$ and the system initial condition $[q(0), \dot{q}(0)] = [q_0, \dot{q}_0]$ (typically $\dot{q}_0 = 0$), the resulting reaching movements have been simulated by integrating (2) with MATLAB SIMULINK[®]. Within the current framework, given a desired movement to be performed $q_d(t)$, $t \in [0, T]$ the applied control strategy is composed of a feedforward component τ_{ff} (relying on an approximation \hat{M} , \hat{C} , \hat{G} of the system dynamics) and a feedback component¹ τ_{fb} :

$$\tau(t, q, \dot{q}) = \tau_{ff}(t) + \tau_{fb}(t, q, \dot{q}) \quad (3)$$

¹In order to simplify the analysis we will not consider generic feedback gain matrices $K_p \in \mathbb{R}^{4 \times 4}$ and $K_v \in \mathbb{R}^{4 \times 4}$. We will define instead $K_p = k_p I$ and $K_v = k_v I$ where I is the 4×4 identity matrix. The effects of feedback have been evaluated by varying the scalar gains k_p and k_v in the intervals $k_p \in [0, 1000]$ and $k_v \in [0, 100]$.

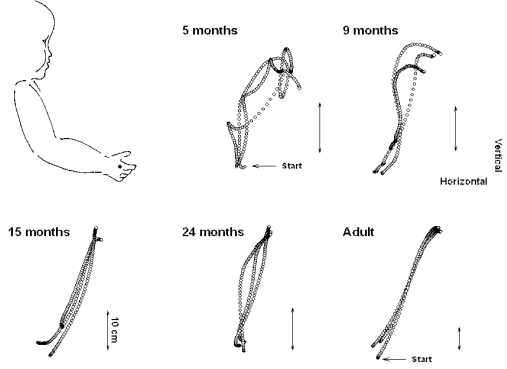


Figure 3: Exemplar hand reaching trajectories at different stages of development (Konczak and Dichgans, 1997). Remarkably, the non smooth profiles displayed by early infants are gradually replaced by straight trajectories characterized by roughly constant curvature.

where:

$$\tau_{ff}(t) = \hat{M}(q_d)\ddot{q}_d + \hat{C}(q_d, \dot{q}_d)\dot{q}_d + \hat{G}(q_d), \quad (4)$$

$$\tau_{fb}(q, \dot{q}, t) = K_p(q - q_d) + K_v(\dot{q} - \dot{q}_d). \quad (5)$$

The desired trajectory $q_d(t)$, $t \in [0, T]$ (i.e., the movement plan) was extracted from a single adult goal-directed reaching movement captured at 100 Hz and interpolated with splines². The use of an adult reaching profile as input for the simulation was based on the notion that it best reflected a biologically plausible movement plan.

3. Results

We systematically evaluated the effect of an imprecise controller for a wide range of incorrect dynamical parameters to obtain a sense of how sensitive the system was to imprecise estimations of inertial, gravitational and inter-segmental torques). The resulting artificial trajectories were then compared with the reaching trajectories of one human infants at different developmental stages (see Figure 2).

At first we verified that if the approximated dynamics (\hat{M} , \hat{C} , \hat{G}) perfectly match the system dynamics (M , C , G) then the system follows the planned trajectory, i.e. $q \equiv q_d$ (see Figure 4).

As a second step, we manipulated the feedforward component of the controller in order to visualize the effect of a mismatch between approximated and real dynamics. As shown in Figure 1 the development of reaching in humans is associated with a decrease in a

²The use of spline interpolation allows to obtain a continuous time function with sufficient smoothness properties for performing the double derivative operation in order to get $\dot{q}_d(\cdot)$ and $\ddot{q}_d(\cdot)$ from $q_d(\cdot)$.

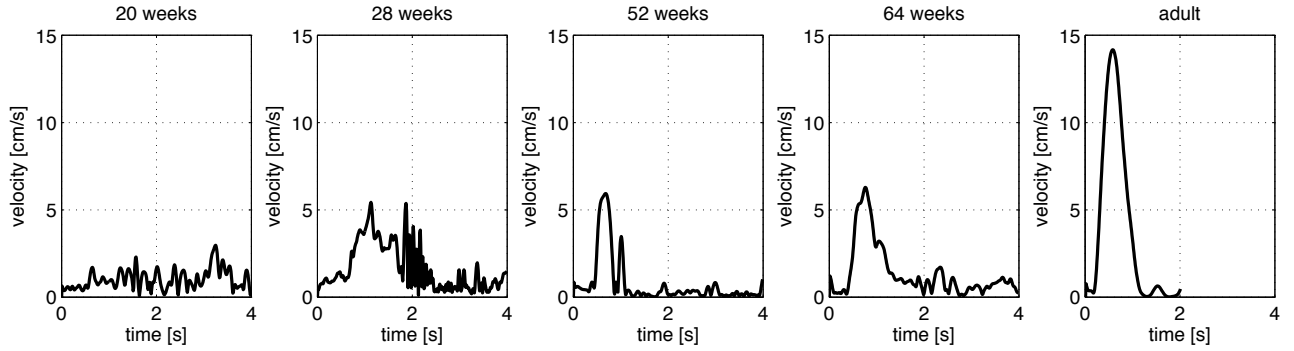


Figure 2: Exemplar 3D resultant hand velocity profiles of reaching movements at different stages of infant development (Konczak et al., 1995); the right graph shows the typical bell shaped velocity profile of an adult, which was used as movement plan.

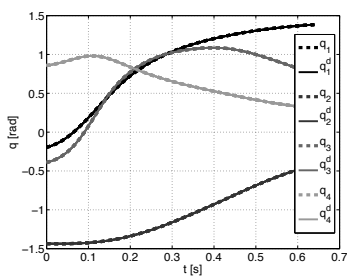


Figure 4: Joint angle trajectories corresponding to a typical reaching movement (data captured from an adult). Solid lines, q , correspond to captured data. Dashed lines, q_d , correspond to the trajectories performed when the feedforward controller perfectly inverts the dynamics of the artificial arm ($\hat{M} = M$, $\hat{C} = C$, $\hat{G} = G$). Clearly, resulting trajectories corresponded to the desired one ($q \equiv q_d$).

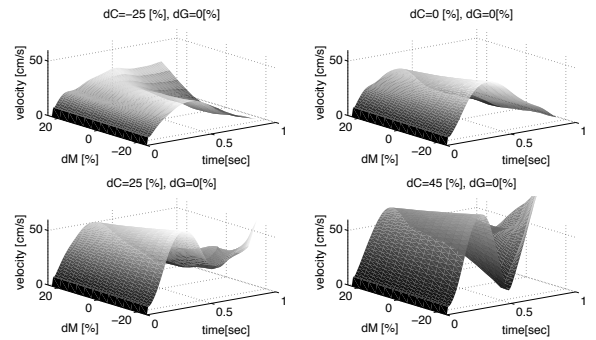


Figure 6: Four examples ($dC = -25\%$, $dC = 25\%$, $dC = 50\%$ and correct estimate $dC = 0\%$) of Coriolis forces miscalculation in a pure feedforward controller as a function of changing estimates of inertial torques. Remarkably, relevant overestimates of the Coriolis component produce double peak velocity profiles which were not present in case of inaccurate gravitational and inertial components.

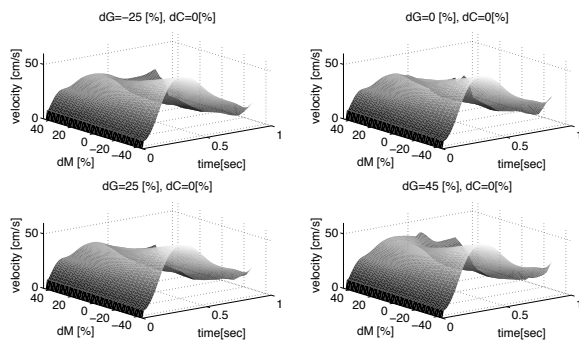


Figure 5: Effects of incorrect controllers estimates in the inertial and gravitational components ($\hat{M} \neq M$, $\hat{G} \neq G$ but $\hat{C} = C$) on the hand velocity profiles when applying a pure feedforward control ($\tau = \tau_{ff}$). Shown are four estimates of gravitational torque ($dG = -25\%$, $dG = 25\%$, $dG = 50\%$ and correct estimate $dG = 0\%$). It is evident that inaccurate gravitational and inertial component do not produce evident double peak velocity profiles.

the number of peaks of the hand velocity. One main result of the simulation was that similar multi-peak velocity profiles were generated when the controller values overestimated the Coriolis forces. This is evident in Figure 5 and 6 where we visualized the effects of a pure feedforward controller on the resulting hand velocity profiles. In particular, Figure 5 shows the effects of unmatched inertial and gravitational terms. The horizontal axes refer to time in seconds and percent error dM in the inertia matrix approximation:

$$\hat{M}(q) = \left(1 + \frac{dM}{100}\right)M(q).$$

Vertical axis (in gray scale) represents the hand tangential velocity and the four different plots refer to different values of the percent error dG in the Coriolis component:

$$\hat{G}(q, \dot{q}) = \left(1 + \frac{dG}{100}\right)G(q, \dot{q}).$$

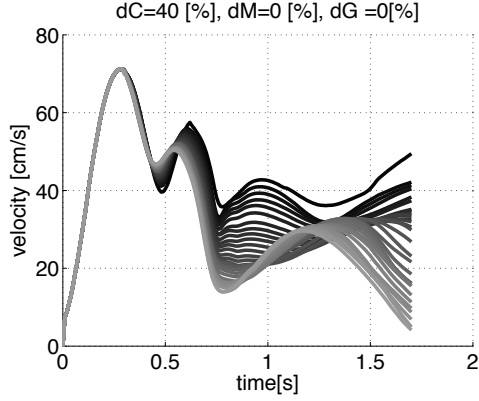


Figure 7: The effect of delayed feedback on hand trajectory formation. Shown are velocity profiles when the controller approximated dynamics did not match the Coriolis arm dynamics. (\hat{C} overestimates C of a 40%). After 300 ms position feedback became available. The various profiles show the effect of different feedback gains: the darker the line the smaller the overall feedback gain $k = k_p + k_v$ is (see footnote 1 at page 3 for the definitions of k_p and k_v).

Similarly, Figure 6 shows the effects of unmatched inertial and Coriolis components with analogous definition for the percent error dC in the Coriolis component. From these pictures it is evident the hand velocity profiles remained single-peaked for incorrect estimation of the inertial and gravity components of the dynamics. In case of relevant errors in the Coriolis approximation (bottom right corner of Figure 6), multi-peak velocity profiles are generated by applying the pure feedforward component of the controller.

As a third step, we tested the effects of feedback on trajectory formation. Feedback became available after 300 ms, which approximately corresponds to delay of visual feedback in humans. It was observed that the combined effect of feedback and approximation errors in the Coriolis part of the feedback controller might result in additional velocity peaks (see Figure 7).

4. Discussion

Infants show ataxic hand trajectories with multiple movement reversals when attempting their first goal-directed reaches at around the postnatal age of 4-5 months. This observed lack of multi-joint coordination is not monocausal, but likely the result of complex interactions within the neuromuscular system. Cognitive accounts of motor development explained the lack of coordination among limb segments not primarily as a failure of a controller, but as a part of strategy of higher motor centers to overcome the deficiencies in low-level control (Berthier 1996). The results our study indicate that a neural controller

with imprecise estimations of the true limb dynamics may generate ataxic endpoint trajectories that are comparable to those observed in human infants around the onset of goal-directed reaching. Especially controller overestimation of the actual Coriolis forces will induce multiple velocity profiles. The use of peripheral feedback will ensure that the hand eventually reaches the target, when the feedback gain is relatively high. It needs to be clear that these results were based on the providing a "perfect" plan to the controller. Thus, one can criticize that the results of the current simulation are limited, because this assumes that movement planning agencies in infants develop earlier than the the controller. There is no firm evidence in place to fully support this assumption. However, we here wanted to make the point that an imprecise controller alone can result in ataxic hand trajectories. Thus, the coordination deficit seen in early hand trajectory formation can be viewed as a the result of an imprecise controller with no need to assume the involvement of higher cognitive functions. The kinematic effects of an incorrect or noisy plan in conjunction with an incorrect inverse model of the plant likely enhances dyscoordination. A systematic investigation of the effect of imprecise planning on hand trajectory formation will be a next step in our series of simulations.

Acknowledgements

This work was supported by European Commission projects RobotCub (IST- 004370) and ITALK (ICT-214668).

A Dynamic equation computation

In this section we describe in details how to compute the matrices M , C and the vector G . Specifically, the (i, j) component of the matrix M denoted M_{ij} has been computed as follows:

$$M_{ij}(q) = \text{tr} \left(J_{1j}(q) \hat{H}_1 J_{1i}^\top(q) + J_{2j}(q) \hat{H}_2 J_{2i}^\top(q) \right); \quad (6)$$

similarly, the i component of the vector $C\dot{q}$ (denoted h_i) has been computed as:

$$h_i(q, \dot{q}) = \sum_{j,m=1}^4 \text{tr} \left(\frac{\partial J_{1j}(q)}{\partial q_m} \hat{H}_1 J_{1i}^\top(q) + \frac{\partial J_{2j}(q)}{\partial q_m} \hat{H}_2 J_{2i}^\top(q) \right); \quad (7)$$

finally, the i component of the vector G denoted G_i is:

$$G_i(q) = -m_1 \mathbf{g}^\top J_{1i}(q) s_1 - m_2 \mathbf{g}^\top J_{2i}(q) s_2, \quad (8)$$

being \mathbf{g} the gravitational vector expressed in Σ_o . In the formulas above, the matrices \hat{H}_1 and \hat{H}_2 are the

pseudo inertia matrices of the upper arm and forearm respectively; matrices J_{1i} and J_{2i} are the derivative of the rigid roto-translations oT_1 and oT_2 :

$$J_{1i} = \frac{\partial {}^oT_1}{\partial q_i} \quad J_{2i} = \frac{\partial {}^oT_2}{\partial q_i} \quad i = 1, \dots, 4,$$

being q_i the i -th component of the vector q . The value of \hat{H}_1 and \hat{H}_2 depends only on s_1, s_2 (centers of mass position), m_1, m_2 (segment masses) and \hat{I}_1, \hat{I}_2 (inertia tensor of the segments with respect to the segment reference frame) according to ($k = 1, 2$):

$$\hat{H}_k = \begin{bmatrix} \frac{-\hat{i}_k^x + \hat{i}_k^y + \hat{i}_k^z}{2} & \hat{H}_k^{xy} & \hat{H}_k^{xz} & m_k s_k^x \\ \hat{H}_k^{xy} & \frac{\hat{i}_k^x - \hat{i}_k^y + \hat{i}_k^z}{2} & \hat{H}_k^{yz} & m_k s_k^y \\ \hat{H}_k^{xz} & \hat{H}_k^{yz} & \frac{\hat{i}_k^x + \hat{i}_k^y - \hat{i}_k^z}{2} & m_k s_k^z \\ m_k s_k^x & m_k s_k^y & m_k s_k^z & m_k \end{bmatrix} \quad (9)$$

The numerical values of all these quantities have been obtained from simple anthropometric measurements following the approach proposed in (Schneider and Zernicke, 1992). Specifically table 2 reports all the equations that can be used in (9) to compute \hat{H}_1 and \hat{H}_2 starting from the segments lengths (l_1, l_2), body mass (b) and segments circumference (c_1, c_2). Finally, the value of J_{ij} has been computed as:

$$\begin{aligned} J_{11} &= T_1 \Delta T_2; & J_{12} &= T_1 T_2 \Delta; \\ J_{13} &= 0; & J_{14} &= 0; \\ J_{21} &= T_1 \Delta T_2 T_3 T_4; & J_{22} &= T_1 T_2 \Delta T_3 T_4; \\ J_{23} &= T_1 T_2 T_3 \Delta T_4; & J_{24} &= T_1 T_2 T_3 T_4 \Delta; \end{aligned}$$

with T_i given as in (1) and:

$$\Delta = \begin{bmatrix} 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Similarly:

$$\begin{aligned} \frac{\partial J_{11}}{\partial q_1} &= T_1 \Delta^2 T_2; & \frac{\partial J_{11}}{\partial q_2} &= T_1 \Delta T_2 \Delta; \\ \frac{\partial J_{11}}{\partial q_3} &= 0; & \frac{\partial J_{11}}{\partial q_4} &= 0; \\ \frac{\partial J_{12}}{\partial q_1} &= T_1 \Delta T_2 \Delta; & \frac{\partial J_{12}}{\partial q_2} &= T_1 T_2 \Delta^2; \\ \frac{\partial J_{12}}{\partial q_3} &= 0; & \frac{\partial J_{12}}{\partial q_4} &= 0; \end{aligned}$$

and:

$$\begin{aligned} \frac{\partial J_{21}}{\partial q_1} &= T_1 \Delta^2 T_2 T_3 T_4; & \frac{\partial J_{21}}{\partial q_2} &= T_1 \Delta T_2 \Delta T_3 T_4; \\ \frac{\partial J_{21}}{\partial q_3} &= T_1 \Delta T_2 T_3 \Delta T_4; & \frac{\partial J_{21}}{\partial q_4} &= T_1 \Delta T_2 T_3 T_4 \Delta; \\ \frac{\partial J_{22}}{\partial q_1} &= \frac{\partial J_{21}}{\partial q_2}; & \frac{\partial J_{22}}{\partial q_2} &= T_1 T_2 \Delta^2 T_3 T_4; \\ \frac{\partial J_{22}}{\partial q_3} &= T_1 T_2 \Delta T_3 \Delta T_4; & \frac{\partial J_{22}}{\partial q_4} &= T_1 T_2 \Delta T_3 T_4 \Delta; \\ \frac{\partial J_{23}}{\partial q_1} &= \frac{\partial J_{21}}{\partial q_3}; & \frac{\partial J_{23}}{\partial q_2} &= \frac{\partial J_{22}}{\partial q_3}; \\ \frac{\partial J_{23}}{\partial q_3} &= T_1 T_2 T_3 \Delta^2 T_4; & \frac{\partial J_{23}}{\partial q_4} &= T_1 T_2 T_3 \Delta T_4 \Delta; \\ \frac{\partial J_{24}}{\partial q_1} &= \frac{\partial J_{21}}{\partial q_4}; & \frac{\partial J_{24}}{\partial q_2} &= \frac{\partial J_{22}}{\partial q_4}; \\ \frac{\partial J_{24}}{\partial q_3} &= \frac{\partial J_{23}}{\partial q_4}; & \frac{\partial J_{24}}{\partial q_4} &= T_1 T_2 T_3 T_4 \Delta^2. \end{aligned}$$

References

- Berthier, N. E. (1999). Learning to reach: A mathematical model. *Developmental Psychology*, 32:811–823.
- Corke, P. (1996). A robotics toolbox for MATLAB. *IEEE Robotics and Automation Magazine*, 3(1):24–32.
- Gandolfo, F., Mussa-Ivaldi, F. A., and Bizzi, E. (1996). Motor learning by the field approximation. In *Proceedings of the National Academy of Science*, volume 93, pages 3843–3846.
- Jansen-Osmann, P., Richter, S., Konczak, J., and Kalveram, K. (1997). Force adaptation transfers to untrained workspace regions in children: Evidence for developing inverse dynamic models. *Experimental Brain Research*, 143:212–220.
- Konczak, J., Borutta, M., Topka, H., and Dichgans, J. (1995). Development of goal-directed reaching in infants: Hand trajectory formation and joint force control. *Experimental Brain Research*, 106:156–168.
- Konczak, J. and Dichgans, J. (1997). Goal-directed reaching: development toward stereotypic arm kinematics in the first three years of life. *Experimental Brain Research*, 117:346–354.
- Metta, G., Sandini, G., and Konczak, J. (1999). A developmental approach to visually-guided reaching in artificial systems. *Neural Networks*, 12:1413–1427.

Upper	$m_1 = 1.2249 \times 10^{-2}b + 1.3067 \times 10^0l_1 + 9.8645 \times 10^{-1}c_1 - 1.9376 \times 10^{-1}$
	$I_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{m_1 l_1^2}{3} & 0 \\ 0 & 0 & \frac{m_1 l_1^2}{3} \end{bmatrix}$
	$\hat{I}_1 = \begin{bmatrix} \hat{I}_2^x & -\hat{H}_2^{xy} & -\hat{H}_2^{xz} \\ -\hat{H}_2^{xy} & \hat{I}_2^y & -\hat{H}_2^{yz} \\ -\hat{H}_2^{xz} & -\hat{H}_2^{yz} & \hat{I}_2^z \end{bmatrix}$
	$= I_1 + m_1 \begin{bmatrix} 0 & -s_k^z & s_k^y \\ s_k^z & 0 & -s_k^x \\ -s_k^y & s_k^x & 0 \end{bmatrix}$
$s_1 = \begin{bmatrix} s_1^x \\ s_1^y \\ s_1^z \end{bmatrix} = \begin{bmatrix} 0.4428 \times l_1 \\ 0 \\ 0 \end{bmatrix}$	
Fore	$m_2 = 5.2671 \times 10^{-3}b + 9.7584 \times 10^{-1}l_2 + 1.1492 \times 10^0c_2 - 1.6886 \times 10^{-1}$
	$I_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{m_2 l_2^2}{3} & 0 \\ 0 & 0 & \frac{m_2 l_2^2}{3} \end{bmatrix}$
	$\hat{I}_2 = \begin{bmatrix} \hat{I}_2^x & -\hat{H}_2^{xy} & -\hat{H}_2^{xz} \\ -\hat{H}_2^{xy} & \hat{I}_2^y & -\hat{H}_2^{yz} \\ -\hat{H}_2^{xz} & -\hat{H}_2^{yz} & \hat{I}_2^z \end{bmatrix}$
	$= I_2 + m_2 \begin{bmatrix} 0 & -s_2^z & s_2^y \\ s_2^z & 0 & -s_2^x \\ -s_2^y & s_2^x & 0 \end{bmatrix}$
$s_2 = \begin{bmatrix} s_2^x \\ s_2^y \\ s_2^z \end{bmatrix} = \begin{bmatrix} 0.4541 \times l_2 \\ 0 \\ 0 \end{bmatrix}$	

Table 2: Dynamical parameters as a function of simple anthropometric measurements (Schneider and Zernicke, 1992): $b[kg]$ = infant body mass ; $l_k[m]$ = segment length; $c_k[m]$ = segment circumference. Values of these anthropometric measurements have been measured on subjects. Dynamical parameters have been computed accordingly: $m_k[kg]$ = segmental mass; $s_k[m]$ = center of mass position with respect to the segment reference frame; $I_k[kg \cdot m^2]$ = inertia tensor with respect to the frame with its origin at the center of mass. Note that in the calculation of the inertia tensors segments have been approximated with one dimensional rods.

Murray, R. M., Li, Z., and Sastry, S. S. (1994). *A Mathematical Introduction to Robotic Manipulation*. CRC Press.

Schneider, K. and Zernicke, R. (1992). Mass, center of mass, and moment of inertia estimates for infant limb segments. *Journal of Biomechanics*, 25:145–148.

von Hofsten, C. (1979). Development of visually directed reaching: the approach phase. *Journal of Human Movement Studies*, 5:160–178.

von Hofsten, C. (1992). *The gearing of early reaching to the environment.*, volume 87. Elsevier, Amsterdam.

Wolpert, D. M., Miall, R. C., and Kawato, M. (1998). Internal models in the cerebellum. *Trends in Cognitive Sciences*, 2:338–347.

Yoshikawa, T. (1990). *Foundations of Robotics: Analysis and Control*. MIT Press.

Learning of Situation Dependent Prediction toward Acquiring Physical Causality

Masaki Ogino Tetsuya Fujita Sawa Fuke Minoru Asada
JST ERATO Asada Synergistic Intelligence Project
Yamadaoka 2-1, Suita, Osaka 565-0871, Japan
Osaka University
Graduate School of Engineering, Department of Adaptive Machine Systems,
Yamadaoka 2-1, Suita, Osaka 565-0871, Japan

Abstract

Physical causality is one of the most important knowledge that human babies learn first after birth through interaction with the surrounding environment. The properties of object movement changes depending on the situation, and so the agent should change its prediction. This paper proposes a learning model which predicts the movement of an attended object depending on the environment around the object. The predictor is formed by three main layered associative modules: (a) an *environment* module, which recognizes the attended object and its surrounding environment; (b) a *predictor module*, which anticipates the movement of the attended object depending on the surrounding environment; (c) an *attention module* which implements bottom-up and top-down attention processes. The proposed method is applied to the robot, and its prediction faculty and adaptability are examined in the simulation and actual environment.

1. Introduction

All infants are physicists. From the day of the birth, they begin to learn the fundamental properties of the world step by step through the interaction with the surrounding environments. The one of the important indices for their progress is object permanence; even when an attended object will be occluded from the view by an obstacle, they can understand the object will not be lost from the world and remain behind obstacles. Although Piaget firstly proposed that infants can acquire object permanence after 18 month old (Piaget, 1954), other researchers has shown that infants can pass object permanence task before one year old (Baillargeon et al., 1985) (Baillargeon and DeVos, 1991). In developmental cognitive robotics, some learning models

are proposed to explain the results shown in these experiments with more restricted facilities (Schlesinger, 2003) (Lovett and Scasselatti, 2004). Although an actual mechanism that enables an infant to show these behaviors even in such a early stage is still unknown, how such higher concepts about the world as object continuity and impossibility can be learned autonomously is also an interesting problem in a robot area (Fitzpatrick et al., 2008). One of the fundamental faculties to realize the higher concepts about the world is to model the phenomena effectively for appropriate prediction. In this paper, we propose a learning model that enables a robot to learn the prediction of the object movement depending on the situation. For this purpose, multiple Restricted Boltzmann Machines (RBMs) (Hinton et al., 2006) are adopted, which can be used for both unsupervised and supervised learnings.

Moreover, with this learning model we treat the problem on the relationship between attention and learning. Attention is thought to consist of two processes; bottom-up and top-down attention (Knudsen, 2007). Whereas bottom-up attention is modeled well by the intrinsic features of the input image, top-down attention is affected by the experience. So, what to be attended is affected by learning, on the other hand what is learned is affected by attention. We set the attention level based on the prediction error and how the attention level affects to the learning.

2. Situation dependent predictor with Restricted Boltzmann Machine

2.1 Overview

In order to realize a situation dependent predictor, the prediction of the attended object should be well merged with recognition of the environment.

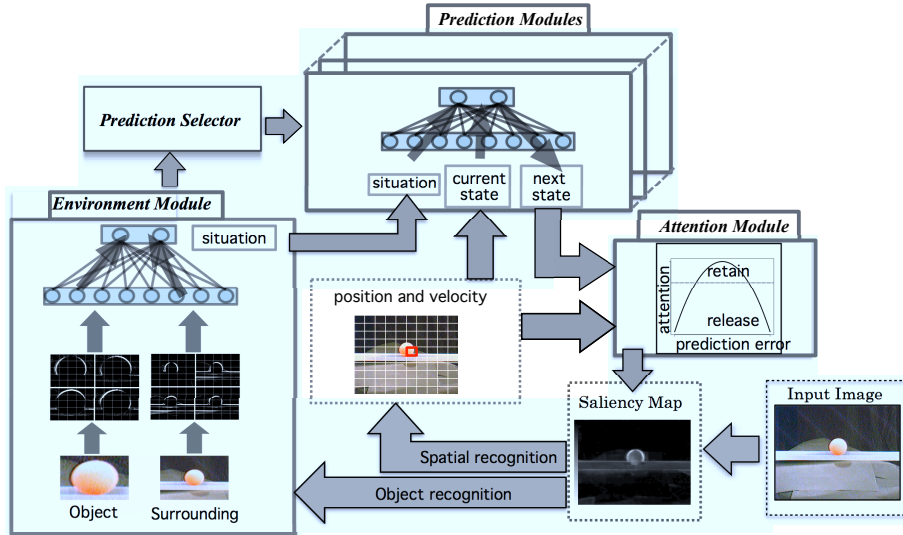


Figure 1: Overview of situation-dependent predictor

This is also an interesting problem as the model of integration of two visual pathways (where pathway and what pathway) in the brain, where appropriate self-organization for information compression and integration should be realized. For that purpose, we apply Restricted Boltzmann Machine (RBM) (Hinton et al., 2006) because this network model possesses good features as a building unit to make a larger system.

Fig. 1 shows a proposed system which consists of 4 modules; attention module, environment recognition module, predictor selector and motion predictor. The attention module determines the attention area in the environment. From the attended area, the geometrical information of an attended object and its surrounding images are extracted and self organized by RBM in the environment recognition module. The self organized information is associated with the information of the movement of the attended object in the prediction module. Based on the association memory feature of RBM, the prediction module can reconstruct the next position of the attended object based on the current position and the current environmental situation. In this section, first, the learning algorithm of the restricted Boltzmann machine is explained. Second, it is explained how RBMs are used in the environment recognition module and prediction module. Then, the attention module is explained

2.2 Restricted Boltzmann Machine

Restricted Boltzmann Machine (Hinton, 2007) (Hinton et al., 2006) is a neural network consisting of two layers, input (visible) layer and hidden layer. There are no connections among units within each

layer. Each unit in the visible layer, v_i , has a symmetrical connection weight, w_{ij} , to each unit in the hidden unit, h_j . Each unit is activated by the following probabilities,

$$\mathbf{P}(h_j = 1) = \frac{1}{1 + \exp(-\sum_i v_i w_{ij} - \beta_{h_j})} \quad (1)$$

$$\mathbf{P}(v_i = 1) = \frac{1}{1 + \exp(-\sum_j h_j w_{ij} - \beta_{v_i})}, \quad (2)$$

where β_{v_i} and β_{h_j} are biases for activation.

The learning of RBM is processed by the calculation process called *reconstruction*. First, the activation level of the hidden layer, h_j , are calculated by the forward calculation based on the input data v_i , the connection weights w_{ij} and biases β_{v_i} with eq. (1). Then the activation level of the visible layer, v_i , is calculated again with the activation level of the hidden layer, h_j with eq. (2). In the following, this re-calculated activation level of the visible layer is called reconstruction data. This calculation process can be proceeded repeatedly. The superscript of the unit v_i and h_j mentions the number of the repeated calculation between layers. When the probabilistic distribution of the input data and the reconstructed data after ∞ repeats of reconstruction are $p(\mathbf{v})$ and $p(\mathbf{v}|\mathbf{w})$, respectively, the purpose of the learning is to adjust the connection weights, w_{ij} , to minimize the difference of the distribution between $p(\mathbf{v})$ and $p(\mathbf{v}|\mathbf{w})$. The distance between two distributions can be measured by the cross entropy error, which is dened by the following equation,

$$L = \langle \log(p(\mathbf{v}|\mathbf{w})) \rangle_{p(\mathbf{v})} \quad (3)$$

$$= \sum_{i=0}^N p(\mathbf{v}_i) \log(p(\mathbf{v}_i|\mathbf{w})). \quad (4)$$

The total energy of the RBM network with the activation level, (\mathbf{v}, \mathbf{h}) in the both layers, can be dened by the following equation,

$$E(\mathbf{v}, \mathbf{h}|\mathbf{w}) = \sum_{i,j} v_i h_j w_{ij}. \quad (5)$$

The probability of the realization of the state (\mathbf{v}, \mathbf{h}) is proportional to the total energy,

$$p(\mathbf{v}, \mathbf{h}|\mathbf{w}) \propto e^{-E(\mathbf{v}, \mathbf{h}|\mathbf{w})}. \quad (6)$$

Thus, when the function of the right side of the equation is described as f like,

$$f(\mathbf{v}, \mathbf{h}|\mathbf{w}) = e^{-E(\mathbf{v}, \mathbf{h}|\mathbf{w})} \quad (7)$$

$$f(\mathbf{v}|\mathbf{w}) = \sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h}|\mathbf{w})}, \quad (8)$$

then the probabilities of the realization of the state (\mathbf{v}, \mathbf{h}) and bfv given the weights bfw can be written with f as

$$p(\mathbf{v}, \mathbf{h}) = \frac{f(\mathbf{v}, \mathbf{h}|\mathbf{w})}{\sum_{\mathbf{v}, \mathbf{h}} f(\mathbf{v}, \mathbf{h}|\mathbf{w})} \quad (9)$$

$$p(\mathbf{v}) = \frac{\sum_{\mathbf{h}} f(\mathbf{v}, \mathbf{h}|\mathbf{w})}{\sum_{\mathbf{v}, \mathbf{h}} f(\mathbf{v}, \mathbf{h}|\mathbf{w})} = \frac{f(\mathbf{v}|\mathbf{w})}{\sum_{\mathbf{v}} f(\mathbf{v}|\mathbf{w})}. \quad (10)$$

Applying the relation $\log p(\mathbf{v}|\mathbf{w}) = \log f(\mathbf{v}|\mathbf{w}) - \log \sum_{\mathbf{v}} f(\mathbf{v}|\mathbf{w})$, the derivation of the cross entropy error, 4, can be transformed as follows,

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{w}} &= \left\langle \frac{\partial}{\partial \mathbf{w}} \log f(\mathbf{v}|\mathbf{w}) \sum_{\mathbf{v}} \frac{f(\mathbf{v}|\mathbf{w})}{Z} \frac{\partial}{\partial \mathbf{w}} \log f(\mathbf{v}|\mathbf{w}) \right\rangle_{p(\mathbf{x})} \\ &= \left\langle \frac{\partial}{\partial \mathbf{w}} \log f(\mathbf{v}|\mathbf{w}) \sum_{\mathbf{v}} p(\mathbf{v}|\mathbf{w}) \frac{\partial}{\partial \mathbf{w}} \log f(\mathbf{v}|\mathbf{w}) \right\rangle_{p(\mathbf{x})} \\ &= \left\langle \frac{\partial}{\partial \mathbf{w}} \log f(\mathbf{v}|\mathbf{w}) \right\rangle_{p(\mathbf{x})} - \left\langle \frac{\partial}{\partial \mathbf{w}} \log f(\mathbf{v}|\mathbf{w}) \right\rangle_{p(\mathbf{v}|\mathbf{w})} \\ &= \left\langle \frac{\partial}{\partial \mathbf{w}} \log f(\mathbf{v}|\mathbf{w}) \right\rangle_{p_0} - \left\langle \frac{\partial}{\partial \mathbf{w}} \log f(\mathbf{v}|\mathbf{w}) \right\rangle_{p_\infty} \end{aligned}$$

where p_0 is the input data (0 th reconstruction data) and p_∞ is the ∞ -th reconstruction data. For actual calculation, instead of p_∞ , 1 st reconstruction data, p_1 , is used for minimization. Then, the derivation can be simplified as

$$\begin{aligned} &\left\langle \frac{\partial}{\partial w_{ij}} \log f(\mathbf{v}|w_{ij}) \right\rangle_{p_0} - \left\langle \frac{\partial}{\partial w_{ij}} \log f(\mathbf{v}|w_{ij}) \right\rangle_{p_1} \\ &= \left\langle \frac{\partial}{\partial w_{ij}} \sum_{i,j} v_i h_j w_{ij} \right\rangle_{p_0} - \left\langle \frac{\partial}{\partial w_{ij}} \sum_{i,j} v_i h_j w_{ij} \right\rangle_{p_1} \quad (11) \\ &= \langle v_i h_j \rangle_{p_0} - \langle v_i h_j \rangle_{p_1} \quad (12) \end{aligned}$$

Thus, the update learning rule for minimizing the cross entropy error can be derived as

$$w_{ij} = \epsilon(v_i^0 \mathbf{P}(h_j^0 = 1) - \mathbf{P}(v_i^1 = 1) \mathbf{P}(h_j^1 = 1)). \quad (13)$$

In the same way, the learning rule for biases can be derived as

$$\beta_{h_j} = \epsilon(\mathbf{P}(h_j^0 = 1) - \mathbf{P}(h_j^1 = 1)) \quad (14)$$

$$\beta_{v_i} = \epsilon(\mathbf{P}(v_i^0 = 1) - \mathbf{P}(v_i^1 = 1)) \quad (15)$$

In the actual learning, the input data are divided into several groups and the parameters are updated group by group to avoid the over learning. Moreover, we added the additional of learning rule to limit the activation rate of each unit. This sparseness constraint seems to be important to describe the input data with more compact patterns of activations in hidden layers (Lee et al., 2008). The convergence of the learning is evaluated by the total error between input data and the reconstruction data,

$$err = v_i^0 - \mathbf{P}(v_i^1 = 1). \quad (16)$$

After learning, the reconstruction process can be used for reconstructing complete data set from the incomplete data. This feature can be used for association of the given multiple data sets. Moreover, when the number of the units in the hidden layer is less than that in the visible layer, the extraction of the important features of the input data can be expected. Hinton stresses that this characteristic of RBM favorable for avoiding local minima in learning of deep layered network. Thus, RBM has both features of supervised and unsupervised self-organization learning properties.

2.3 Environment Module

An object will change the movement depending on the environment where the object is put. The properties of the movement will be affected by the shape of the object. For example, we expect a ball shape will be expected to move easily but not for a square object. And the same object will change its movement depending on the pathway the object is put on. The environment module categories visual information of an attended object and its surroundings.

To extract the geometrical information from the images of an attended object and its surroundings, the results of the gabor filters of these images are input to RBM. The result images of the gabor filters of $\psi = [0^\circ, 45^\circ, 90^\circ, 135^\circ]$ are segmented into 5×5 units. In each unit, the pixel values are summed and normalized to the attended area. The activation level of the j -th unit, I_j , is determined by the normalized summed value a_j and some threshold,

$$I_j = \begin{cases} 1 & (a_j > th) \\ 0 & (a_j \leq th) \end{cases}. \quad (17)$$

The input to the environment module RBM, $\mathbf{v}^{<env>}$, is the combination of the vectors of the gabor filter

results for the object image, $\mathbf{I}^{<obj>}$, and the vectors of the gabor filter results for the surrounding image, $\mathbf{I}^{<around>}$,

$$\mathbf{v}^{<env>} = (\mathbf{I}^{<obj>}, \mathbf{I}^{<around>}). \quad (18)$$

Fig. 2 shows the flow chart of the processing.

After learning, the activation pattern in the hidden layer, $\mathbf{h}^{<env>}$, is expected to describe self-organized information of the input images. Thus, these information is used in the prediction module for prediction of the movement of the attended object.

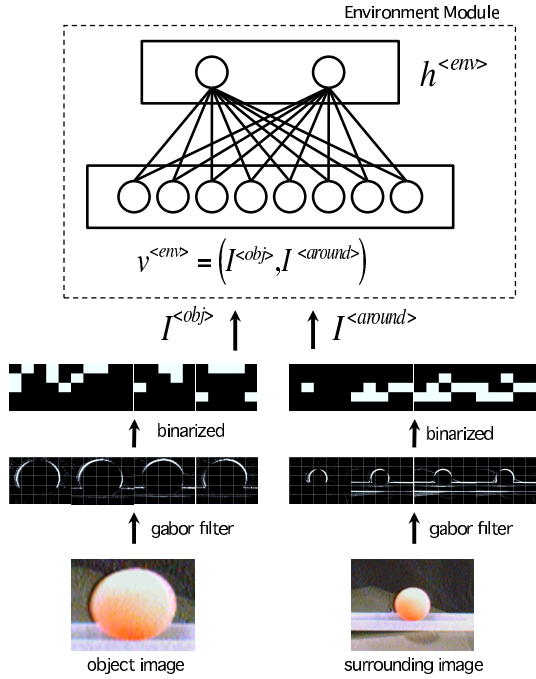


Figure 2: The environment module

2.4 Prediction Module

Fig. 3 shows the schema of the prediction module. The RBM in prediction module associates the current movement information $S(t)$, the previous movement information $S(t-1)$, and the situation information $\mathbf{h}^{<env>}$. The position information consists of the position and the velocity of an attended object,

$$\mathbf{S}(t) = (x_0, x_1, \dots, x_{n-1}, y_0, y_1, \dots, y_{m-1}, dx_0, dx_1, \dots, dx_{2n-1}, dy_0, dy_1, \dots, dy_{2m-1}).$$

When the image size is $W \times H$ and it is divided into $n \times m$, and the coordinates of the attended object are (x, y) , then the position nodes are determined by the following equations,

$$x_i = \begin{cases} 1 & (\frac{x}{W/n} < i < \frac{x}{W/n} + 1) \\ 0 & \text{else} \end{cases} \quad (19)$$

and

$$y_j = \begin{cases} 1 & (\frac{y}{H/m} < j < \frac{y}{H/m} + 1) \\ 0 & \text{else} \end{cases}. \quad (20)$$

When the shift of the attended object between observed steps is (dx, dy) , the velocity nodes are determined by

$$dx_i = \begin{cases} 1 & (\frac{dx}{W/n} + n - 1 < i < \frac{dx}{W/n} + n) \\ 0 & \text{else} \end{cases} \quad (21)$$

and

$$dy_j = \begin{cases} 1 & (\frac{dy}{H/m} + m - 1 < j < \frac{dy}{H/m} + m) \\ 0 & \text{else} \end{cases}. \quad (22)$$

In order to realize the prediction in the various time scales and spatial frames, several RBM with various kinds of time scale and spatial segmentation sizes are prepared. Among them, the appropriate predictor is selected based on the reliability of the predictors. The reliability of i -th predictor, c_i , is calculated based on the hidden layer of the environment recognition module, $\mathbf{h}^{<env>}$, as

$$c_i = \sum_j w_{ij}^s \times h_j^{<env>}. \quad (23)$$

The connection weights, w_{ij}^s , is learned based on the following Hebbian learning,

$$w_{ij}^s = \epsilon (e^{-r_i} \times h_j^{<env>}) \quad (24)$$

where r_i is the prediction error of the movement in i -th prediction module, ϵ is the learning rate. The activation level of each RBM, $a_i^{<RBM>}$, is calculated based on the reliability c_i ,

$$a_i^{<RBM>} = \frac{1}{1 + \exp(-\sum_i c_i)} \quad (25)$$

and the RBM that has the maximum value is selected as the predictor under the current situation.

2.5 Attention Module

We hypothesized that attention consists of three processes; catch, retain and release. First, in the catch process, the attended point is selected based on the saliency (Itti et al., 2003). For that purpose, the saliency map is calculated with regard to various image features such as intensity, color, motion, etc. Once the attended point is decided, the attended object area is evaluated as the set of pixels that have the same color of attended point. Then, the attended object area is segmented and used for the template for pattern matching. The object area is normalized and binarized as the input for the attention module, $\mathbf{I}^{<obj>}$ (Fig. 2). The surrounding image of the attended object whose size is the half of the camera image is normalized and binarized as the input for the

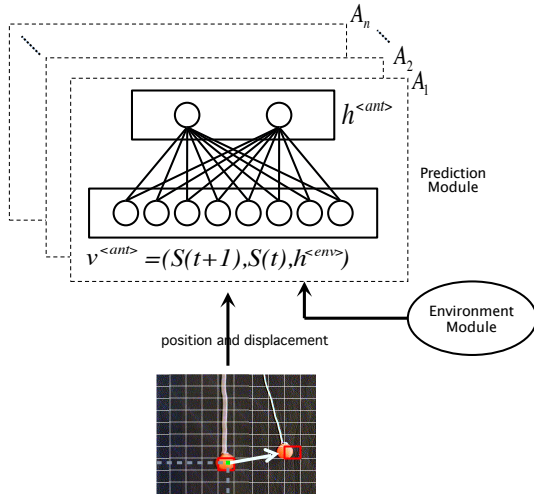


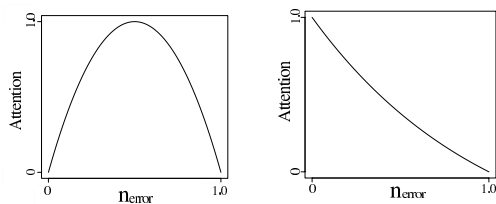
Figure 3: The prediction module

attention module, $I^{<around>}$. The attention point is retained for the learning until the trigger for releasing attended point is given. For effective learning, it is supposed that the attended points whose movement can be predicted completely should be released. The points whose movement are random should also be released earlier because such points may well be noise. On the other hand, the points whose movement can be partly predicted should be retained long for learning more. For that purpose, we compared two kinds of functions that decide the probabilities to release the attention points.

$$attention1 = \frac{e^{n_{error}} e^{1-n_{error}} + e + 1}{2e^{0.5} + e + 1} (26)$$

$$attention2 = \frac{e^{1-n_{error}}}{e} \frac{1}{1} (27)$$

where n_{error} is the rate of the number of the prediction modules that fails to predict. The graphs are shown in Figs. 4. The attention is released when the above attention level becomes less than some threshold.



(a) Attention function1 (b) Attention function2

Figure 4: Attention function

3. Experiments

3.1 Learning Prediction without attention

To validate the prediction faculty, the proposed system is applied to the real robot. Fig. 5 shows the robot *FK* used in the experiment. Although this robot has two IEEE 1394 cameras and 2 degrees of freedom (pan and tilt) to move the camera, only the right camera is used with eye position x_{ed} . The camera image is captured with 33 [frames/sec].

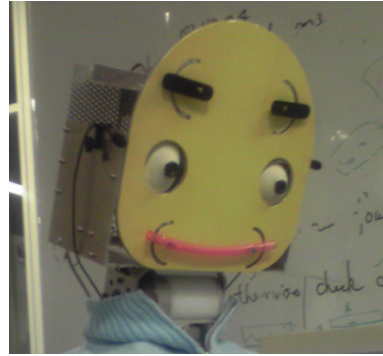


Figure 5: robot *FK*

To validate the prediction faculty in various situations, three kinds of situations are prepared; a ball on the horizontal rail, a ball on the vertical rail and a ball in the pendulum 6. In each situation, 4 tri-

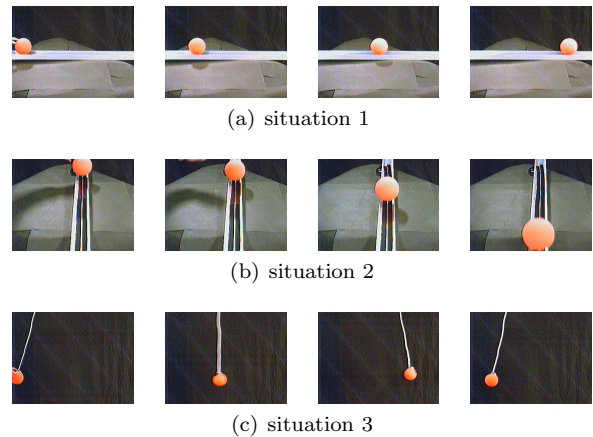


Figure 6: Situations of experiments

als are recorded, each of which has about 90 steps. For the prediction module, 6 RBMs are prepared (2 kinds of segmentations (40×40 , 10×10) and 3 kinds of time steps (5, 10, 20 steps)). The numbers of the visible and hidden units of the environment module are 200 and 50, respectively. The numbers of the visible and hidden units of the prediction modules are 368 and 92 for the segment size 40×40 , 128 and 32 for the segment size 10×10 . The attended area to be

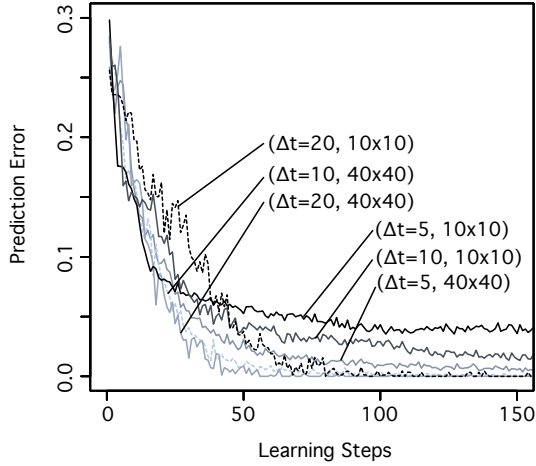


Figure 7: Prediction Error without attention

attended is given as the image template (orange ball) by the designer in this experiment.

Fig. 7 shows the learning error of all RBMs. The learning of each RBM converges within 100 learning steps. The examples of the prediction after learning is mentioned in Fig. 8. These are the predictions of RBMs that have 10×10 segments and 5, 10, 20 prediction time steps (20 step prediction is shown only in every 20 step).

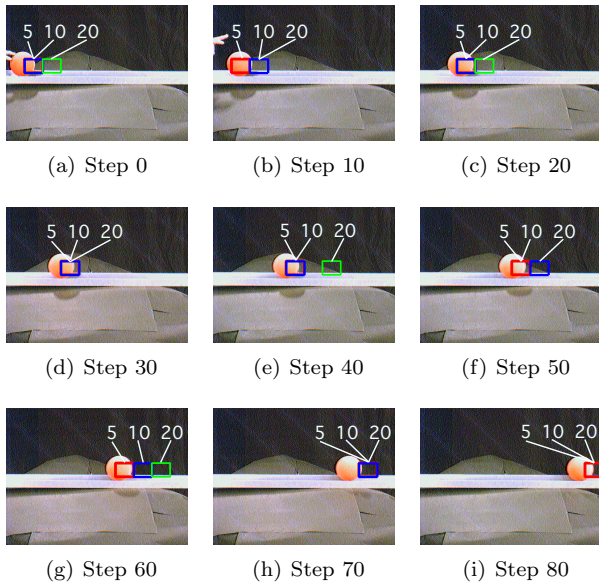


Figure 8: Examples of prediction of the movement after learning

3.1.1 Supplemental Learning

To validate the faculty of the predictor in additional learning, after learning in one situation (Fig. 9(a)), additional data in another situation (Fig. 9(b)) is

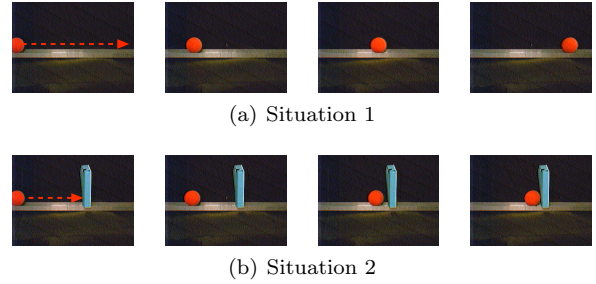


Figure 9: The training data for supplemental learning

given to the network. For each situation, 3 trials (each trial consists of 68 steps) are recorded for training data. The other conditions are the same as the previous subsection. The data of second situation is added to the training data of the situation network after the 250 steps of learning in the first situation. Fig. 10 shows the time courses of the averaged error rate per one node through the learning process (only 2 of 6 predictors are shown). Around the 250-th learning steps, the error rate rises when new data is added to the training data. However, in the following 100 steps, the error rate decreases to around 0.3 indicating that the network successfully represent both the new and old situation. Fig.

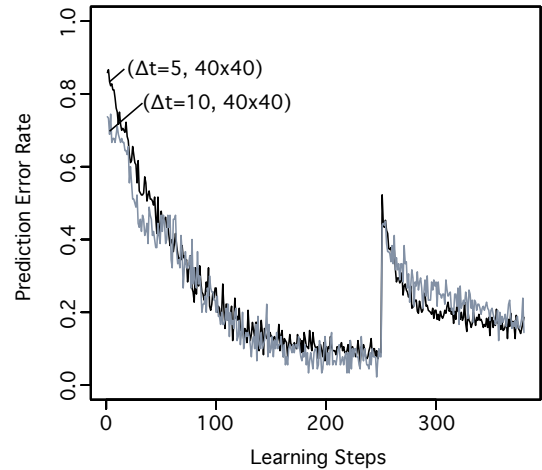


Figure 10: The error rate per one node in the supplemental learning

11 shows the predicted position of the attended object before and after the supplemental learning. (a) Before the supplemental learning, the robot predicts the attended ball will go through the wall because he does not experienced such kind of situation (squares are the predicted positions in the next 5, 10 and 20 steps. The big and small squares are the prediction in 10×10 and 40×40 segments.). (b) After the supplemental learning, the robot can make appropriate predictions depending on the situations. Before the supplemental learning, the robot predicts that

the ball will move through the wall in right direction as before (b). After the supplemental learning, the robot can predict that the ball will stop at the wall (c).

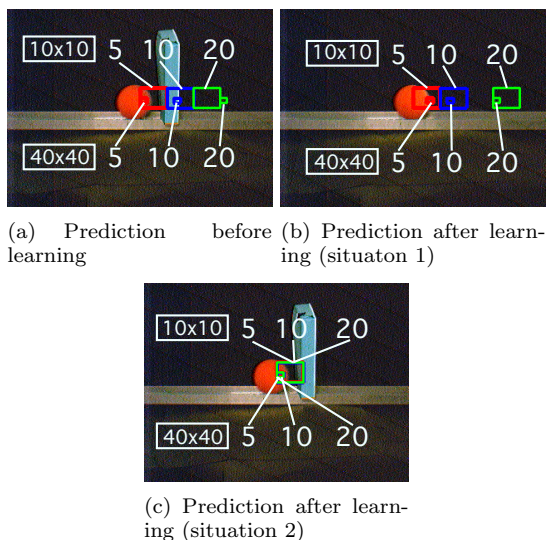


Figure 11: Prediction of the ball position before and after the supplemental learning

3.2 Learning Prediction with Attention

In the experiments of previous subsection, the object to be attended is given by the designer in advance. In order for a robot to learn the physical causality autonomously, it is important to implement an attention control system appropriately. For this purpose, we applied the attention module to the same situations as the experiments explained in the previous subsection. Each situation consists of 3 trials that have 90 steps. For the prediction module, 6 RBMs are prepared (2 kinds of segmentations (40×40 , 10×10) and 3 kinds of time steps (5, 10, 20 steps)) for the prediction modules.

Figs. 12 shows the time course of the error rate in the learning procedures with the attention function 1 (Fig. 12 (above)) and the attention function 2 (Fig. 12 (below)). Whereas the learning is not stable with the attention function 1, the learning with the attention function 2 converges to some stable state. This is because with the attention function 1 the robot easily change its attention to another point (often shiny noise point in the environment other than the object) when the first part of the movement can be learned. Figs. 13 show the timing when the robot changes its attention in the middle of the learning procedure for situation 3 with the function 1 (above) and with the function 2 (below). In these graphs, the gray line indicates the rate of the prediction failure modules, the black line indicates the attention level (calculated by eq. (26) and eq. (27)), and the dashed

line indicates the threshold that the robot changes its attention (the arrows indicate the timing when the robot changes its attention). With the function 1, the robot predicts the first part of the movement successfully and loses its attention easily because the prediction is successfully done. This makes the learning unstable. On the other hand, with the function 2, the robot can keep its attention once the appropriate attention point (the object) is found. And the stable learning data can be obtained.

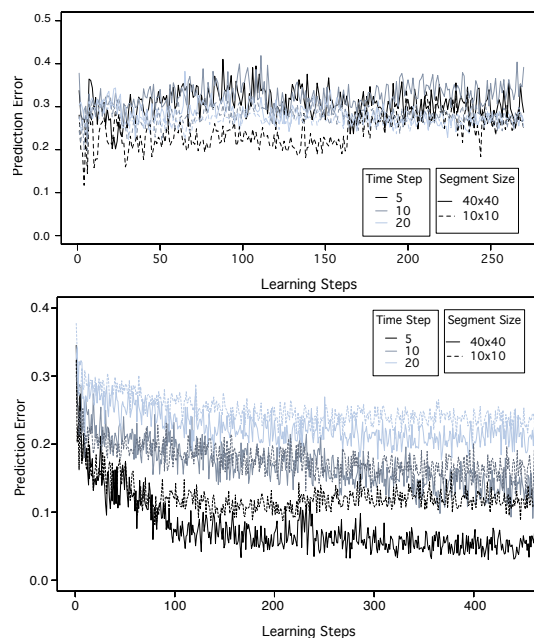


Figure 12: Prediction error based on the attention module with the function 1 (above) and the function 2 (below)

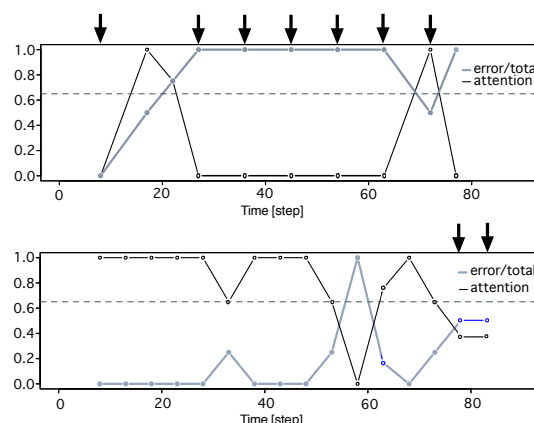


Figure 13: Attention level and attention changes based on the attention module with the function 1 (above) and the function 2 (below)

4. Discussion

In this paper, we proposed a layered associative network that can predict the movements of the observed object depending on the surrounding situation. The higher abstract concept such as object permanence can be acquired through the learning of many concrete phenomena in the real world. The proposed network could be extended to more higher representation of the world. Fig. 14 shows the result of the principle component analysis of the activation patterns in the hidden layer of the prediction module (the image segments are 10×10 and the prediction time step is 5). This graph shows the activation pat-

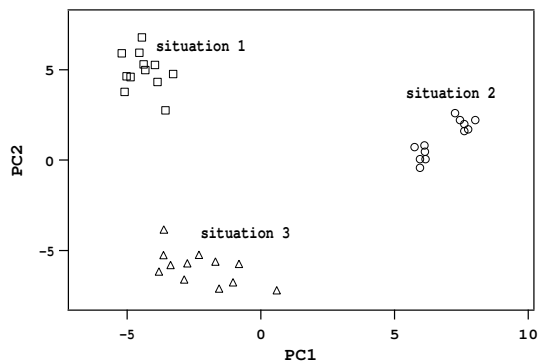


Figure 14: Principal component analysis of the activation patterns of hidden layer in prediction module

terns can be self-organized depending on the situation. So, the information of the activation patterns can be used to discern the states such as The ball is goes to left on the horizontal line. This implies the possibility to construct higher abstract concept based on the self-organization of lower data through the bottom-up approach.

Object permanence is thought to be closely related to memory. To realize an object does not disappear behind an obstacle and will appear again, an agent should recognize that the reappeared object is the same one as the previous one. In the proposed network, if the attended object disappears behind some obstacle, the robot could not retain its attention because the robot will release its attention based on the attention level function 2. However, if the prediction module that enables long term prediction is available, the robot can retain its attention and relate the object behavior during disappearing and reappearing. The key faculty for this learning is how long working memory can record the series of events and how the prediction module will learn from the events in the working memory. In fact, it is reported that the working memory ability of infants is enhanced from 7.5 months (2 secs) to 12 months (12 sec) (Schwartz and Reznick, 1999) (Reznick et al., 2004). We are now conducting the

experiments to relate the prediction ability of a disappeared object and the time length of memory.

References

- Baillargeon, R. and DeVos, J. (1991). Object permanence in young infants: further evidence. *Child development*, 62:1227–1246.
- Baillargeon, R., Spelke, E. S., and Wasserman, S. (1985). Object permanence in ve-month-old infants. *Cognition*, 20:191–208.
- Fitzpatrick, P., Needham, A., Natale, L., and Metta, G. (2008). Shared challenges in object perception for robots and infants. *Journal of Infant and Child Development*, 17(1):7–24.
- Hinton, G. E. (2007). Learning multiple layers of representation. *TRENDS in Cognitive Sciences*, 11(10):428–434.
- Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554.
- Itti, L., Dhavale, N., and Pighin, F. (2003). Realistic avatar eye and head animation using a neurobiological model of visual attention. *Proceedings of SPIE*.
- Knudsen, E. I. (2007). Fundamental components of attention. *Annual Review of Neuroscience*, 30:57–78.
- Lee, H., Ekanadham, C., and Ng, A. Y. (2008). Sparse deep belief net model for visual area v2. In *Proceedings of the Neural Information Processing Systems (NIPS) 20*.
- Lovett, A. and Scasselatti, B. (2004). Using a robot to reexamine looking time experiments. In *Proceedings of the 4th International Conference on Development and Learning (ICDL)*.
- Piaget, J. (1954). *The construction of reality in the child*. basic books.
- Reznick, J. S., Morrow, J. D., Goldman, B. D., and Snyder, J. (2004). The onset of working memory in infants. *Infancy*, 6(1).
- Schlesinger, M. (2003). A lesson from robotics: Modeling infants as autonomous agents. *Adaptive Behavior*, 11(2).
- Schwartz, B. B. and Reznick, J. S. (1999). Measuring infant spatial working memory using a modified delayed-response procedure. *Memory*, 7:1–17.

Reward-free Learning using Sparsely-connected Hidden Markov Models and Local Controllers

Kohtaro Sabe, Kenta Kawamoto, Hirotaka Suzuki, Katsuki Minamino, and Kenichi Hidai
Corporate R&D System Technologies Laboratories, Sony Corporation
Gotenyama TEC, 5-1-12, Kitashinagawa Shinagawa-ku, Tokyo, 141-0001 Japan
Kohtaro.Sabe@jp.sony.com

Abstract

A novel framework for behavior learning for an autonomous agent without any reward functions is presented. It is focused on how to build and control internal representations of various environments from incomplete data without any a priori knowledge. Hidden Markov Model (HMM) has the potential to represent hidden structures of environments from partial observations, yet learning of HMM parameters without any assumptions remains very difficult. Without loss of generalization, we bring constraints to the parameters of HMM to reduce learning difficulty. The only motive given to the agent is to get the control of learned internal states so that local controllers assigned to HMM nodes are learned to achieve this goal. The framework is tested on two different types of environments: a mobile robot and a robot manipulator. In the mobile robot environment, only the range finder sensors are given to the agent while the robot randomly moves through the maze-like room. As a result, a place representation is captured in the hidden states of HMM. We show that by learning to control these internal states to arbitrary states, the robot can exhibit the skill of navigation. The same model is applied to a single link robot manipulator environment and the robot acquires the control of its dynamical properties.

1. Introduction

The objective of our work is to build an autonomous agent with open-ended learning capability, which is driven by the ultimate goal to acquire the abilities to predict and control everything that the agent has experienced (Sabe et al., 2005) (Fujita, 2009). For such a system, the ability to cope with multiple tasks without a priori knowledge or even without objectives about each task is necessary. We think that the requirements of such open-ended learning agents can be defined as follows:

1. The agent self-organizes internal representations of the unknown environments from partial observation sequence.
2. The agent recognizes past and current states as well as predicts future states.

3. The agent controls the self-organized states to arbitrary target states if possible.
4. The agent has intrinsic motivations to improve above three abilities.
5. The agent has extrinsic motivations to sustain itself within the environments.

Reinforcement learning is a machine learning method that acquires the optimal behaviors based on actual experiences and rewards in unknown environments. However, this framework itself does not provide the solution to the first requirement. For the second and the third requirement, prediction and control are tightly coupled with task specific objective functions so that learned knowledge about the environments cannot be reused efficiently.

In the field of robotics, the framework of Partially Observed Markov Decision Process (POMDP) has been adopted in robot navigation tasks called Simultaneous Localization and Mapping (SLAM) (Leonard and Durrant-Whyte, 1991) in which robot develops a map of unknown environments and at the same time localizes and controls itself anywhere on the learned map. Most of the successful systems adopt the task dependent representation of the environment such as Occupancy Grids (Elfes, 1989). Motion models and observation models are usually known or parameterized from the configuration of the robots.

SLAM is derived from EM algorithm implemented for Hidden Markov Model (HMM) known as Baum-Welch algorithm (Baum, 1970). A HMM has the potential to represent hidden structures of environments from observations, yet learning of HMM parameters without any assumptions remains very difficult. Successful applications such as a speech recognition task usually define the structure and the observation model as the left-to-right HMM with a Gaussian Mixture Model.

In theory, fully connected HMM (ergodic HMM) can be used to estimate the structure of a learning subject because a set of parameters expressing the true structure of the subject is the global minimum in HMM learning. However when learning a fully connected large scale HMM, we can not expect it to converge to the global minimum.

We put an assumption that almost all of the real world phenomenon can be expressed with a sparse structure, e.g. Small World Network. From the analogy of neural

connections in the cortex, the structure is at most 3 dimensionally constrained. Since the cortex is a sheet of neurons folded in 3D space, a 2D constrained structure may be sufficient.

Our scenario of the developmental learning is as follows.

1. The agent acts with randomly generated actions or innate actions (motor babbling). Sparsely-connected HMM is used to learn the sensor readings for certain periods of time to build the HMM structure of the environment.
2. The agent learns the local controllers assigned to each HMM node using the observations and actions whenever node transitions occurred during motor babbling.
3. The agent select one of its learned HMM nodes as a target node. It makes a plan to reach the target node from currently recognized node. It activates local controllers along the planned path to emit actions.
4. The agent repeats step 3 to refine HMM and controllers while the performance of controlling to desired nodes improves.

We will show how this general framework of development works by two different classes of simple simulated environments: One is a mobile robot and robot and the other is a robot manipulator.

In section 2, the overall method is described in which learning of sparsely-connected HMM and local controllers, and planning and execution procedures are explained. In section 3, the experiments and results using a mobile robot are described. In section 4, the experiments and results using a pendulum are described. The conclusions and the future works are discussed in the last section.

2 Methods

2.1 Framework of Reward-free Learning

The system diagram of proposed method is shown on Fig.1. It consists of the predictor module, the controller module, the planner module, and the innate controller.

In the learning phase, observed sensor signals are fed to the predictor module where Hidden Markov Model (HMM) is used to learn the internal models of observed sequences. After learning the model, the same sequences are used to recognize the corresponding state at each time step. The results of recognitions are fed to the controller module. Whenever state transitions occur, relations from sensor observations to actions are learned in local controllers assigned to each node of the HMM.

In the execution phase, the target state in the HMM is selected from the learned states and given to the execution module. A plan of HMM state transitions from current state to the target state is generated by

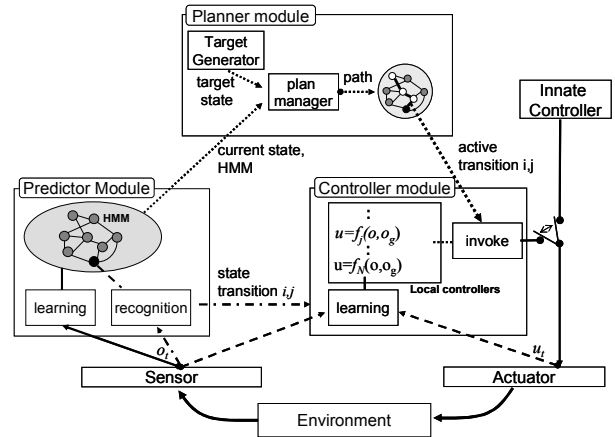


Figure 1: Framework of Reward-free Learning means of graph search. This plan is executed by activating the corresponding local controllers along the path of the generated plan.

2.2 Sparsely-connected HMM

In the proposed method, we take the strategy to learn the internal models without its own actions and to learn the associations of actions and internal states later. Because in open-ended learning, everything observed and predicted may not be controllable and the causalities of actions to sensors should be discovered by the agent. In this paper, we assume that given actions have direct influences on the observations, but the learning of the predictors and controllers are separated for future extensions.

Hidden Markov Model (HMM) is a useful tool to model the observation sequences which are parameterized with initial state probabilities π_i , transition probability matrix a_{ij} , and observation likelihood at each state, modeled by mean μ_i and variance σ_i^2 of a Gaussian distribution.(Fig.2)

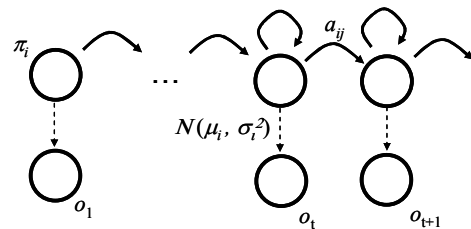


Figure 2 : Graphical model of HMM

To estimate the HMM parameters $\{\pi_i, a_{ij}, \mu_i, \sigma_i^2\}$ from given observation sequence o_t , Baum-Welch algorithm (Baum 1970), (Rabiner and Juang, 1993) is used. In the E step, the Forward Backward algorithm is used to estimate state probabilities at each time step and likelihood of given sequences. In the M step, HMM parameters are updated to maximize this likelihood. These steps are iteratively applied until parameters converge.

Standard Baum-Welch algorithm

Initialize HMM parameters $(a_{ij}, \pi_i, \mu_j, \sigma_j^2)$

for $k=1:\text{max_iteration}$

E-step : Calculate $\alpha_i(t), \beta_i(t)$

$$b_j(o_t) = N(x, \mu_j, \sigma_j^2) \quad (1)$$

$$\alpha_i(i) = \pi_i \quad (2)$$

$$\alpha_{t+1}(j) = \left[\sum_i \alpha_t(i) a_{ij} \right] b_j(o_{t+1}) \quad (3)$$

$$\beta_T(j) = 1 \quad (4)$$

$$\beta_t(i) = \sum_j a_{ij} b_j(o_{t+1}) \beta_{t+1}(j) \quad (5)$$

Calculate log likelihood L_k of observation sequence $o_{1:T}$

$$L(o_{1:T}) = \log \left(\sum_i \alpha_T(i) \right) \quad (6)$$

Abort if $L_k - L_{k-1} < \varepsilon$

M-step : Estimate HMM parameters

$$\bar{\pi}_i = \alpha_i(i) \beta_i(i) \quad (7)$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{t=1}^{T-1} \alpha_t(i) \beta_t(i)} \quad (8)$$

$$\bar{\mu}_j = \frac{\sum_{t=1}^T \alpha_t(j) \beta_t(j) o_t}{\sum_{t=1}^T \alpha_t(j) \beta_t(j)} \quad (9)$$

$$\bar{\sigma}_j^2 = \frac{\sum_{t=1}^T \alpha_t(j) \beta_t(j) o_t^2}{\sum_{t=1}^T \alpha_t(j) \beta_t(j)} - \bar{\mu}_j^2 \quad (10)$$

Update HMM parameters with estimated parameters.

In the theoretical aspect, Baum-Welch algorithm provides means to estimate a structure (=transition matrix a_{ij}) of an unknown environment. (Chrisman, 1992) showed the experiment in a small discrete POMDP setting, but it is usually very difficult to train such a HMM without any a priori knowledge.

In most of the practical applications handling time-series data such as speech recognition and gesture recognition, left-to-right HMMs are used. Reducing the complexity of HMM structure makes training easier, but it becomes too task-specific. It is also a problem because we need to segment the data to train left-to-right HMMs. For autonomous agents, training data are neither labeled nor segmented. (Clarkson and Pentland, 1999) used segmental K-means algorithm to segment automatically the data of lifetime log, but this approach only copies the segmented data and does not capture the true structure of the entire environments.

In this paper, we took the approach to put all the data into one big HMM to self-organize the structure of the environment.

In order to overcome the difficulties of training large scale HMM, several concepts are introduced. One is to restrict the topological configuration but not as tight as

left-to-right. It may be natural to assume that networks with a large number of local connections in low dimensionality can describe most of the physical systems. Therefore, we allocate the nodes of HMM in 2D square grid or 3D cubic lattice and connect each other with nodes closer than a threshold, θ_d . The distance between the adjacent nodes of the grid is defined as 1. For examples, if θ_d is set to 1, 4 adjacent nodes are connected and if θ_d is set to $\sqrt{2}$, 8 adjacent nodes are connected.

The second is to reduce the redundancy in parameters by simplifying the observation model to a single Gaussian model rather than using a mixture model.

The third is an annealing effect. Just like standard clustering problems, initial observation models have stronger influences on the result of training than transition probabilities. We set the same initial values to all μ_j and σ_j^2 so that training of transition probabilities proceeds before observation models are fixed. We also allowed μ_j and σ_j^2 in equation 9 and 10 to change gradually by introducing trace rule (Eq.11) between the iteration steps.

$$\bar{\mu}_j^{(k)} = (1 - \tau) \bar{\mu}_j^{(k-1)} + \tau \bar{\mu}_j^{(k-1)} \quad (11)$$

where τ in k^{th} step is $\tau = k / \text{max_iteration}$.

The last concept is time hierarchy. The long sequence of observations is sub-sampled. Coarse transitions are trained at the earlier stage and the finer transitions are trained later.

In addition to the concepts mentioned above, the scaling technique introduced in (Rabiner and Juang, 1992) to avoid underflow when calculating likelihood and state probabilities of very long sequences becomes also important.

Learning procedure

1. Initialize HMM parameters with following grid constraints and observation likelihood.

$$a_{ij} = \begin{cases} \text{distance}(i, j) \leq \theta_d : 1/n' \\ \text{distance}(i, j) > \theta_d : 0 \end{cases}$$

n' is the normalize factor, which is number of nonzero connections in j^{th} column.

θ_d is a threshold distance in the node topology

$$\mu_i = 0.5, \sigma_i = 0.05$$

$$\pi_i = 1/N$$

2. Subsample every m th data of original observation sequence. $m=1/r$, where r is sub-sampling rate.
3. Train HMM with Baum-Welch algorithm.
4. If sub-sampling rate is 1, finish learning.
5. Reduce the sub-sampling rate. Add small values, ε to all transitions probabilities and normalize the probabilities in each column. Go to step 2 without changing other parameters.

2.3 Recognizing HMM states

After training HMM, the current state can be estimated using Viterbi algorithm (Forney, 1973). It provides the optimal estimate of hidden state sequence for a given observation sequence.

Recognition procedure

1. Set uniform probabilities at $t = 1$
 $\delta_1(i) = 1/N$
2. for $t=1:T-1$
 $\delta_{t+1}(j) = \max(b_i(o_{t+1})a_{ij} + \delta_t(i))$ for i
 $\psi_{t+1}(j) = \operatorname{argmax}(b_i(o_{t+1}) a_{ij} + \delta_t(i))$ for i
 Rescale at each time step
 $\delta_{t+1}(j) = \delta_{t+1}(j) / \sum_k \delta_{t+1}(k)$
3. Backtrack δ
 $s(T) = \operatorname{argmax}(\delta_T(i))$ for i
 for $t = T-1:-1:1$
 $s(t) = \psi_{t+1}(s(t+1))$
4. Output $s(t)$ as winner node and $\delta_t(i)$ as probabilities of state i at time t

2.4 Learning local controllers

Once the state space is known, various methods of reinforcement learning can be applied by assigning a reward on a target state. However, we do not want to acquire a single task actor.

In order to achieve multiple targets, a target state should also be a part of the state representation, which would require $N \times N$ states combinations and is not feasible. Since the state space is divided with the assumption that their transitions are sparse and local, local controllers that handle only the transitions local to a state may be easy to learn. Figure 4 shows the conceptual schema of transitions on continuous state space. All the observation data in neighboring nodes i which later make a transition to node j are used for learning local controller in node j . Transitions are expressed with the triplets of data (o_t, o_{t+1}, u_t) .

For each time step, HMM recognizes current winning node i and buffers a triplet. Once the winning node turns into node j , all the buffered triplets are used to train the local controller in node j , which means that all of the data are used for training any one of the controllers. The local controllers are trained in the following functional form to output directly the action u_t from the observation.

$$u_t = f_j(o_t, o_{t+1}) \quad (12)$$

In the execution phase at time t , instead of passing o_t and o_{t+1} , we give o_t and μ_j (mean observation of node j) as inputs to the controller. As a result, a local attractor which flows into the centroid of node j is formed as figure 5. The agent can control the transition from node i to node j by calling this attractor repeatedly until

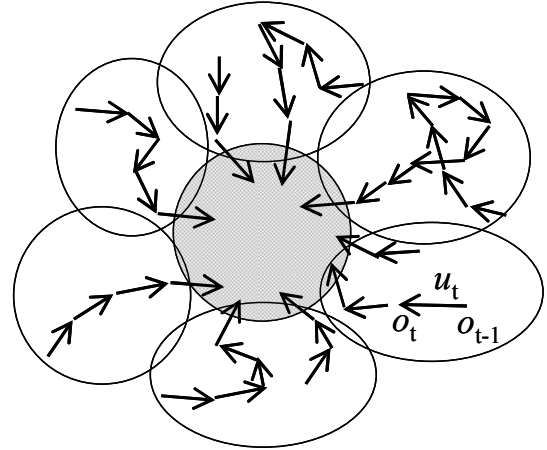


Figure 4: Transitions on state space around node j . Each arrow shows a transition from time t to $t+1$ with action u_t . Circle means the coverage of states state space by a certain node i . The gray circle is a destination node j .

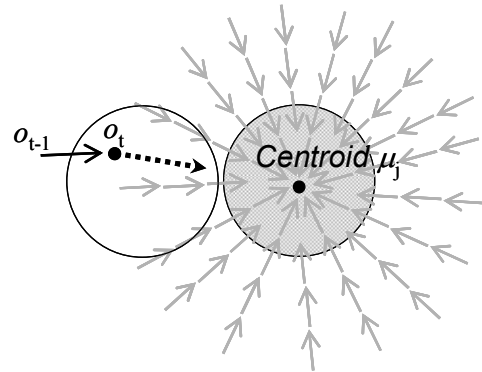


Figure 5: Local attractor formed around node j

entering to node j .

For the function estimator in equation 12, we used Accurate Online Support Regression (Ma et al., 2003) given by following equation,

$$y = \sum_i \alpha_i K(x_i, x) + b \quad (13)$$

where the following RBF kernel is used.

$$K(x_i, x) = \exp\left(-\frac{|x_i - x|^2}{2\sigma^2}\right) \quad (14)$$

2.5 Planning and execution for achieving target states

For autonomous agents, the objectives are given from the motivational system derived from intrinsic or extrinsic motivations. It somehow associates different objectives to different internal states.

Once the desired states are provided, a simple path planning is carried out using transition matrix a_{ij} to generate path from the current state to the desired state. We use Viterbi algorithm again with some modifications for use in planning. Since we do not have observation sequence for future, calculations of the

observation likelihood $b_j(o_t)$ are omitted (considered them as 1.0). The values of probability in a_{ij} are also ignored and set to 1.0 for all the nonzero transitions because the innate controller might have the bias to the action and we only care whether the transitions are enabled or not. The planning procedure is described below.

Planning procedure

1. Index of the target node is d .
2. Set the probability $\delta_1(i)$ of current state to 1.0 and others to 0.0.
3. Set transition probabilities a_{ij} above ε ($=0.01$) to 1.0.
4. Apply probability propagation in step 2 of Recognition Procedure in section 2.3 until $\delta_i(d)$ becomes nonzero or reaching to maximum steps of propagations.
5. If $\delta_i(d)$ is nonzero, apply backtrack in step 3 of Recognition Procedure to fill the path of states. If $\delta_i(d)$ is zero, fill null path.

To execute this plan, simply local controllers on the path are invoked according to the current state recognition. The detail of the execution procedure is described as follows.

Execution procedure

1. Calculate current probabilities of all the state by Viterbi algorithm using past τ steps of observations.
2. Find maximum probability of the state along the path between a previous state and a target state and set it as a current state.

$$s(t) = \underset{s \in \text{path}(s(t-1) \rightarrow \text{target})}{\text{arg max}}(P(s))$$
3. If probability of $s(t)$ is below threshold, abort plan.
4. Use a controller f_j of the next state j along the path to calculate action u_t using observation o_t and state observation mean μ_j .

$$u_t = f_j(o_t, \mu_j)$$
5. Go to step 1 until reaching a target state.

If the execution is aborted, then it re-plans the path and executes again until it reaches maximum limit of

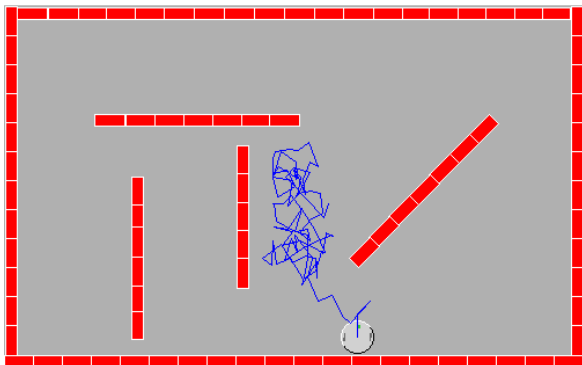


Figure 5: Khepera simulator

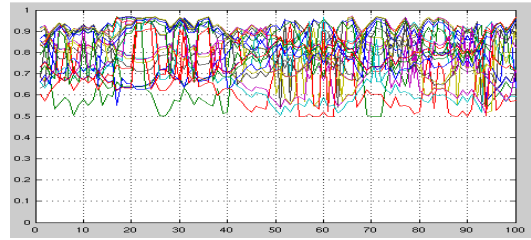


Figure 6 : observation sequence of distance sensor repetitions. In this case, the agent aborts the target and gets a new target.

3. Mobile Robot Environment

3.1 Experimental settings

For the first experiment, we used a mobile robot environment using Khepera simulator (Michel, 1996) for a two-wheel robot in maze-like room.

There are several modifications to the original simulator. The distance sensors measure the relative distance from the robot to the nearest wall. The number of distance sensors is extended to 24 from original of 8. Sensors surround a robot 15 degrees apart from each other. The range of them is extended to 500 from the original of 60. The action to the robot becomes $\delta X, \delta Y$ instead of a wheel command, δL and δR . $\delta X_{\max}=10, \delta Y_{\max}=10$ for action commands. The size of the room is about 600×1000 . The light sensors are not used in this task.

The agent is connected to this simulated environment through 24 sensor inputs and 2 action outputs whose ranges are normalized to fit between 0.0 and 1.0, and no a priori knowledge about this environment is given.

In the standard robot navigation task, state representations such as robot positions (X, Y) and an environmental map (Occupancy grids), an observation model (how walls are observed given a map), and a motion model (how positions changes by action

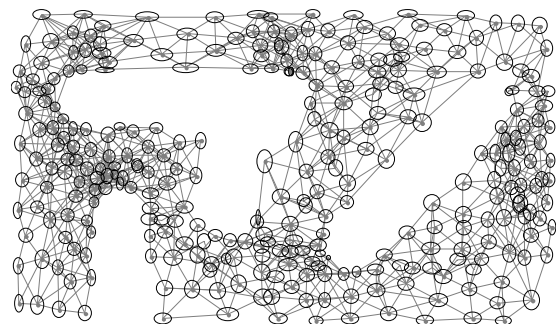


Figure 7: HMM nodes and their transitions
A circles represents a HMM node. They are plotted on average of ground truth robot positions while the node is active. Lines connecting the nodes are indicating the transitions with nonzero probabilities.

commands) are given. The objective of the task is to build a map and control positions.

In our framework, the agent knows nothing mentioned above, instead, it has general modules namely the predictor, the controller, and the planner. It has to find out everything through the responses of the sensors from its actions. We are going to show that even with these challenging settings, a navigation skill emerges as a consequence of learning the environmental structure.

The random innate controller is used to generate actions in training (babbling) phase, in which action δX , δY are generated from uniform distribution in the range of $[-\delta X_{\max}, \delta X_{\max}]$ and $[-\delta Y_{\max}, \delta Y_{\max}]$ at every time step. An example of the trajectory from such actions is shown in the figure 5. An example of 24 dimensional distance sensor measurements is shown in the figure 6.

3.2 Result of learning HMM

HMM is trained with the observation sequence generated by the innate controller with the following parameters.

Parameters

- Length of the data: 10,000 steps
- Number of nodes (2D Grid): $20 \times 20 = 400$
- Number of neighbor connections: 12 ($\theta_i = 2.0$)
- Max. EM iterations: 200
- Subsample rate: 1/10, 1/5, 1/2, 1
- Min. observation variance σ^2 : 0.01^2

As a result, a topological configuration in Fig. 7 is learned. When compared with figure 5, it is clear that each state in HMM represents a location in the simulated environment. The global structure of the room was captured very well even from the partially observed distance data, which are similar in different places of the environments.

The nodes, which have never been used throughout

the learned sequence, are removed. Out of 400 nodes initially prepared 309 nodes are remained. Out of 10,000 transitions initially assigned to nonzero, 2243 transitions remained nonzero.

3.3 Result of learning local controllers

After learning the HMM, same sequence is used to recognize state transitions. Using this state transitions, every triplets (o_b, o_{t+1}, u_t) in the sequence can be assigned to one of the local controllers for learning.

Figure 8 shows the result of attractors formed at two different nodes. The zoomed figure corresponds to particular positions on the simulated environment and center of the figure is the centroid of the node. It seems that in both cases all the movement of the robot is heading towards the centroid of the node.

3.4 Testing the acquired behaviors

Using learned HMM and local controllers, the tests are carried out to examine what kinds of behaviors are acquired in the agent. The agent is randomly given a target state which is one of the learned states. The planning and execution phase described in the previous section is carried out for that given target state.

Figure 10 shows the snapshot of execution. The target node pointed by the arrow was given to the agent. The node near the upper right corner which was the other end of the connected line, was the state that agent recognized when it made the plan to the target. The connected line is the planned path. The agent invoked the controller along path and reached the half way of the path. The simulated robot moved from right to left corresponding to the change of the node recognition.

To show the performance quantitatively, we measured the success rate of the tasks. A task is given by selecting one of the states randomly. A single trial finishes either the agent gives up or reach a target. The next task starts from the last state in the previous trial.

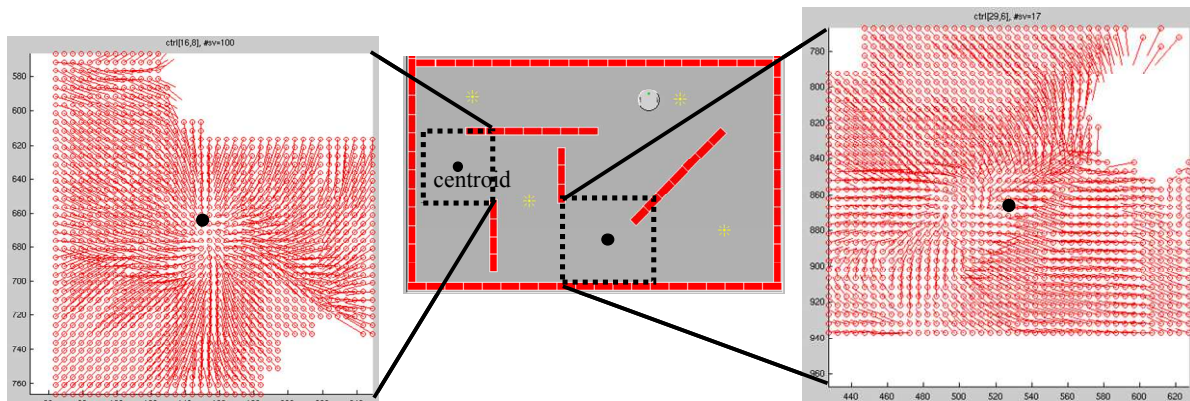


Figure 8: Result of attractors formed around HMM nodes
Circles in zoomed figures are grid positions on simulator. Each arrow represents robot motion caused at the circle using the action from local controllers.

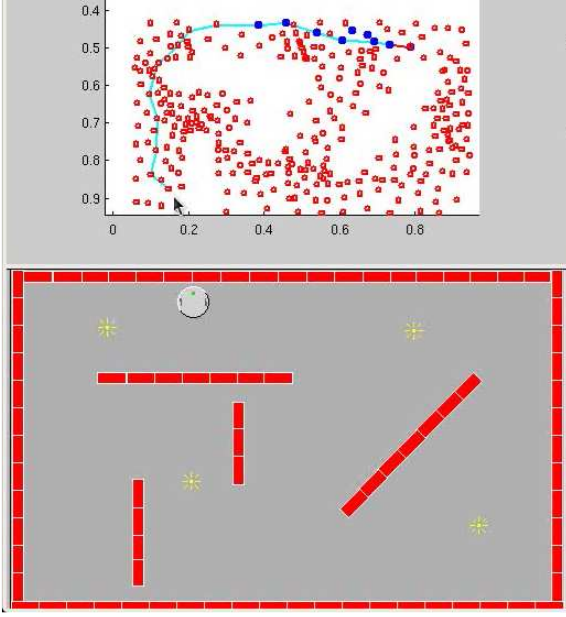


Figure 9 : snapshot of execution

Lines connecting the nodes are the path. Filled circles are the nodes activated recently. Lower part of the figure is the simulated robot moving right to left along the corridor

Out of 118 trials, 108 times were successful in reaching the target, which means the agent thought it had reached the target. The success rate was 95.6%.

We showed that after learning the structure of the environment and learning to control arbitrary transitions, the optimal combinations of these local controls to achieve desired states become some meaningful and appropriate behaviors even no task specific objectives are given in the learning framework.

4. Robot Manipulator Environment

4.1 Experimental Settings

We have also tested the agent on a different environment, which is a simple robot manipulator shown in Figure 10(a). We have used the same physical parameters described in (Doya et al., 2002). It has 1DOF joint with some friction. The torque of the joint is set weaker than gravity so that it needs to swing back and forth to be in a desired position. The task of

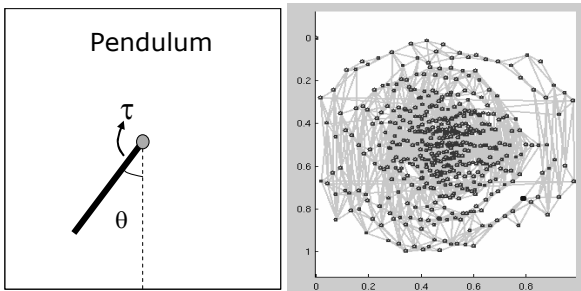


Figure 10: (a) Pendulum (b) Learned HMM
In (b), states in HMM is plotted on ground truth of (θ, ω) space

swinging up to the top position is often used as the standard reinforcement-learning problem. Since the state of this dynamical system can be fully described with angle θ and angular velocity ω , a torque controller that maximize the reward which is given at the top position can be learned in the state space (θ, ω) .

In our environmental settings, no reward is given and angular velocity ω is hidden. Even with these difficult settings we are going to show that not only the swing-up but also the ability to control to arbitrary states can be learned with our proposed methods.

For the innate controller, we design again a random torque generator that switches torque from one of three values $\{-\tau_{\max}, 0, \tau_{\max}\}$ every 30 time steps.

HMM is trained using the one dimensional observation sequence of angle θ . Almost same learning parameters are used for this environment.

Parameters

Length of the data: 10,000 steps

Number of nodes (2D Grid): $22 \times 22 = 484$

Number of neighbor connections: 12 ($\theta_t = 2.0$)

Max. EM iterations: 200

Subsample rate: 1

Min. observation variance $\sigma^2 : 0.01^2$

4.2 Results

As a result, a graphical representation in figure 10(b) is learned. Same as the previous experiment, each node is plotted on the ground truth of the system state (θ, ω) . It captures very well of the hidden structure of pendulum dynamics. Local controllers are also learned for this task. The execution tests are performed with random selections of the targets out of the learned states.

Figure 11 shows the snapshot of the execution. The left figure is the plot of the node in (θ, ω) state space and the center is $\theta=0, \omega=0$. The created path goes around the center, which means that it swings back and forth to go down to the lowest position with highest speed. Green circles are ground truth trajectory of the pendulum and blue are the estimated nodes. Some recognition results come off the path but the actual pendulum is controlled well on the planned path.

5. Conclusion

We have proposed a novel framework of reward-free learning through which the agent can acquire appropriate behaviors or the skill suitable for the environment without any task specific objectives. Two experimental results are shown in which the agent acquired navigation and manipulation skills respectively.

We have also shown that learning the entire

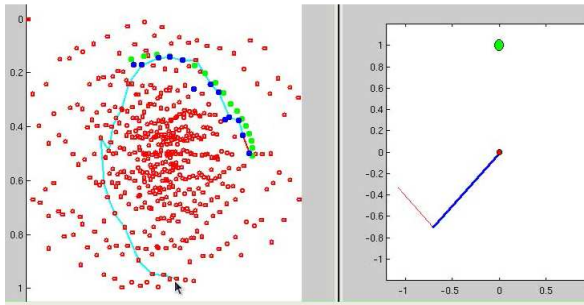


Figure 11 : snapshot of execution

observation sequence with single large sparsely-connected HMM can be possible and it enables the self-organization of hidden structures of environments.

The roles of skill formation and intrinsic motivation are important factors in open-ended learning. In previous works (Sutton, et al., 1999), (Barto et al., 2004), (Oudeyer, P.-Y et al., 2007) dealing with these functions, the main focuses are put on implementing individual mechanisms in which states or state space are predefined and goals and sub-goals are predetermined. Our work provides the outer framework integrating these functions to achieve open-ended learning. It is important for agents to create and to solve problems other than given ones.

There seems to be many extensions needed to cope with larger scale problems, especially hierarchy and recursiveness are important factors to implement. There are hierarchical approaches to improve the efficiency of training and data representation such as Hierarchical HMM (Fine et al., 1998), Layered HMM (Oliver et al., 2004). By applying the knowledge from such systems, the extension may not be so difficult.

References

- Barto, A. G., Singh, S., and Chentanez, N. (2004). Intrinsically motivated learning of hierarchical collections of skills. *Proceedings of the Third International Conference on Developmental Learning*, pp. 112-119.
- Baum, L.E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Statist.*, vol. 41, no. 1, pp. 164--171.
- Chrisman, L. (1992). Reinforcement learning with perceptual aliasing: The perceptual distinctions approach. In *Proceedings of the Tenth International Conference on Artificial Intelligence*, pages 183-188. AAAI Press, San Jose, California.
- Clarkson, B. and Pentland, A. (1999). Unsupervised Clustering of Ambulatory Audio and Video. *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing, IEEE CS Press*, vol. 6, pp. 3037-3040.
- Csikszentmihalyi, M. (1990). Flow: The psychology of optimal experience. *New York: Harper and Row*.
- Doya, K., Samejima, K., Katagiri, K., and Kawato, M. (2002). Multiple model-based reinforcement learning. *Neural Comput.* 14, 6, 1347-1369.
- Elfes, A. (1989). Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation. *PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University*.
- Fine, S., Singer, Y., and Tishby, N. (1998). The Hierarchical Hidden Markov Model: Analysis and Applications. *Machine Learning*, vol. 32, p. 41-62.
- Forney, G. D. (1973). The Viterbi algorithm. *Proceedings of the IEEE* 61(3), 268–278, March 1973.
- Fujita, M. (2009). Intelligence Dynamics: a concept and preliminary experiments for open-ended learning agents. *Autonomous Agents and Multi-Agent Systems*.
- Leonard, J. J., Durrant-Whyte, H. F. (1991). Simultaneous mapbuilding and localization for an autonomous mobile robot. *Proceedings of IEEE Int. Workshop on Intelligent Robots and Systems: 1442-1447*.
- Ma, J., Theiler, J., and Perkins, S. (2003). Accurate on-line support vector regression, *Neural Computation Vol.15, Issue 11 pp.2683-2703*.
- Michel, O (1996). Khepera Simulator Package version 2.0. *Freeware Khepera simulator. It can be downloaded from* <http://diwww.epfl.ch/lami/team/michel/khep-sim/>
- Oliver, N., Garg, A., and Horvitz, E. (2004). Layered Representations for Learning and Inferring Office Activity from Multiple Sensory Channels. *Computer Vision and Image Understanding (CVIU)*, 96, pp. 163-180.
- Oudeyer, P.-Y., Kaplan, F., Hafner, V.V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation, Special Issue on Autonomous Mental Development*, 11 (1), pp. 265-286.
- Sutton, R. S., Precup, D., and Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement Learning, *Artificial Intelligence* 112, pp.181–211.
- Rabiner, L. and Juang, B.-H. (1993). Fundamentals of Speech Recognition. *Prentice Hall Signal Processing Series*. Prentice Hall, Englewood Cliffs, NJ.
- Sabe, K., Hidai, K., Kawamoto, K., and Suzuki, H. (2005). A proposal of intelligence model, MINDY for open-ended learning system. *Proceedings of IEEE International Conference on Humanoid Robots Workshop on Intelligence Dynamics*.
- Shatkay, H. and Kaelbling, L. P. (1997). Learning topological maps with weak local odometric information. *Proceedings of Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97)*, pp. 920-929.

Formalization of different learning strategies in a continuous learning framework

Danijel Skočaj Matej Kristan Aleš Leonardis
University of Ljubljana
Faculty of Computer and Information Science
Tržaška 25, SI-1001 Ljubljana, Slovenia
{danijel.skocaj,matej.kristan,ales.leonardis}@fri.uni-lj.si

Abstract

While the ability to learn on its own is an important feature of a learning agent, another, equally important feature is ability to interact with its environment and to learn in an interaction with other cognitive agents and humans. In this paper we analyze such interactive learning and define several learning strategies requiring different levels of tutor involvement and robot autonomy. We propose a new formal model for describing the learning strategies. The formalism takes into account different levels and types of communication between the robot and the tutor and different actions that can be undertaken. We also propose appropriate performance measures and show the experimental results of the evaluation of the proposed learning strategies.

1. Introduction

An important characteristic of a robot that operates in a real-life environment is the ability to expand its current knowledge. The system has to create and extend concepts by observing the environment – and has to do so continuously, in a life-long manner.

As an example of such a learning framework, we need look no further than at the successful application of *continuous learning* in human beings. As humans, we can learn, for example, a new visual concept (e.g., an object category, an object property, an action pattern, an object affordance, etc.) by encountering a few examples of one. Later, as we come across more instances, different to the original examples, we not only recognise them, but also update our representation of learned visual concepts based on the salient properties of the new examples and without having visual access to the previous examples. In this way, we update or enlarge our ontology in an efficient and structured way by encapsulating new information extracted from the perceived data, which enables adaptation to new visual inputs and the handling of novel situations we may encounter.

Since humans are social beings this learning often takes place not in isolation, but rather in communication with other people. This communication can facilitate learning by exposing the knowledge that other possess also to the learner. It is very important for a robot, which is supposed to operate in a real world environment, to possess similar capabilities as well. The robot should be able to learn by interacting with the environment and with other knowledgeable cognitive systems (e.g., a tutor), which may facilitate the learning process and make it robust and reliable.

In this paper we focus on such interactive continuous learning, where the robot is learning and continuously updating its knowledge autonomously or in a dialogue with a tutor. With respect to this, several learning strategies can be used; the robot can continuously learn while communicating with the tutor with different levels of tutor involvement and different levels of robot autonomy.

For performing a thorough analysis and evaluation of various learning strategies, it is necessary to formally describe the learning process and defined performance metrics. In this paper we propose such a formalism for specifying different learning strategies. In the proposed formal framework we also define four learning strategies ranging from tutor-driven to tutor-unassisted learning.

The paper is organised as follows. In the next section we first describe the related work. In Section 3. we then describe four learning strategies and in Section 4. the general formal model of learning strategies. This is followed by experimental evaluation of the presented learning strategies. The paper concludes with a final discussion and outlook.

2. Related work

A tutor's involvement by interaction plays an important role in the learning process in cognitive agents. Studies of human infants, for example (Pea, 1993), indicate that being able to exploit the expertise of others is a critical part of learning. Another point is the capability of the infants to take lead in the inter-

action, which is a foundation for many situated learning activities. Weng et al. (Weng et al., 2001) propose that similar measures should be undertaken in machine learning scenarios, in which the tutor should mentally *rise* the developmental robot through real-time interaction. This assumption is supported in the theory of cognitive development proposed by Vygotsky (Vygotsky, 1962), which states that social interactions are of essential importance for the development of individual intelligence. Building on a similar assumption, Thomaz (Thomaz, 2006) casts the machine learning problem as a strongly involved interaction between the human and the machine. As a feature of strong interaction (Thomaz, 2006) propose that the tutor has to have a *level of insight* into what the learner knows and which parts of the knowledge are ambiguous – the learner should be *transparent* to the tutor. In that respect, an involved interaction as a dialogue based learning scenario was also presented by Roy et. al (Roy and Pentland, 2002, Roy, 2002). Their system in (Roy and Pentland, 2002) was designed to learn word forms and visual attributes from speech and video recordings, and subsequently, Roy extended this work for generating spoken descriptions of scenes (Roy, 2002).

Researchers have dealt with various levels of tutor involvement in the process of learning in machines. At one extreme is an example in which the tutor is absent and the agent has to *learn on its own* starting from a very small or no prior knowledge, e.g., (Mugan and Kuipers, 2008, Oudeyer and Kaplan, 2004). However, allowing *learning from demonstration* (Argall et al., 2009) or *learning by imitating* (Schaal, 1999) the tutor can drastically reduce the search space for the agent’s task and speed up learning. Examples of implicit or explicit learning from a *passive observation* can be found, for example, in the works of (Kuniyoshi et al., 1994, Billard and Dautenhahn, 1999, Lieberman, 2001). Another level of tutor’s involvement is teaching by *directly influencing* the the actions of the machine. Such an example is when user biases the action selection in the machine (Maclin et al., 2005) or to allow direct control of robot’s actions to supervise the process of reinforcement learning (Smart and Kaelbling, 2002). Kaplan et al. (Kaplan et al., 2001) explored animal training techniques to teach a robot to perform complex tasks. An example where the tutor plays an oracle was explored by Schohn and Cohn (Schohn and Cohn, 2000) – in that scenario, the agent provides some *level of transparency* by identifying the relevant examples and querying the tutor for the required labels. Allowing the robot to *actively express its uncertainty*, or a gap in the knowledge, was explored in the ”Ask for

Help” framework (Clouse, 1996) and, for example, (Nicolescu and Mataric, 2003). An approach to reinforcement learning which can *learn from tutor’s feedback* was presented in (Knox and Stone, 2008).

Learning in cognitive robots can therefore be described in terms of different levels of tutor involvement as well as levels of learner’s responsiveness and learner’s transparency. As noted above, various researchers have dealt with scenarios with various levels of the tutor-learner interaction, leading to different learning strategies. With this respect, the closest related work is (Chernova and Veloso, 2009), where the authors propose and evaluate similar learning strategies to those discussed in this paper (although in a different learning domain). The main contribution of this paper, however, goes beyond the definition of the learning strategies; we also propose a formalism for modeling these strategies. In fact, also the learning strategies like those presented in (Chernova and Veloso, 2009) could be modeled with the formal model presented here. This is also the main goal of our work; to introduce a formalism that would enable simple and efficient definition, evaluation and comparison of different learning strategies.

3. Learning strategies

The interaction between the tutor and the robot plays an important role in a continuous learning framework. The goal of the learning mechanism is to continuously learn and update the acquired concepts, i.e., to find associations between the words spoken by the tutor (and related amodal concepts) and features, which are automatically extracted from the observations. Such a continuous learning framework should communicate with the tutor, perform recognition, and update the representations according to the current learning strategy. In this section we define several learning strategies which alter the behaviour of the system and require different levels of tutor involvement.

In the core of any learning strategy is a **learning algorithm** that actually builds and updates the representations. Before we proceed with the definition of the learning strategies, let us introduce several requirements for the learning algorithm.

Most importantly, the learning algorithm has to be **incremental**; the representation, which is used for modeling the observed world, has to allow for updates when presented with newly acquired information. This update step should be efficient and should not require access to previously observed data, while still preserving the previously acquired knowledge.

In addition, in continuous learning scenarios the noise in the input data has a detrimental effect on the learnt representations, especially when the robot learns autonomously. If, for example, the recognition algorithm fails at some point to correctly inter-

pret the visual scene and erroneously updates the current knowledge, the models of the concepts tend to degrade and the performance of the system will typically decrease severely. However, in interactive settings the tutor can help the robot to recover from the errors through interaction, by, e.g., indicating to the robot that its belief about a certain concept is wrong. The system should be then able to **unlearn**, i.e. to update the representation by considering the wrongly classified sample as a negative training example. Unlearning step may lead to the correction of the current representation, which can improve the performance considerably.

Finally, it is obvious that the system is supposed to have a certain level of **self-understanding**; it should be able to estimate whether its current knowledge suffices to interpret the current scene, or it should ask the tutor for help. Therefore, it should have a recognition capability, i.e., the ability to interpret the current observation to some extent. And even more importantly, the system should be able to evaluate the reliability of this recognition process.

We therefore assume that the learning algorithm, which is used in the continuous learning framework, fulfills the criteria mentioned above.

We define a **learning strategy** as a common strategy of the tutor and the robot that specifies the behaviour of the robot and the tutor in the continuous learning process. It specifies when the robot updates its knowledge autonomously and how and when the tutor and the robot communicate in order to extend the robot's knowledge. According to this definition and considering different levels of interaction between the tutor and the robot, various learning strategies are possible. Here we identify four such strategies:

- **Tutor-driven.** The tutor drives the learning by describing the observation and giving all available information to the robot. The communication is one-directional, the learning process is completely controlled by the tutor.
- **Tutor-supervised.** The robot establishes transparency; the tutor assesses the robot's knowledge and detects its ignorance. When the robot fails to correctly interpret the current observation, the tutor provides the correct information, which helps the robot to update or unlearn the current representations accordingly.
- **Tutor-assisted.** The robot tries to interpret the current observation. If it succeeds to do this reliably, it updates the current model, otherwise asks the tutor for the correct interpretation. The tutor therefore gives the information to the robot only when asked for assistance.
- **Tutor-unassisted.** The system updates the

model with the automatically obtained interpretation of the visual input. No assistance from the tutor is required. There is no communication between the tutor and the robot.

The dialogue in the first two learning strategies is initiated by the tutor, while in the second two cases the robot takes the initiative. These four learning strategies range across the entire spectrum of different levels of the tutor involvement and the robot's autonomy. In Tutor-driven mode the tutor completely drives the learning process, in Tutor-supervised mode he intervenes only when necessary, in Tutor-assisted mode only when he is asked for, and in Tutor-unassisted mode even never. On the other hand, the autonomy of the robot increases from Tutor-driven mode, where the robot does not influence the learning process, to Tutor-unassisted mode, where it completely autonomously controls the learning. This is also depicted in Fig. 1.

The spectrum of different learning modes is of course not discrete as presented here; it is continuous and one could define additional learning strategies with similar properties. It is also possible to combine different learning strategies, to execute them in a sequence and to switch between them when necessary. In practice, the learning strategy should change over time, adapting to the current level of knowledge and complexity and novelty of the environment the robot is currently situated in. We believe, however, that the presented four learning strategies span across the entire space of possible learning strategies and cover a major part of its variability.

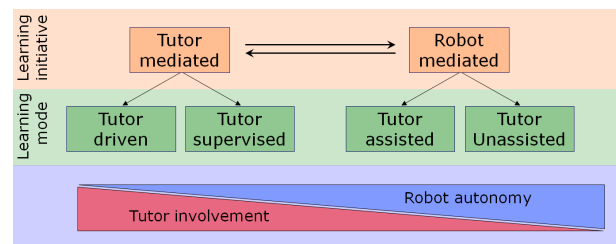


Figure 1: Learning strategies.

4. Formal model

In the previous section we have conceptually described a few possible learning strategies. Here we present a general formalism, which can be used to formally define these or many other learning strategies.

We will limit our analysis on the continuous learning scenarios, in which a robot observes a scene and learns new concepts through interaction with a tutor. This interaction can be quite simple or very complex; different learning strategies employ different levels of

communication. We assume that the robot and the tutor can establish the common ground; they have all necessary communication capabilities, they observe the same scene, and in the dialogue they refer to the same object.

The robot and the tutor are involved in a continuous and interactive learning process; the robot continuously observes objects, it tries to recognize them and learn something new about them. Every learning step therefore starts with the robot trying to interpret the current scene. It tries to recognize all the concepts it currently knows. Based on the classification confidence (see Fig. 2), the robot can assign **soft labels** when trying to determine whether the current observation is indicative of a given concept or not:

- **‘Yes’ (YES):** The recognition confidence is very high, the robot reliably classifies the current observation as being an instance of a particular concept.
- **‘Probably yes’ (PY):** The recognition confidence is relatively high, however the robot is not certain about its current interpretation.
- **‘Probably no’ (PN):** The recognition confidence is relatively low; the current observation probably does not indicate the particular concept.
- **‘No’ (NO):** The recognition confidence is very low, therefore the robot reliably classifies the current observation as not being an instance of a particular concept.
- **‘Don’t know’ (DK):** The recognition was not sufficiently reliable to determine the answer.
- **‘Unknown’ (UK):** The robot has not yet encountered the certain concept it was asked about.

Based on the output of the classifier and as instructed by the chosen learning strategy, one of the following four **actions** follows:

- **Do nothing.** The robot does not update its current knowledge nor does request an interaction with the tutor.
- **Autonomously update.** The robot updates the current knowledge with the information autonomously inferred from the current observation without involving the tutor.
- **Tell.** The tutor gives the correct information about the current observation to the robot.
- **Ask.** The robot asks the tutor for clarification about the current observation and the tutor replies with the correct answer.

In the latter three cases an update of the current knowledge follows (either based on the automatically extracted information or on information obtained by the tutor). Two different kinds of **update** are possible:

- **Update with a positive example.** The robot updates its current knowledge by integrating the positive training sample into its current representation of the particular concept.
- **Unlearn with a negative example.** The robot unlearns its current knowledge; based on the given negative example, it corrects the current representations not to model this negative example.

To fully describe the learning strategy we also need to define the intensity of communication between the robot and the tutor. We define three such **communication levels**:

- **Ignoring.** The tutor ignores the robot’s output; the state and performance of the robot do not influence the tutor’s behavior.
- **Listening.** The tutor listens to the robot and correctly answers with ‘yes’ or ‘no’ when being asked a polar question.
- **Transparency facilitated assessment.** The robot establishes transparency and the tutor is able to assess the robot’s current interpretation of the observation.

Now, let us denote the above mentioned four actions with the following signs: ‘/’ for ‘do nothing’, ‘U’ for ‘auto-update’, ‘T’ for ‘tell’, and ‘A’ for ‘ask’. In addition, with a suffix next to these signs we will denote an *update with positive example* with the plus sign (+) and an *unlearning* request with the minus sign (-). For instance, ‘U₊’ means that the system will automatically update the current knowledge with the information inferred from the current observation, while ‘A₋’ means that the robot will ask the tutor for clarification, the tutor will reply with a negative answer and the robot will unlearn its current knowledge accordingly. Similarly, let us denote the communication levels with ‘ign’ (*ignoring*), ‘lst’ (*listening*), and ‘tfa’ (*transparency facilitated assessment*).

To fully describe a learning strategy, we need to define what will happen if the robot correctly or incorrectly interprets the current observation with respect to all known concepts. Therefore, we need to define the action that will be undertaken depending on the robot’s autonomous interpretation of the scene (**soft label** *sl* that is autonomously assigned for a particular concept). We assume that the tutor is omniscient and always gives the correct information to the robot; therefore the tutor’s actions will

also depend on the **ground truth data** (gt), which tells if the observation is an instance of the particular concept or not.

Now, a learning strategy can be defined as a 13-tuple LS :

$$\begin{aligned}
 LS &= [act_{sl,gt}, cl], \text{ where} & (1) \\
 sl &\in \{YES, PY, PN, NO, DK, UK\} \\
 gt &\in \{yes, no\} \\
 act_{..} &\in \{/, U_+, U_-, T_+, T_-, A_+, A_-\} \\
 cl &\in \{ign, lst, tfa\}
 \end{aligned}$$

Note that $act_{sl,gt}$ denotes 12 elements (2×6 combinations of sl and gt , i.e., $act_{YES,yes}$, $act_{YES,no}$, $act_{PY,yes}$, etc.¹). This vector exactly specifies what will happen in certain situations. When the robot observes a new observation it tries to determine whether it belongs to a certain concept or not, and assigns a soft label (sl) as described above. This label is then together with the known ground truth (gt) used to index in the vector LS ; the obtained action $act_{sl,gt}$ exactly specifies which action (or sequence of actions) will be undertaken.

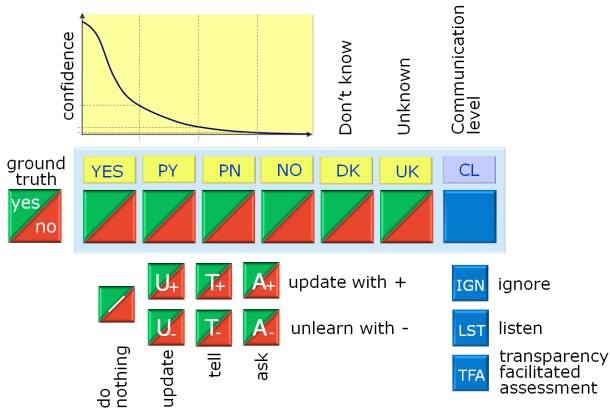


Figure 2: Parametrisation of learning strategies.

To demonstrate this formalism, let us formally define the four learning strategies presented in the previous section (see also Fig. 3):

$$\begin{aligned}
 LS_{TD} &= [T_+, /, T_+, /, T_+, /, T_+, /, T_+, /, T_+, /, ign] \\
 LS_{TS} &= [U_+, T_-, U_+, T_-, T_+, /, T_+, /, T_+, /, tfa] \\
 LS_{TA} &= [U_+, U_+, A_+, A_-, A_+, A_-, /, /, A_+, A_-, T_+, /, lst] \\
 LS_{TU} &= [U_+, U_+, U_+, U_+, /, /, /, /, /, /, T_+, /, ign]
 \end{aligned}$$

In *Tutor-driven* learning mode, the tutor ignores the output of the robot (ign); it always gives the robot the correct (positive) information about the current observation (T_+). In *Tutor-supervised* mode, the tutor observes the robot and assesses its current knowledge (tfa). The tutor lets the robot automatically update the current knowledge (U_+), when its

¹With capital letters (e.g., YES), we denote the label autonomously inferred by the robot, while with small letters (e.g., yes) we denote the actual (ground truth) label for a particular concept.

interpretation is correct, or he corrects the robot, when its interpretation is incorrect by telling it the correct information (T_- or T_+). In *Tutor-assisted* mode the tutor listens to the robot (lst), which autonomously decides either to update the knowledge automatically (U_+), when it trusts to its recognition result, or to ask the tutor for help, when the recognition was not reliable. In the latter case, the tutor responds with ‘yes’ (A_+) or ‘no’ (A_-) according to the ground truth label, which in turn enables the robot to update or unlearn its current knowledge. Finally, in the *Tutor-unassisted* learning, the robot only relies on its current recognition abilities and does not ask the tutor for help. The robot is therefore ignored by the tutor (ign) and updates its current knowledge autonomously (U_+).

	YES	PY	PN	NO	DK	UK	CL
TD	T_+	T_+	T_+	T_+	T_+	T_+	IGN
TS	U_+	U_+	T_+	T_+	T_+	T_+	TFA
TA	U_+	A_+	A_+	A_-	A_+	T_+	LST
TU	U_+	U_+	U_+	U_+	U_+	T_+	IGN

Figure 3: Formal definition of four learning strategies.

Such learning formalism allows us to formally define evaluation measures. Instead of standard recognition rate we propose to use a **recognition score**, which rewards successful recognition (true positives and true negatives) and penalizes incorrectly recognised concepts (false positives and false negatives) by taking into account soft labels. The scoring rules are presented in Table 1; it shows how many points (-1 to 1) the system is rewarded with for each of the answers given in the first row, depending on the correct answer as given in the first column.

Table 1: Scoring table.

	YES	PY	PN	NO	DK	UK
yes	1	0.5	-0.5	-1	0	0
no	-1	-0.5	0.5	1	0	0

The recognition score thus measures how successfully the robot recognizes the learned concepts (therefore, how successful the learning was). However, in interactive learning scenarios another criterion is also important; the **tutoring costs**. Obviously, one would prefer that the robot learns autonomously as much as possible, without involving the tutor too frequently. During the learning process different types of tutoring costs may occur (in

different learning strategies):

- C_{inf} : costs of providing some information to the robot.
- C_{ans} : costs of answering a polar question to the robot.
- C_{ign} : costs of ignoring the robot’s output.
- C_{lst} : costs of listening to the robot.
- C_{tfa} : costs of assessing the current robot’s knowledge.

Let us suppose that at a particular learning step the tutor gave N_{inf} concept labels about the correct observation to the tutor and answered N_{ans} polar questions. Now we can define the overall tutoring costs at that particular learning step as

$$TC = N_{inf}C_{inf} + N_{ans}C_{ans} + C_{cl} \quad (2)$$

where cl is one of three communication levels as defined above.

The values of the parameters C_* depend on the actual costs that occur during the interactive learning. In this paper we use the values presented in Table 2. We set the cost of assessing the robots knowledge

Table 2: Tutoring costs.

C_{inf}	C_{ans}	C_{ign}	C_{lst}	C_{tfa}
1	.25	0	.25	2

high, since this is not a trivial task for the tutor. If, for instance, the robot would establish the transparency by verbalizing its current beliefs, the tutor would just have to listen to it and the cost of assessing the knowledge would be lower, i.e., $C_{tfa} = C_{lst}$.

5. Experimental results

For performing large scale experiments and evaluating different learning strategies we have developed *Interactive Continuous Learning Simulator*, which implements the formal model of learning strategies presented in the previous section. This simulation environment uses as observations the features that were automatically extracted from the previously captured, automatically processed and manually labeled real data; the tutor is replaced by an omniscient oracle, which has the ground truth data available. The simulator enables large scale experiments and a thorough evaluation and comparison of different learning methods and strategies.

We performed a number of experiments to evaluate different learning strategies on different learning domains. Here we present the results of the experiment where the goal was to learn basic visual attributes like colour and shape by observing

a set of everyday objects (some of them are depicted in Fig. 4(a)). Six visual attributes were considered; four colours (red, green, blue, yellow) and two shapes (elongated, compact). The database that we used for learning contains 500 images. 400 images were used to incrementally learn the representations of six visual properties, while the rest 100 of them were used as test images. We repeated the experiment for 100 runs by randomly splitting the set of images into the training and test set and averaged the results across all runs. In all the experiments we used the extended algorithm for incremental learning that we have previously proposed (Skočaj et al., 2008, Kristan et al., 2009).

During the experiment, we kept incrementally updating the representations with the training images using different learning strategies as defined in the previous section. At each step, we evaluated the current knowledge by recognising the visual properties of all test images. The learning performance was evaluated using two above defined performance measures: recognition score and tutoring costs.

Figs. 4(b,c) show the evolution of the learning performance over time for all four learning strategies. First thing to note is that the overall results improve through time. The growth of the recognition score is very rapid at the beginning when new models of newly introduced concepts are being added, and still remains positive even after all models are formed due to refinement of the corresponding representations.

Tutor-driven and Tutor-supervised learning yield similar recognition score; they almost achieve the perfect score (600 in this case). Tutor-supervised learning performs slightly better, since it sooner achieves better results. This is somehow expected, since in this case the tutor corrects the robot when necessary and the robot unlearns the erroneous representations. The inherent problem of any continuous learning framework, which involves autonomous updating of the knowledge, is propagation of errors. The tutor supervision efficiently helps the robot to recover from this errors, if the robot transparency has been achieved. The error recovery is in this experiment less effective in the Tutor-assisted case. The errors are in this case detected by the robot (and not by the tutor). Obviously, this error detection is not so efficient, therefore the recognition score is lower. In this experiment, Tutor-unassisted learning did not perform well; without sufficiently good initial knowledge it was not able to improve without any assistance from the tutor.

We also have to take into account the tutoring costs that occur during the learning. In Tutor-driven learning mode they are almost constant; the tutor always gives all the information about the current object, which is available. The costs of Tutor-assisted learning are significantly lower. The robot keeps ask-

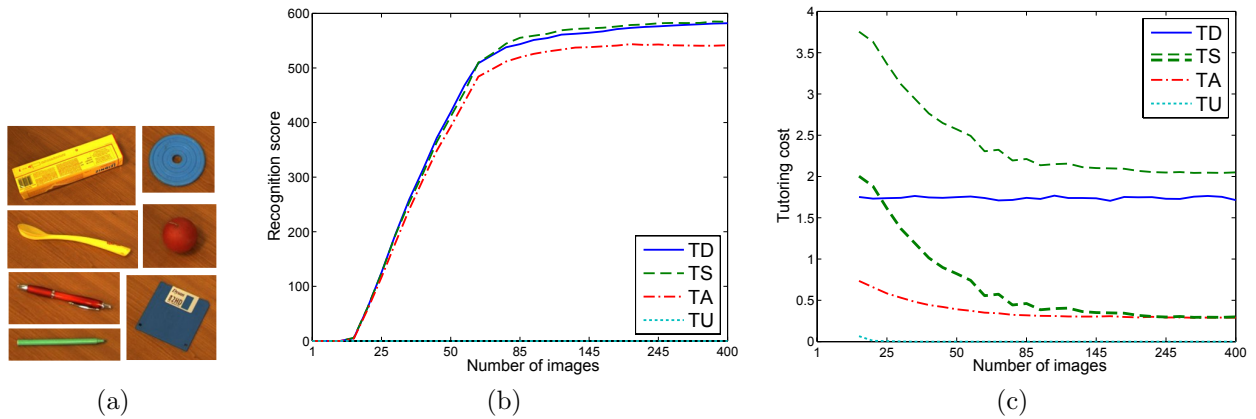


Figure 4: Experimental results: (a) Seven everyday objects from the database. (b) Evolution of Recognition Score, (c) Tutoring Costs. Note the logarithmic scale along abscissa.

ing the tutor only at the beginning of the learning process; after its knowledge gets improved the number of questions drops and most of the costs relate to the fact that the tutor has to listen to the robot and await for its questions. The costs of Tutor-supervised learning are relatively high, since in this experiment we use the settings presented in Table 2, which assume that it is relatively expensive to assess the robot’s knowledge. In addition to that, at the beginning there is a lot of communication between the tutor and the robot, which again drops when the models of the concepts get stabilized. If the robot establishes its transparency by verbalizing its beliefs about current observations, the costs of assessing the knowledge are significantly lower, and the overall tutoring costs significantly decrease (the strong dashed line in Fig. 4(c)), making Tutor-supervised learning more efficient than the Tutor-driven. This holds true also in practice; it is more convenient (and effective) for the tutor just to listen and correct the learner occasionally than to continuously giving it new information.

6. Conclusion

In this paper we have introduced a new formal model for formalizing learning strategies. We define a learning strategy as a common strategy of the tutor and the robot that specifies the behaviour of the robot and the tutor in the continuous learning process. The formalism takes into account different levels and types of communication between the robot and the tutor and different actions that can be undertaken. By specifying these actions and communication levels, the learning strategy can be uniquely defined.

In general, it is very difficult to objectively compare different (incompatible) learning processes; the presented formalism makes this comparisons straightforward. This will allow us to analyse different learning strategies, to efficiently combine them

and to find a way how to exploit the properties of the individual strategy best.

In addition, we introduced four learning strategies that span across the entire space of possible learning strategies and cover a major part of its variability. They range across the entire spectrum of different levels of the tutor involvement and the robot’s autonomy. We also evaluated these four learning strategies using the proposed performance metrics.

While the currently presented formalism may appear to simplistic to apply to richer scenarios with shifting the focus of attention and more complex dialogues, we believe that it forms a solid base of building blocks for basic tutor-learner interaction. In our future work we will build upon this base and establish means of combining these blocks into more complex framework which will account for more complex situations.

Our primary goal is to develop a robot that would be able to efficiently acquire new concepts and to update the existing ones in collaboration with a human teacher. We have implemented the learning strategies introduced in this paper on a real robot (for details the reader is referred to (Vrečko et al., 2009)). When conducting research on interactive learning it is crucial to have a real implementation of the learning framework on real robots and to test its functionality in real-world settings. However, it is equally important also to have formalisms and tools to perform large scale experiments, which enable thorough evaluation and analysis of the proposed methods. We believe that the proposed formal model can facilitate such research and enable further development of related approaches.

Acknowledgements

This research has been supported in part by the EU FP7 project CogX (ICT-215181) and the Research program Computer Vision (RS).

References

- Argall, B., Chernova, S., Veloso, M., and Browning, B. (2009). A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469–483.
- Billard, A. and Dautenhahn, K. (1999). Experiments in learning by imitation - grounding and use of communication in robotic agents. *Adaptive Behavior*, 7(3/4):415–438.
- Chernova, S. and Veloso, M. (2009). Interactive policy learning through confidence-based autonomy. *Journal of Artificial Intelligence Research*, 34:1–25.
- Clouse, J. (1996). *On integrating apprentice learning and reinforcement learning*. PhD thesis, University of Massachusetts Amherst.
- Kaplan, F., Oudeyer, P.-Y., Kubinyi, E., and Miklosi, A. (2001). Taming robots with clicker training: a solution for teaching complex behaviors. In *European workshop on learning robots, LNAI*, Springer.
- Knox, W. and Stone, P. (2008). TAMER: Training an agent manually via evaluative reinforcement. In *IEEE 7th International Conference on Development and Learning*, pages 292–297.
- Kristan, M., Skočaj, D., and Leonardis, A. (2009). Online kernel density estimation for interactive learning. *Image and Vision Computing*.
- Kuniyoshi, Y., Inaba, M., and Inoue, H. (1994). Learning by watching: Extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation*, 10:799–822.
- Lieberman, H., (Ed.) (2001). *Your Wish is My Command: Programming by Example*. Morgan Kaufmann, San Francisco.
- Maclin, R., Shavlik, J., Torrey, L., Walker, T., and Wild, E. (2005). Giving advice about preferred actions to reinforcement learners via knowledge-based kernel regression. In *National Conference on Artificial Intelligence*.
- Mugan, J. and Kuipers, B. (2008). Towards the application of reinforcement learning to undirected developmental learning. In *8th International Conference on Epigenetic Robotics*.
- Nicolescu, M. and Mataric, M. (2003). Natural methods for robot task learning: Instructive demonstrations, generalization and practice. In *Proceedings of the second international joint conference on Autonomous agents and multi-agent systems*, pages 241–248. ACM New York, NY, USA.
- Oudeyer, P.-Y. and Kaplan, F. (2004). Intelligent adaptive curiosity: a source of self-development. In *Proceedings of the 4th International Workshop on Epigenetic Robotics*, volume 117, pages 127–130. Lund University Cognitive Studies.
- Pea, R. D. (1993). *Distributed cognitions: Psychological and educational considerations*, chapter Practices of distributed intelligence and designs for education, pages 47–87. Cambridge University Press, New York.
- Roy, D. K. (2002). Learning visually-grounded words and syntax for a scene description task. *Computer Speech and Language*, 16(3):353–385.
- Roy, D. K. and Pentland, A. P. (2002). Learning words from sights and sounds: a computational model. *Cognitive Science*, 26(1):113–146.
- Schaal, S. (1999). Is imitation learning the route to humanoid robots? *Trends in Cognitive Science*, 3(6):233–242.
- Schohn, G. and Cohn, D. (2000). Less is more: Active learning with support vector machines. In *17th International Conference on Machine Learning*, pages 839–846.
- Skočaj, D., Kristan, M., and Leonardis, A. (2008). Continuous learning of simple visual concepts using Incremental Kernel Density Estimation. In *International Conference on Computer Vision Theory and Applications*, pages 598–604.
- Smart, W. D. and Kaelbling, L. P. (2002). Effective reinforcement learning for mobile robots. In *IEEE International Conference on Robotics and Automation*, pages 3404–3410.
- Thomaz, A. L. (2006). *Socially Guided Machine Learning*. PhD thesis, Massachusetts Institute of Technology.
- Vrečko, A., Skočaj, D., Hawes, N., and Leonardis, A. (2009). A computer vision integration model for a multi-modal cognitive system. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*.
- Vygotsky, L. (1962). *Thought and Language*. Cambridge, MA: MIT Press.
- Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., and Thelen, E. (2001). Autonomous mental development by robots and animals. *Science*, 291(5504):599 – 600.

Learning the Sensorimotor Structure of the Foveated Retina

Jeremy Stober and Lewis Fishgold
Department of Computer Sciences
The University of Texas at Austin
{stober,lewfish}@cs.utexas.edu

Benjamin Kuipers
Computer Science and Engineering
University of Michigan
kuipers@umich.edu

Abstract

We identify two properties of the human vision system, the foveated retina, and the ability to saccade, and show how these two properties are sufficient to simultaneously learn the structure of receptive fields in the retina and a saccade policy that centers the fovea on points of interest in a scene.

We consider a novel learning algorithm under this model, *sensorimotor embedding*, which we evaluate using a simulated roving eye robot on synthetic and natural scenes, and physical pan/tilt camera. In each case we compare learned geometry to actual geometry, as well as the learned motor policy to the optimal motor policy. In both the simulated roving eye experiments and the physical pan/tilt camera, our algorithm is able to learn both an approximate sensor map and an effective saccade policy.

The developmental nature of sensorimotor embedding allows an agent to simultaneously adapt both geometry and policy to changes in the physical model and motor properties of the retina. We demonstrate adaptation in the case of retinal lesioning and motor map reversal.

1. Introduction

In the human eye, the retina is a non-uniform array of photoreceptive rod and cone cells. The human retina has a foveal pit, a single region of maximum density of cone photoreceptors. In addition, a human can change the location of the retina relative to a scene through ballistic actions known as saccades (Palmer, 1999). The combination of a small, high-resolution fovea with the ability to saccade to regions of interest is an economical strategy for both humans and robots to achieve high-resolution vision across large fields of view.

Gathering and interpreting visual information requires a *motor map* and a *sensor map* of the retina. The motor map encodes the motor commands necessary to move the eye to new locations in the visual scene and is used in generating saccades. The sensor map represents the geometric structure of the retina, specifically the positions of

sense elements within the sensor array, and can be used to perform geometric operations on the visual signal such as edge detection. We show how, by exploiting the relationship between motor commands and sensor geometry, an autonomous agent with foveated vision can *simultaneously* learn both the motor and sensor maps.

For simple sensors, these maps can be manually specified, but as sensors become more complex and adaptive, learning approaches such as ours are of increasing value to robotics. In addition, as lifetimes of autonomous robots increase, the robust nature of this developmental approach will allow robots to adapt to changing sensors and motors.

2. Related Work

2.1 Learning Motor Maps

In previous work on learning motor maps for saccades, the learning was driven by the two-dimensional difference between the pre-saccadic and post-saccadic position of a target on the retina. These models assume that the structure of the retina is known when learning the motor map, allowing calculation of the distance between a target and the fovea.

In (Pagel et al., 1998) the authors use learning to improve upon rough predictions made by first-principle geometric calculations. They represented the motor map using growing neural gas. Using a training scheme that involves corrective saccades, the agent experiences more training examples in the foveal region, causing an increase in the density of units in the region of the motor map that represents the fovea.

In (Rao and Ballard, 1995) the authors also used a strategy based on corrective saccades. They relied on the ability to locate a point of interest in the post-saccadic image using multiscale spatial filters, though the ability to locate interest points using this method may be too strong an assumption for a young infant with an immature visual cortex (Slater, 1999).

In (Shibata et al., 2001), the authors use fifth order splines and saliency maps (Itti and Koch, 2001) to generate realistic saccade trajectories and that closely resemble human motion. In this work, we opt for a simpler saccade

model that allows us to learn both sensor and motor maps simultaneously.

The model used in (Weber and Triesch, 2006) is one of the most recently published models and is the most similar to ours. Like us and unlike previous work, they use an error signal based on total retinal activation, exploiting cases where the total activation of a foveated retina is proportional to the degree of success of a saccade. Their model treats learning the horizontal and vertical components of saccades separately in accord with the experimental results of (Noto and Robinson, 2001).

2.2 Learning Sensor Maps

In previous work on learning sensor maps, (Pierce and Kuipers, 1997) demonstrated how sensor maps for a mobile robot can be discovered from uninterpreted high-dimensional sensor streams while motor babbling, and (Olsson et al., 2006) later extended these results to physical robots with visual perception. These studies generate sensor maps using dimensionality reduction algorithms that discover low-dimensional sensor arrangements that approximate distances between sensor trace histories. Two sensors are close in the sensor map if their corresponding sense histories are highly correlated.

In this work, we take a complementary but related approach and exploit some additional available structure, namely the availability of motor commands. We base our embedding, which we call *sensorimotor embedding*, on the motor system’s ability to change the sensory signal.

The algorithm we present here utilizes the relationship between sense and action to *simultaneously* extract useful geometric features (i.e. sensor position) along with primitive animate vision behaviors. Our method is appropriate for cases with an easily identifiable reward signal (e.g. activation), linear ballistic motor commands, and a high number of sense elements. We exploit the structure of the sensorimotor domain to produce an explicit mapping between motor commands and sensor features. This map has two interpretations, one as a primitive behavior that maximizes reward (the policy interpretation), and another as a structure for the sensor array (the geometric interpretation).

3. A Foveated Retina

3.1 Model

Our abstract model of the foveated retina is inspired by the anatomy of the human retina. In our model, a retina is a collection of receptive fields, or sense elements, with fixed geometry arrayed across a two dimensional surface. Each receptive field responds to sensory input from a portion of an image or scene according to its own activation function. Our learning rule requires that the distribution of activations across the retina be non-uniform and achieve a single maximum at the fovea. In addition, un-

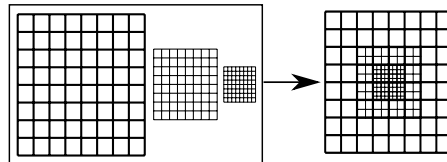


Figure 1: Our implementation of the fovea consists of overlapping layers of receptive fields. As the layer resolution increases, the extent of each receptive field decreases, and the number of bits necessary to describe the layer state remains constant.

der our model, ballistic motions instantaneously change the location of the retina in an image or scene.

Many implementations of a foveated retina satisfy this model. In biological systems, receptive fields are often distributed according to a log-polar distribution (Schwartz, 1977) and many computational models of saccade generation build upon this model of foveation (Weber and Triesch, 2006, Rao and Ballard, 1995). For this work, we view the specific distribution of receptive fields as an implementation issue, and expect that any distribution that satisfies the modeling assumptions above will behave similarly to our implementation.

3.2 Implementation

In our implementation, the learning agent has a foveated retina with N layers of receptive fields (Figure 1). Each layer has receptive fields of uniform extent and resolution. Layers with higher resolution and smaller extent overlap layers with lower resolution in the center of the retinal field of view. The fovea is the region with the highest concentration of overlapping receptive fields, and is also the region of maximal activation, so this implementation satisfies the model assumptions specified above. We stress that alternative implementations satisfying the model assumptions should behave similarly.

The implementation of each individual receptive field may also vary. In this case, each receptive field must map a patch of underlying pixel or sensor values to an activation level. Let I_k denote the image patch that affects the state of the k^{th} receptive field. Let \mathcal{I} denote the set of all such patches.

In addition to the image patch associated with each receptive field, the activation depends on the global state of the entire retina. In the case of a pan/tilt camera, we can describe the retina state using the horizontal and vertical angle of the camera lens (θ, ϕ) . In the case of the roving eye, we can describe the state of the retina in terms of the horizontal and vertical offsets (u, v) that describe the position of the retina in the larger image. However the state space is parametrized, we denote the set of all states by \mathcal{S} .

We require that the receptive field implement an activation function $\delta : \mathcal{I} \times \mathcal{S} \rightarrow [0, 1]$. In our implementation, $\delta(I_k, s)$ is the total activation of the pixels in the image patch I_k given the current retina state s , normalized to

$[0, 1]$ as a fraction of the maximum possible activation.

The activation over the entire retina is the sum of the activations for each receptive field for the current retina state,

$$R_{\mathcal{I}}(s) = \sum_{I_k \in \mathcal{I}} \delta(I_k, s) \quad (1)$$

4. Reinforcement Learning Problem

In our computational model, saccades result in 2D displacements of the image on the retina or pan/tilt changes for a physical camera. Each action or saccade $a : \mathcal{S} \rightarrow \mathcal{S}$ is described by two-element vector denoting horizontal and vertical motion and results in a single globally rigid transformation of the image or scene.

If the receptive fields in the retina are of uniform size and distribution, and they are exposed to input consisting of a small spot of light against a uniform background, then $R_{\mathcal{I}}(s)$ would be approximately constant for all retinal states s , regardless of where the spot of light falls. However, with a *foveated* retina, $R_{\mathcal{I}}(s)$ will have a dramatic maximum for retina states that cause the spot of light to fall on the fovea, due to the larger density of receptive fields there.

Using the total activation of all the receptive fields for the current retina state, $R_{\mathcal{I}}(s)$ in Equation 1 as the reward, combined with saccade actions, we can define a simple reinforcement learning problem, the goal of which is to find a policy, or choice of action, that maximizes retinal activation.

We factor the global learning problem into an individual learning problem for each receptive field. The goal of each receptive field is to learn a policy that greedily maximizes the total retinal activation $R_{\mathcal{I}}(s)$,

$$\pi_k(s) = \arg_a \max R_{\mathcal{I}}(a(s)) \quad (2)$$

The problem is episodic and spans a pre- and post-saccade state. The collective policy π^* for the entire retina is the weighted average of the actions preferred by the individual receptive fields,

$$\pi^*(s) = \frac{1}{R_{\mathcal{I}}(s)} \sum_{I_k \in \mathcal{I}} \delta(I_k, s) \cdot \pi_k(s) \quad (3)$$

In this factored learning problem, the only information a receptive field has about the state of the retina is the intensity level for that receptive field's visible patch I_k . If the intensity is high ($\delta(I_k, s)$ is close to 1), then the policy $\pi_k(s)$ will have a large impact on the global policy calculated in Equation 3. In this case, we want the policy to suggest an action $\pi_k(s) = a$ that maximizes the reward $R_{\mathcal{I}}(a(s))$. The action that accomplishes this takes the activation that the current receptive field sees and shifts it to the fovea, where the density of receptive fields is higher.

If the intensity is low, then the policy for that receptive field will have little impact on the policy for the entire retina since $\delta(I_k, s)$ is close to zero. As a consequence,

we can treat $\pi_k(s)$ as a constant. So in the factored problem, each receptive field only needs to estimate the optimal action and observe its own intensity level.

We predict that (after sufficient training), the action specified by π_k will approximate the saccade that moves an image-point from receptive field k directly to the fovea. Consider the inverse $-\pi_k$ of the policy estimate for each receptive field. This is the action that would move an image-point from the fovea to the receptive field k . In other words, the inverse of the policy is a position for the receptive field relative to the fovea. We expect that physically proximate receptive fields will have similar saccade policies, and hence similar learned positions. Note that we have not used any knowledge of the location of receptive fields within the fovea. In fact, that knowledge has been learned by the training process, and is encoded in the policy π_k . Spatial knowledge that was implicit in the anatomical structure of the retina becomes explicit in the policy.

The reinforcement learning problem described above has two unusual properties that constrain the choice of learning algorithm. First, the action space is continuous (as opposed to small and discrete). Second, the problem is episodic, and each episode spans only one choice of action.

During learning, each receptive field maintains an estimate for π_k , the current best action, and R_k , the current maximum estimated reward after performing the current best action. Initially, each π_k is set to a random action, and the reward estimate is initialized to zero.

At the beginning of each iteration or training, we randomly reposition the retina. For exploration, some noise ϵ is added to the current greedy policy. The retina agent executes $\pi^*(s) + \epsilon$, and measures the reward (R). Each individual receptive field's reward estimate and current policy are updated proportional to its state activation prior to the saccade ($\delta_k = \delta(I_k, s)$) since the optimal policy π^* is weighted according to those activations. We use a moving average learning rule to update both the reward estimate and current policy. For each receptive field k , we update the reward as follows

$$R_k^{new} = \begin{cases} R_k^{old} + \delta_k \cdot \alpha \cdot (R - R_k^{old}) & \text{if } R > R_k^{old} \\ R_k^{old} & \text{otherwise} \end{cases} \quad (4)$$

If the reward received, R , is greater than our current reward estimate, we move the current policy π_k for that receptive field closer to the global policy responsible for the increased reward

$$\pi_k^{new} = \pi_k^{old} + \delta_k \cdot \alpha \cdot (\pi^* - \pi_k^{old}) \quad (5)$$

By varying the learning rate α , we can change how much recent experience affects both the estimate of reward (R_k) and the estimate of the optimal saccade (π_k) itself. We discuss cases where R_k may decrease in Sections 5.2 and 5.3.

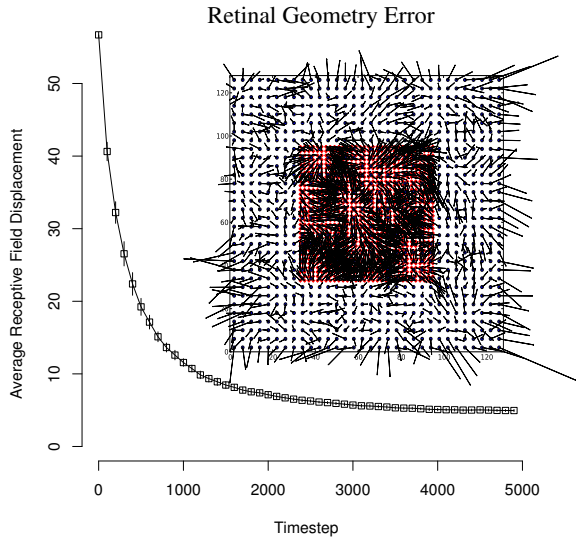


Figure 2: This figure plots the mean geometric error as a function of training time. The mean and standard errors are shown for ten independent training runs using a single dot image. The subfigure shows the result of interpreting learned receptive field policies as positions. Each line represents the error between the true position and learned position — the head (dot or diamond depending on the layer) is the true location of the field. The tail is the learned position. For clarity, only two layers are shown.

5. Experimental Evaluation

5.1 Simulated Saliency

We trained a simulated foveated retina with four layers of receptive fields on an image with a single white spot on a black background, meant to simulate the result of a saliency map. Each retina layer contained 32×32 receptive fields. The extent of each receptive field varied by layer, with the largest layer having receptive fields of size 4×4 (for a total retinal pixel area of 128×128). Actions corresponded to horizontal and vertical translations of the retina across the image.

We randomly initialized the policy for each receptive field and used a training rate $\alpha = 0.5$. ϵ was normally distributed with a mean of 0 and a standard deviation of 10 pixels.

We use two criteria to measure the success of our learning algorithm. The first computes the mean of the Euclidean distances between the learned position (interpreted as the additive inverse of the policy) and the true position $pos(I_k)$ of all receptive fields (Equation 6).¹ The results of training are shown in Figure 2.

$$E_{geometry} = \frac{1}{N} \sum_{k=1}^N \| -\pi_k - pos(I_k) \|_2 \quad (6)$$

For the second criterion, we compare the accuracy of the learned saccade against the optimal saccade, which

¹This analysis compares pixel positions to action space positions. This is only possible since translations of the roving eye retina are specified in pixels. In experiments using a pan/tilt camera, we do not have the same access to error free ground truth actions.

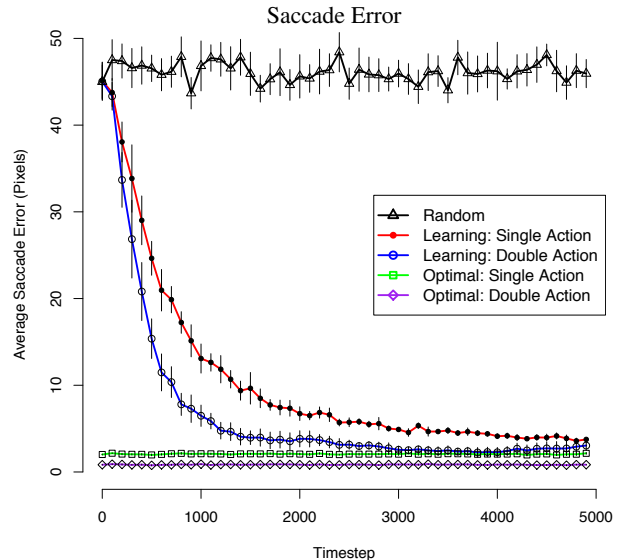


Figure 3: The saccade error as a function of the number of training iterations using the learning algorithm of Section 4. The saccade error is computed over thirty random repositions every 100 timesteps for ten independent trials. Note that even with an optimal policy, saccades are not entirely accurate because of low resolution in the periphery of the retina.

would center the retina on the area of high activation. We also test two-saccade accuracy, where the retina makes a second saccade after the first during testing but not training.

During the training process, every 100 training steps, we stop training and test saccade and two saccade accuracy for 30 random repositions. The average and standard errors of the accuracies over ten training trials are shown in Figure 3, which also includes comparisons with a randomly initialized policy and an optimal policy (where each policy is initialized to the inverse of that receptive field's position).

The learning algorithm achieves near-optimal saccade accuracy after 5000 training steps. Comparing Figures 2 and 3, we see that the geometric error decreases as accuracy increases, though the final sensor map only approximates the true positions of the receptive fields. Our algorithms final saccade error of 5 pixels is less than that of (Pagel et al., 1998) and requires only a quarter of the number of training steps.

5.2 Lesioning

In natural scenes, or in cases where the number of receptive fields in the fovea changes as with macular degeneration, the maximum achievable reward changes. In these cases, the maximum achievable reward may decrease to a level below the current reward estimate for each receptive field, $R < R_k^{old}$ and so no updates will take place. To account for this kind of variation over time, we can change the learning rule to maintain a recency-weighted average estimated reward, instead of maintaining an estimate of maximum reward.

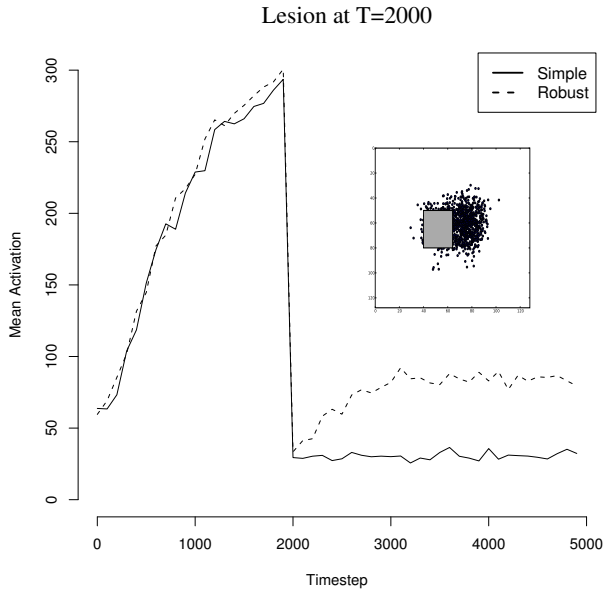


Figure 4: As a result of lesioning, a retina, with a robust learning rule as described in this section, adapts its policy to favor saccades to regions just outside the damaged region (see subfigure), providing higher post-saccadic activation in the case of lesioning than the previous optimal saccades directly to the fovea. We note that this increases the position error relative to the ground truth, but provides a coordinate system consistent with the sensorimotor properties of the damaged retina. The basic learning rule from Section 4 fails to adapt following a lesioning event.

This learning rule would require that the reward estimate be updated each timestep

$$R_k^{new} = R_k^{old} + \delta_k \cdot \alpha \cdot (R - R_k^{old}) \quad (7)$$

instead of only updating during timesteps where $R > R_k^{old}$.

We tested the ability of this modified algorithm to adapt to lesioning a small off-center part of the foveal region of the retina after 2000 steps of normal training. The mean post-saccade activation increases after lesioning when the agent uses the the robust learning rule (Figure 4). The basic learning rule, however, does not adapt to the lesioning event.

5.3 Motor Map Reversal

The modified algorithm presented above to deal with lesioning may require very high sample complexity to properly adapt to large changes in the motor model of the foveated retina.

Even though the reward estimates for each receptive field would adjust downward after a large change in the semantics of the motor commands, exploration still depends on adding noise to the previous policy estimate for each receptive field. In cases where the motor model changes radically, this exploratory bias may handicap any attempt to learn an alternative motor map.

Humans have shown some capacity for adapting to drastic changes in sensorimotor experience. For exam-

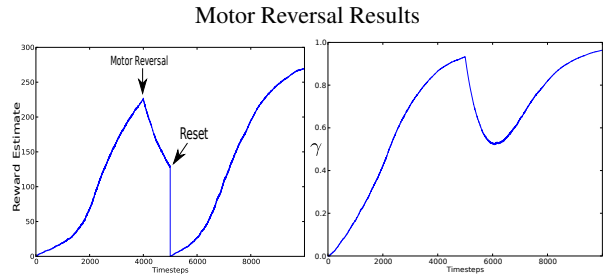


Figure 5: The left figure shows the moving average estimate of rewards experienced during training. A reversal in the motor map occurs after 4000 timesteps results in a decrease in the moving average reward estimate. After decreasing over 1000 timesteps, the retina resets the rewards estimate and the estimates for each receptive field and begins adapting to the new motor model. This results in a decrease in γ and an increase in exploration as shown on the right.

ple, in a self study using prismatic inverting eye-wear (Dolezal, 1982), Dolezal reports both initial difficulty in simple reaching tasks followed later by comfortable mastery.

In Dolezal’s inverted perceptual world, pointing up results in the visual perception of pointing down. By reversing the result of a motor command along one axis, we can simulate a similar (but less complex) change in the relationship between the motor actions and perceptual response. Though our experiment does not capture the full range of altered sensorimotor contingencies presented in (Dolezal, 1982), this experiment illustrates the need for a different kind of adaption in the face of significant changes in sensorimotor contingencies.

In this modification, each receptive field maintains an estimate of the optimal reward and policy as before. The retina also maintains an estimate of the maximum observed reward, a moving average of all the observed rewards, along with the reward estimates associated with each receptive field.

The exploration/exploitation trade-off is driven by a parameter, γ , that is meant to measure the extent to which the learned policy for currently active receptive fields will be able to achieve the maximum observable reward as estimated by the retina as a whole.

For a given pre-saccade retina state s , we compute both the current action estimate a and the reward estimate r_a . γ is then the ratio of r_a to r_{max} , the maximum observed reward for the entire retina. Intuitively, if r_a is close to r_{max} then the action a is likely close to optimal, and so little exploration is necessary. Similarly, if r_a is less than r_{max} , the action a is likely suboptimal, and so more exploration is required. The actual action taken is

$$\gamma a + (1 - \gamma) a_{exp}$$

where a_{exp} is a random saccade.

We use a large negative change in the moving average of all the rewards as an indicator of a major change to the retina motor or sensor map (Figure 5). When detecting

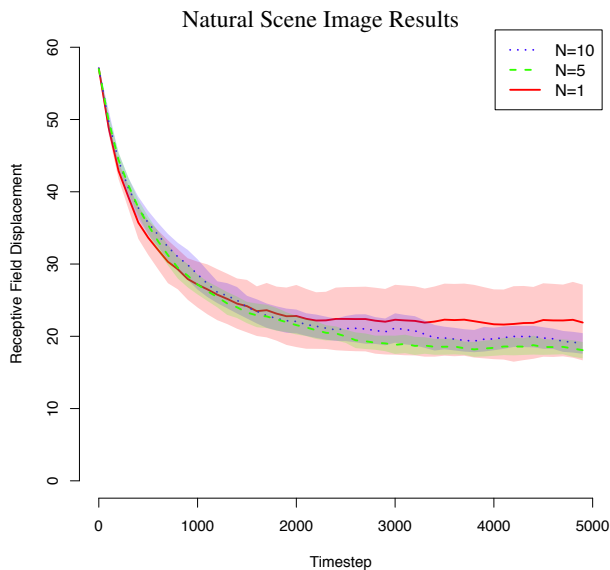


Figure 6: For this experiment, subsets of natural scene images were chosen randomly. This graph shows the mean and variance of ten runs for each subset size and is best viewed in color. Training across sets of images results in more consistent learning curves than training over single images, since the variance is smaller for training that takes place across subsets. Even in the single image case (where each run drew training examples from a single image) the mean learning curve was qualitatively similar to the others, but the high variance suggests that some images are “bad” sources of training examples.

this kind of change, the retina resets the reward estimates of all the receptive fields to their original values. This significantly decreases γ , triggering an increase in exploration and decreasing the contribution of the previously learned policy.

5.4 Natural Scenes

To recapture the features of the single spot case in natural scenes, we construct a *proto-saliency* map from natural scenes by first blurring the image under the retina using a Gaussian blur with a 5×5 filter size², then thresholding the image and taking pixels that fall into the top one percent brightness level in the region under the retina. If the number of active pixels is less than 500 pixels, we proceed to train on that portion of the image, otherwise the agent performs a new random saccade without training. This is to avoid training in situations of homogeneous brightness that wash out any existing progress on learning the optimal policy.

We note that humans tend to avoid saccading to areas of high luminance at low spatial scales (e.g. sky, solid colors) (Tatler et al., 2005). By avoiding training when the number of active pixels after thresholding is too high, we avoid training on precisely these kinds of high-

²Blurring is incompatible with the assumption that geometric information is not available. However, this blurring step is meant to simulate the optical characteristics of infants during early development (Slater, 1999), not infant visual processing.

luminance inputs.

Due to the variation in learning performance across images, we examine how the learning process behaves when trained over subsets of images randomly chosen from the Berkeley segmentation dataset (Martin et al., 2001). For each run, we select a set of images ($N=1, 5$ or 10) to train over. We cycle through the images, training 19 times over each image before moving to the next image in the cycle to continue training. As before, we evaluate the learning performance by measuring geometric errors every 100 timesteps of training. The results are shown in Figure 6.

Even though the final error rates are higher than when trained with the synthetic scene (Section 5.1), we note that the fixed point behavior of the policy (allowing repeated corrective saccades) does result in accuracy comparable to what training achieves on an ideal version of a saliency map after a similar number of training steps. The following table shows the accuracy after one and two saccades, as well as after the number needed to reach a fixed point (or in rare cases, a cycle – in which case the closest cycle point is counted).

1 Saccade	2 Saccades	Fixed Point
20.4	12.5	7.6

5.5 Pan/Tilt Camera

For the physical pan/tilt experimental setup, we used a Logitech QuickCam Orbit AF placed 15 feet from a single light source. To reduce training time, we modified the exploration policy to search randomly for a bright light. The agent performs a random saccade away from the light source. During training the agent then performs the opposite saccade back towards the light source, and uses the resulting retinal activations to learn a function from field activation to optimal saccades using the algorithm described in Section 4 with the proto-saliency method as described in Section 5.4. Unlike a learned policy, this open-loop training policy cannot account for relocation of the salient light source.

Figure 7 shows the decrease in saccade error and the increase in post-saccade reward (or activation) after intervals of 100 training steps. Each data point is the mean of 10 test trials. Each trial randomly saccades away from the light source, then computes the return saccade as the activation weighted average of the learned receptive field policies. For a trained retina, the post-saccade reward is independent of the initial random saccade, since the state of highest reward is reachable from any random starting position.

In our simulation experiments, the learned policies correspond to ground truth pixel geometry, since actions for the simulated roving eye camera are pixel unit translations over an image. The action space of the pan/tilt camera, however, is not represented in pixel unit shifts. The motor commands represent control signals sent directly to the piezoelectric motors in the camera appara-

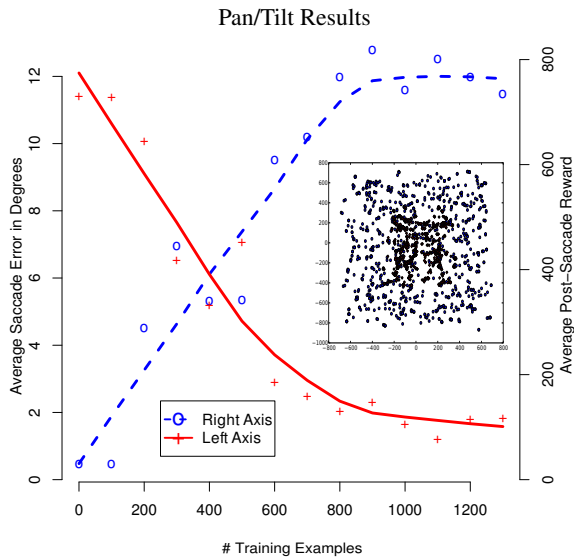


Figure 7: Every 100 training timesteps, we perform 10 test trials with the pan/tilt camera, randomly saccading away from the light source, then using the learned saccade policy to attempt to recenter on the light source (as opposed to using the inverse of the random saccade as in training). As training progresses, each receptive field learns a policy that centers local activation at the fovea resulting in greater post-saccade reward (dashed line) and lower saccade error (solid line). The subfigure shows the corresponding action space coordinates of each receptive field for two different layers of receptive fields after training.

tus. Camera geometry, along with irregularities in camera control, make the correspondence between motor signals and pixel shifts in the field of view necessarily inexact. We made no attempts to improve the correspondence through any alternative method of system identification beyond running our algorithm.

As a result of the learning process, for each region of interest we have access to the motor coordinates that center the camera on the region of interest. The geometry of these action space coordinates approximates (up to a scale factor) the ground truth geometry of the receptive fields in pixels.

Our approach is not limited to finding a sensor map in the coordinate system of the action space. With access to the ground truth pixel geometry for each receptive field, we can also construct a map from ground truth pixel coordinates to the corresponding action space coordinates, providing the ability to switch between pixel and motor geometry as a method of controlling the pan tilt camera. Selecting pixel coordinates (and activating the corresponding receptive fields) for a region of interest is sufficient to generate the corresponding motor mapping that brings those pixels to the center of the field of view. In other words, the learning algorithm autonomously provides a method for going from pan/tilt (or joystick) control, to point and click control in the view frame.

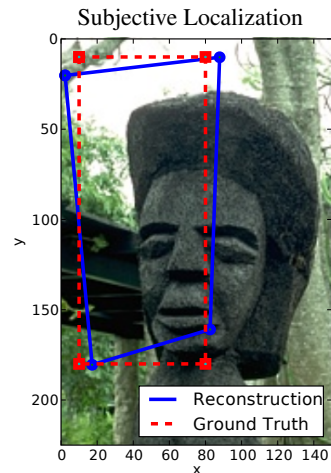


Figure 8: The results of localization in a roving eye domain. A roving eye was able to use features and their associated policies learned through sensorimotor embedding to reconstruct a visual path.

6. Future Directions

Sensorimotor embedding can be applied to other types of structure discovery problems. As an example, an agent can use sensorimotor embedding to visually localize by associating sensor inputs with ballistic actions that bring about desired changes in sensor state. This provides an alternative to *action respecting embedding* (Bowling et al., 2007) in continuous action spaces.

We applied sensorimotor embedding to the “roving eye” domain by first generating a set of 50 principle component basis vectors using random samples of a scene. We then formed a feature set consisting of principle projections of random samples onto these principle components. Associated with each feature is a reward and ballistic policy estimate just like the receptive fields described above.

During training, the projection of each eye image is compared to each feature. The winning feature determines the next (noisy) action. After each action, the reward is the least of the inverse of the distance to a predefined point in the scene or one. Updates to reward and policy estimates are the same as in Section 4. Once trained, a sequence of images can be embedded directly in the learned motor space by comparing each images projection with the feature set. An example embedding for a visual path of a roving eye is shown in Figure 8.

7. Discussion

Our experimental results confirm that, under simple assumptions, an agent can simultaneously discover motor and sensor maps for a foveated retina. Like Weber and Triesch, we use total activation as a reward signal to learn the motor map; however, we demonstrate the ability to learn without prior knowledge of the sensor map. To do so, we generate a proto-saliency map directly from natural scenes in a geometry-free way. After learning

the motor map, we generate the sensor map by exploiting the relationship between sensor geometry and motor commands. Previous approaches to sensor map construction use dimensionality reduction techniques and do not exploit additional available domain structure, namely access to motor commands.

Representing the sensor map in motor units may appear to be a limitation of the approach. However, in the absence of some external system identification, we would expect that a developmental agent would have difficulty discovering sensor geometry in units *other* than those which correspond in some way to motor semantics.

Our method is appropriate for cases with an easily identifiable reward signal (e.g. activation), linear ballistic motor commands, and a high number of sense elements. We exploit the structure of the sensorimotor domain to produce an explicit mapping between motor commands and sensor features. This map has two interpretations, one as a primitive behavior that maximizes reward (the policy or motor map interpretation), and another as a structure for the sensor array (the geometry or sensor map interpretation).

The *sensorimotor embedding* algorithm we present above, and the general approach of utilizing action spaces to better understand sensor spaces represents a fundamental first step in building a computational model of vision that follows the “seeing is acting” paradigm (O’Regan and Noë, 2001).

Any developmental process or autonomous robot depends on robust sensorimotor primitives that can adapt to changes over time. We demonstrate the robustness of our learning process under both lesioning and motor map reversal. We believe that focusing on associating structure with motor commands that bring about desirable changes in perceptual state, as in foveated retina and localization, will result in precisely the kind of robust sensorimotor primitives required for autonomous developmental robots.

Acknowledgments

This work has taken place in the Intelligent Robotics Lab at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Intelligent Robotics lab is supported in part by grants from the Texas Advanced Research Program (3658-0170-2007), from the National Science Foundation (IIS-0413257, IIS-0713150, and IIS-0750011), and from the National Institutes of Health (EY016089).

References

Bowling, M., Wilkinson, D., Ghodsi, A., and Milstein, A. (2007). Subjective localization with action respecting embedding. *Robotics Research*, 28:190–202.

Dolezal, H. (1982). *Living in a World Transformed: Per-*

ceptual and Performatory Adaptation to Visual Distortion. Academic Press.

Itti, L. and Koch, C. (2001). Computational modelling of visual attention. *Nature Reviews Neuroscience*, 2(3):194–203.

Martin, D., Fowlkes, C., Tal, D., and Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int’l Conf. Computer Vision*, volume 2, pages 416–423.

Noto, C. and Robinson, F. (2001). Visual error is the stimulus for saccade gain adaptation. *Cognitive Brain Research*, 12(2):301–305.

Olsson, L. A., Nehaniv, C. L., and Polani, D. (2006). From unknown sensors and actuators to actions grounded in sensorimotor perceptions. *Connection Science*, 18(2):121–144.

O’Regan, J. and Noë, A. (2001). A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24(05):939–973.

Pagel, M., Maël, E., and von der Malsburg, C. (1998). Self calibration of the fixation movement of a stereo camera head. *Autonomous Robots*, 5(3):355–367.

Palmer, S. (1999). *Vision science: photons to phenomenology*. MIT Press, Cambridge, MA.

Pierce, D. and Kuipers, B. J. (1997). Map learning with uninterpreted sensors and effectors. *Artificial Intelligence*, 92(1-2):169–227.

Rao, R. P. N. and Ballard, D. H. (1995). Learning saccadic eye movements using multiscale spatial filters. *Advances in Neural Information Processing Systems*, 7:893–900.

Schwartz, E. (1977). Spatial mapping in the primate sensory projection: Analytic structure and relevance to perception. *Biological Cybernetics*, 25(4):181–194.

Shibata, T., Vijayakumar, S., Conradt, J., and Schaal, S. (2001). Biomimetic oculomotor control. *Adaptive Behavior*, 9(3-4):189–208.

Slater, A. (1999). *Perceptual Development: Visual, Auditory and Speech Perception in Infancy*. Psychology Press.

Tatler, B., Baddeley, R., and Gilchrist, I. (2005). Visual correlates of fixation selection: effects of scale and time. *Vision Research*, 45(5):643–659.

Weber, C. and Triesch, J. (2006). A possible representation of reward in the learning of saccades. *Proceedings of the Sixth International Workshop on Epigenetic Robotics*, pages 153–60.

Bottom-up social development through reproducing contingency with sensorimotor clustering

Hidenobu Sumioka¹ Yuji Takeuchi¹ Yuichiro Yoshikawa²
Minoru Asada^{1,2}

¹ Graduate School of Eng., Osaka Univ.

² JST ERATO Asada Synergistic Intelligence Project

2-1 Yamadaoka, Suita Osaka, 565-0871 Japan

{hidenobu.sumioka,yuji.takeuchi,asada}@ams.eng.osaka-u.ac.jp
yoshikawa@jeap.org

Abstract

This paper presents a learning mechanism that finds a reasonable segmentation to achieve social behavior as well as that incrementally acquires it by reproducing the contingency in interactions with a caregiver. The robot autonomously categorizes sensorimotor activity according to a contingency measure based on transfer entropy. The advantage of adaptive categorization is tested in a task of acquiring joint attention behaviors. The results of computer simulations of human-robot interaction indicate that a robot acquires a series of joint attention behaviors such as gaze following and alternation and finds suitable segmentation over time that improves gaze following performance.

1. Introduction

Human infants acquire a variety of social behaviors through interaction with others. In particular, joint visual attention is one of the building blocks for such social capabilities as language communication and mind-reading (Moore and Dunham, 1995). Understanding how infants acquire a variety of joint attention behaviors such as gaze following, gaze alternation, i.e., successive looking between a caregiver and an object, and pointing is a central topic in developmental psychology. However, how infants acquire such behaviors remains a mystery.

Recently in robotics, joint attention studies have been receiving increased attention not only from the viewpoint of building communicative robots (Imai et al., 2001) but also from synthetic approaches for modeling and understanding human developmental processes (Nagai et al., 2003, Triesch et al., 2006), as argued in surveys (Kaplan and Hafner, 2004, Asada et al., 2009). Sumioka *et al* addressed how

a robot can acquire different joint attention behaviors (Sumioka et al., 2008) by emphasizing a statistical structure that reflects that infants can often attain consistent consequences when they respond adequately to a preceding stimulus that includes the behavior of their caregivers. Such a structure of the relationship (called contingency) among a preceding stimulus, one's own action, and its consequence was utilized to find more contingent sets including sensory and motor variables that provide consistent consequences to a robot among several candidates, and to construct sensorimotor maps based on the found sets. The results of computer simulations of human-robot interaction indicated that finding the contingency and reproducing it enable a robot to acquire a series of joint attention behaviors such as gaze following and alternation in an order that is almost identical to infant development.

In their study (Sumioka et al., 2008), each random variable was quantized in advance into sufficiently reasonable segments to reproduce the contingencies of interaction between a robot and a caregiver for the robot to acquire social behavior. However, it is not trivial for a robot to adequately quantize a variable since the most reasonable segmentation depends on the robot's sensor resolution, the control resolutions of the caregiver's and the robot's behaviors, and the observed object size and its location. Once the robot found the contingency based on rough segmentation, it is expected that it can find stronger contingency if it has more sophisticated segmentation. Therefore, we utilize a contingency measure to obtain more reasonable segmentation of variables by re-quantizing them so that a robot can find stronger contingency. We hypothesize that quantizing variables to experience more contingent consequences enable the robot to acquire social behavior to satisfactorily interact with its caregiver.

This paper presents a learning mechanism that adaptively quantizes each variable, finds the contin-

gency, and reproduces the contingent relationship. The contingency measure based on information theory (Sumioka et al., 2008) is utilized for adaptive quantizing whose advantage is tested in a task of acquiring joint attention behavior. The results of the computer simulations of human-robot interaction indicate that a robot acquired a series of joint attention behaviors such as gaze following and alternation and found suitable segmentation that improved over time the gaze following performance.

2. Face-to-face interaction to develop joint attention behavior

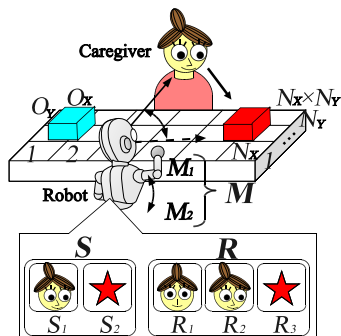


Figure 1: Environmental setting

To examine whether a robot can acquire a variety of joint attention behaviors by quantizing sensory and motor variables, we start with a rough model of a caregiver's gaze shift and simulate almost the same interaction as in previous studies (Sumioka et al., 2008).

Figure 1 shows the interaction's environmental setting. The robot sits across from the caregiver at a fixed distance. An interaction where both of the caregiver and the robot successively take an action is defined as a time step. A table has $N_X \times N_Y$ sections where two identical objects are placed randomly, each of which occupies $O_X \times O_Y$ sections. The positions of the objects are determined randomly every ten steps.

In an interaction, the caregiver observes her environment and shifts her gaze to the robot or an object based on a few policies described in section 4.1.2. Next, the robot observes its environment and obtains information about the direction of the caregiver's face (S_1) and the object's presence (S_2) as sensory variables. It stores the information about what it is looking at as the result of its actions that are called resultant sensory variables: caregiver's frontal face (R_1), caregiver's profile (R_2), and the presence of an object (R_3). Finally, it shifts its gaze to the caregiver or a table section, makes a hand gesture, and stores motor commands for gaze shift (M_1) and one for hand gesture (M_2) as motor variables.

Here, a contingency inherent in the interaction appears as a dependency of state transition of a resultant sensory variable on sensory and motor variables. We call a triplet of variables (S_i, M_j, R_k) an event variable. Moreover, an event variable that involves strong dependency is called a contingent event variable. The robot's task is performed by finding a contingent event variable and acquiring a sensorimotor mapping based on the found event variable. Moreover, the robot has to determine how it should quantize the sensory and motor variables.

3. Proposed mechanism to successfully develop social behavior with adaptive partitioning

Instead of designers who quantize a random variable into several segments in advance, our robot quantizes them autonomously. A contingency measure proposed by Sumioka *et al.* (Sumioka et al., 2008) is utilized for quantizing variables and constructing sensorimotor maps. The proposed architecture shown in Figure 2 consists of three features: (1) a contingency monitor that sends commands to quantize sensory and motor variables, (2) a state/motor categorizer to output one of the components in the sensory or motor variable based on the observed features or the motor commands, and (3) a sequential contingency learning module that enables the robot to acquire several actions by finding the interaction's contingency and its reproduction.

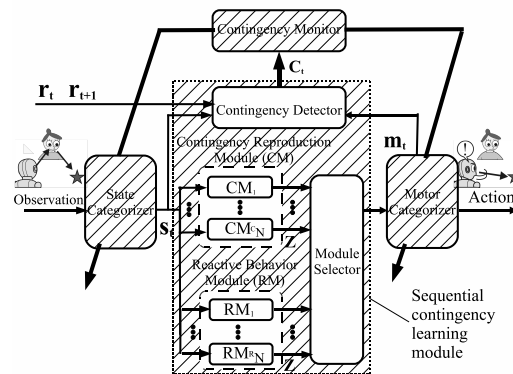


Figure 2: Proposed mechanism

Observed sensory features are organized by the state categorizer as one of the components in each sensory variable. The selected components are sent to the sequential contingency learning module, which decides one component in each motor variable based on the acquired sensorimotor mapping or the innate behavior policies described in Section 3.2.3. Finally, the motor categorizer selects the motor commands based on the selected components. During this process, an event variable's contingency

is evaluated in the sequential contingency learning module by calculating the contingency measure (Sumioka et al., 2008), as described in the next section. According to the measure, the contingency monitor commands the state and motor categorizers to update the segmentation in the sensory and motor variables.

3.1 Contingency measure

Based on transfer entropy (Schreiber, 2000), Sumioka *et al.* proposed an information theoretic measure of contingency called saliency of contingency (C-saliency) to quantify the contingency of an event variable (Sumioka et al., 2008).

Suppose that time series variables X and Y may be approximated by first-order Markov processes and that they form the following contingency: x^{t+1} , the value of X at time $t + 1$ is only influenced by x^t and y^t , i.e., the values of X and Y at previous time t . Here, the transfer entropy that indicates the influence of Y on X is defined by

$$T_{Y \rightarrow X} = \sum p(x^{t+1}, x^t, y^t) \log \frac{p(x^{t+1}|x^t, y^t)}{p(x^{t+1}|x^t)}. \quad (1)$$

C-saliency $C_{i,k}^j$, which quantifies the joint effect of sensory variable S_i and motor variable M_j on resultant sensory variable R_k , is defined as

$$\begin{aligned} C_{i,k}^j &= T_{(S_i, M_j) \rightarrow R_k} - (T_{S_i \rightarrow R_k} + T_{M_j \rightarrow R_k}) \\ &= \sum_{s_i^t, r_k^t} p(r_k^t, s_i^t) \sum_{r_k^{t+1}, m_j^t} e(r_k^{t+1}, m_j^t | r_k^t, s_i^t), \end{aligned} \quad (2)$$

where $e(r_k^{t+1}, m_j^t | r_k^t, s_i^t)$ is called an element of C-saliency under a pair of observed values (r_k^t, s_i^t) and is given by:

$$\begin{aligned} e(r_k^{t+1}, m_j^t | r_k^t, s_i^t) &= \\ & p(r_k^{t+1}, m_j^t | r_k^t, s_i^t) \log \frac{p(r_k^{t+1} | r_k^t, s_i^t, m_j^t)}{p(r_k^{t+1} | r_k^t, s_i^t)} \\ & p(r_k^{t+1}, m_j^t | r_k^t) \log \frac{p(r_k^{t+1} | r_k^t, m_j^t)}{p(r_k^{t+1} | r_k^t)}. \end{aligned} \quad (3)$$

The element of C-saliency represents the strength of the state transition's dependency from r_k^t to r_k^{t+1} on pair (s_i^t, m_j^t) . If triplet (r_k^t, s_i^t, m_j^t) causes r_k^{t+1} , the difference between $p(r_k^{t+1} | r_k^t, s_i^t, m_j^t)$, and $p(r_k^{t+1} | r_k^t, s_i^t)$ becomes larger. The event variable with the highest C-saliency is regarded as the contingent event variable.

Additionally, C-saliency has an interesting feature for evaluating the performance of an acquired sensorimotor map. If a sensorimotor map usually causes contingent consequences, it enables a robot to predict the state transitions of a resultant sensory variable only by the states of a sensory variable. The

C-saliency related to such a map gets lower because the first term's value in Eq. (3) is reduced. Therefore, a derivative of C-saliency is useful to evaluate the acquired sensorimotor map's accuracy; if the derivative is negative, the robot has acquired a sensorimotor map to sufficiently reproduce the contingency, but the robot needs to quantize the variables related to the map if the derivative is not negative.

3.2 Components in proposed mechanism

The C-saliencies of event variables are utilized by the proposed mechanism not only to find the contingency and reproduce it but also to quantize the variables based on more reasonable segmentation. Here, the roles of the components in the mechanism are described.

3.2.1 Contingency monitor

The contingency monitor modulates the quantization of the sensory and motor variables. The quantization of a variable consists of two processes: how should it be quantized by the existing segments (arrangement process) and how many segments should it be quantized into (insertion process). In each process, we used a derivative of C-saliency. Here, the derivative of C-saliency for event variable (S_i, M_j, R_k) at t time steps is indicated as $C_{i,k}^j(t) = C_{i,k}^j(t) - C_{i,k}^j(t-1)$, where $C_{i,k}^j(t)$ indicates the C-saliency for (S_i, M_j, R_k) at t time steps.

The arrangement process is always applied for all sensory and motor variables. In this process, segments in the variables are re-quantized by the sensory or motor categorizer. How much these segments should be modulated are determined by value, C_{max}^S or C_{max}^M , which is sent from the contingency monitor. C_{max}^S and C_{max}^M for sensory variable S_i and motor variable M_j are given by $C_{max}^S = \max_{j,k} C_{i,k}^j$ and $C_{max}^M = \max_{i,k} C_{i,k}^j$, respectively. To avoid modulation when the C-saliencies are overestimated due to insufficient samples, these values are sent when the variance for the moving average of each C-saliency during T^A time steps, $\sigma_{i,k}^j$, is lower than ε_A .

After the contingency detector selects a contingent event variable and acquires actions to reproduce the contingency of the event variable described in Section 3.2.3, the contingency monitor use the insertion process that decides whether it should insert new segments into a sensory or a motor variable included in a contingent event variable. Let $C_{i,k}^j$ be C-saliency for contingent event variable (S_i, M_j, R_k) . New segments are inserted to S_i and M_j when variance $\sigma_{i,k}^j$ for the moving average of its derivative $C_{i,k}^j$ keeps

a lower value than ε_S during T^I time steps. Once new segments are inserted into the sensory and motor variables, the insertion process is not applied for those variables during T^D time steps.

3.2.2 Sensory/motor categorizer

The state/motor categorizer outputs one segment in each of the sensory or motor variables for given inputs. Suppose that variable V is quantized into N_v codebook vectors and vector ${}^v\mathbf{a}_\ell$ represents segment ${}^v c_\ell$ ($\ell = 1, 2, \dots, N_v$). When vector ${}^v\mathbf{x}$ related to V is input to the state/motor categorizer, it selects segment ${}^v c_\ell$ with probability $P(V = {}^v c_\ell)$:

$$P(V = {}^v c_\ell) = \frac{\exp\{1/(\tau_v \|{}^v\mathbf{x} - {}^v\mathbf{a}_\ell\|)\}}{\sum_{q=1}^{N_v} \exp\{1/(\tau_v \|{}^v\mathbf{x} - {}^v\mathbf{a}_q\|)\}}, \quad (4)$$

where τ_v is a positive constant.

Selected codebook vector ${}^v\mathbf{a}_\ell$ is updated based on C (which stands for C_{max}^S or C_{max}^M described in Section 3.2.1), which is related to an event variable including V :

$${}^v\mathbf{a}_\ell^{t+1} = \eta \cdot {}^v\mathbf{a}_\ell^t + \eta \cdot \nu_{x, {}^v\mathbf{a}_\ell} \cdot C \left[\begin{array}{c} {}^v\mathbf{x}^t - {}^v\mathbf{a}_\ell^t \\ \text{if } {}^v\mathbf{x}^t \in {}^v c_\ell \end{array} \right], \quad (5)$$

where, η is a learning rate. $\nu_{x, {}^v\mathbf{a}_\ell}$ is given by $\nu_{x, {}^v\mathbf{a}_\ell} = \exp\left\{-\frac{\|{}^v\mathbf{x}^t - {}^v\mathbf{a}_\ell^t\|}{\zeta}\right\}$, where ζ is a constant value. C is defined as $C = \xi \tanh(C)$, and ξ is constant.

In addition, the sensory or motor categorizer inserts new codebook vectors for a variable when the insertion process is applied to the variable by the contingency monitor. Each categorizer decides where to insert the vectors based on the policy described in Section 4.

3.2.3 Sequential contingency learning module

We used the learning module proposed by Sumioka *et al* that consists of a contingency detector to calculate C-salencies for all event variables, contingency reproduction modules (CMs) that construct sensorimotor maps to reproduce found contingency, reactive behavior modules (RMs) that output a motor command based on a fixed behavior policy, and a module selector that selects motor commands among several outputs from CMs and RMs (Sumioka *et al.*, 2008).

The sequential contingency learning module keeps acquiring different sensorimotor mappings as follows. At the beginning of learning, since there are no CMs, the module selector selects the outputs of the RMs. As interaction between a caregiver and the robot is iterated, the contingency detector finds a contingent event variable and generates a new CM that constructs a sensorimotor map to reproduce the found

contingency. Once a CM is generated, the module selector starts to select outputs from the CM and from the RMs. The robot acquires several actions by this iteration of finding contingency and its reproduction.

Note that whenever a new CM is generated, a new sensory variable S and a new motor variable M are added to their sets to indicate whether the new CM was used and is going to be used, respectively. The contingency detector also starts to evaluate new event variables including S or M . Such event variables may be selected as a next contingent event variable if the found contingency leads to novel contingency. Therefore, the robot is expected to find a series of contingent events.

The sensorimotor map in CM is modulated every 200 time steps to utilize more reasonable segmentation. Hereafter i -th CM, which is constituted for event variable (S_i, M_j, R_k) , is defined as $\Pi_i(R_k|S_i, M_j)$.

4. Experiment

We conducted computer simulations to test whether the proposed mechanism can acquire joint attention actions in different environments. We first examined whether a robot can acquire a series of joint attention behaviors such as gaze following and alternation in a simple environmental setting. The size of the objects was then changed to show that the mechanism can find the more reasonable segmentation. After that, the mechanism's performance was tested in more complex situations where a robot has to deal with high-dimensional information or a field of view as the bias inherent in humans. In all experiments, the policies for the RMs and the parameters were set so that the robot could at least find the contingency related to gaze following.

4.1 Experimental setting

4.1.1 Environment and infant model

The initial set of variables is listed in Table 1. The sensory variable for the caregiver's face is denoted by S_1 , which consists of two segments, $({}^{S_1}c_1$ and ${}^{S_1}c_2)$, and two additional components that indicate whether an infant model (hereafter a robot) is looking at her frontal face (f_r) or is not looking at the caregiver (f_ϕ), respectively. In the experiments, we fixed the number of components in the sensory variable for object S_2 . Each member of S_2 indicates whether the robot is looking at an object (o) or at something else (o_ϕ).

Resultant variables R_1 , R_2 , and R_3 are designed as binary variables that indicate whether the robot is looking at its preferred face or object (1) or not (0). The robot's gaze shift denoted by M_1 consists of two segments, $({}^{M_1}c_1, {}^{M_1}c_2)$, and additional motor

command g_c that indicates a gaze shift to the caregiver’s face. Likewise, the gesture denoted by M_2 consists of two segments ($M_2 c_1, M_2 c_2$) and h_c , which indicate it is pointing its hands at the caregiver’s face.

Two RMs are used to determine the gaze and hand movements by selecting a component of M_1 and M_2 . The RM for M_1 is designed to select either g_c with probability 0.1 or one segment with probability 0.9, and the RM for M_2 randomly selects a component of M_2 . The parameters in the proposed mechanism are set as $(T^A, T^I, T^D, \varepsilon_A, \varepsilon_I, \tau_v) = (2.0 \times 10^3, 5.0 \times 10^3, 2.0 \times 10^5, 1.0 \times 10^{-10}, 1.0 \times 10^{-12}, 2.0 \times 10^{-2})$. The joint and conditional probabilities to calculate the C-saliencies were estimated using the histograms of the values of the event variables.

Table 1: Initial variables in robot

Type	Name	Elements
S	caregiver’s face	$S_1 = \{S_1 c_1, S_1 c_2, f_r, f_\phi\}$
	object	$S_2 = \{o, o_\phi\}$
M	gaze shift	$M_1 = M_1 c_1, M_1 c_2, g_c$
	hand gesture	$M_2 = M_2 c_1, M_2 c_2, h_c$
R	frontal face of caregiver	$R_1 = \{0, 1\}$
	profile of caregiver	$R_2 = \{0, 1\}$
	object	$R_3 = \{0, 1\}$

4.1.2 Behavior rules for caregivers

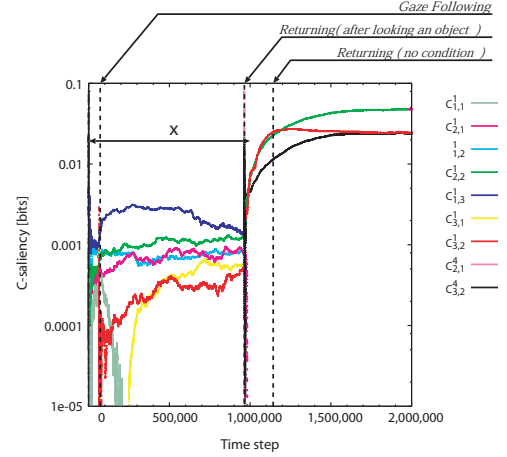
We used the caregiver model described in a previous study (Sumioka et al., 2008). The caregiver, who always looks at the robot’s face or an object on the table, not only randomly selects a target but also shows joint attention behavior.

She usually selects a target randomly. If she is looking at the robot’s face, she follows the robot’s gaze with probability p_{rja}^c . If she is looking at an object, she shifts her gaze between the robot and an object with probability p_{ija}^c . In addition, the caregiver shifts her gaze to the robot’s face with probability p_{aja}^c if she and the robot are successfully looking at the same object. In the following experiments, we used $(p_{rja}^c, p_{ija}^c, p_{aja}^c) = (0.5, 0.5, 1.0)$.

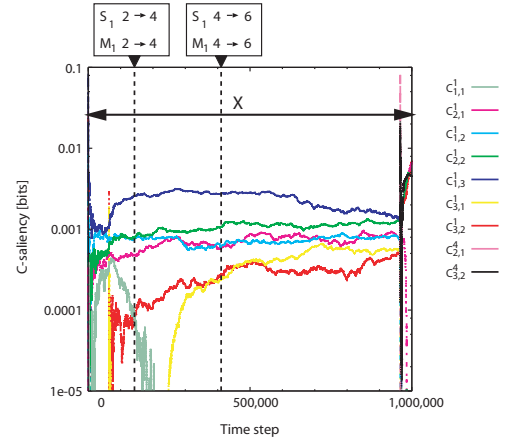
4.2 Development of joint attention with adaptive segmentation

We first confirmed that the proposed mechanism enables a robot to acquire a variety of actions related to joint attention with quantizing sensory and motor variables. We ran 2,000,000 time step simulations five times where two objects, each of which occupied 5×1 sections, were arranged on a table with 50×1 sections. We set $(\sigma, \zeta, \xi) = (0.1, 10, 6.0 \times 10^4)$. In

each simulation, two codebook vectors were added to the positions of the two existing codebook vectors selected randomly when the contingency monitor selected the insertion process for sensory and motor variables.



(a) Timing of generating CMs



(b) Timing of increasing codebook vectors

Figure 3: Time courses of saliency of contingency of event variables in simulation face-to-face interactions between caregiver and robot

An average of 2.8 CMs was obtained. In 80% of the simulations, a particular set of CMs was generated in the following fixed order: $\Pi_1(R_3|S_1, M_1)$, $\Pi_2(R_2|S_2, M_1)$, and $\Pi_3(R_2|S_3^{-1}, M_1)$. Each of these CMs allowed the robot to achieve social behavior: following the caregiver’s gaze ($\Pi_1(R_3|S_1, M_1)$; hereafter called *following-gaze* module), shifting its gaze to the caregiver after seeing an object ($\Pi_2(R_2|S_2, M_1)$; hereafter called *returning (seeing-object)* module), and shifting its gaze to the caregiver regardless of whether gaze following was achieved ($\Pi_3(R_2|S_3^{-1}, M_1)$; hereafter called *returning (no-condition)* module).

Figure 3 shows examples of the time courses of C-saliencies for nine event variables whose C-saliencies

are higher than others. The vertical axis indicates the logarithmic value of the C-saliencies. We also show the timing of generating new CMs as arrows at the top of the graph in Fig. 3(a). After sufficient interaction data were collected, $C_{1,3}^1$ became the highest among all C-saliencies (blue curve in Fig. 3(a)). As a result, a new CM ($\Pi_1(R_3|S_1, M_1)$) corresponding to the *following-gaze* module was generated, and S_3^{-1} and M_3^{-1} were added as sensory and motor variables, respectively. The robot then began to follow the caregiver's gaze with the *following-gaze* module. However, the gaze following success rate was not so high at many areas on the table (see Fig. 4(a)) because S_1 and M_1 have only two segments; that is, the robot classifies the caregiver's face looking at the table as only two different patterns. In this case, $C_{1,3}^1$ does not decrease since segmentation is not reasonable to achieve gaze following. Therefore, new segments are inserted into S_1 and M_1 (see Fig. 3(b)). Finally, the number of segments in S_1 or M_1 averaged 6.4. The codebook vectors in each variable were arranged at almost equal distance at the simulation's end (Fig. 5). The found segment arrangement enables the robot to successfully achieve gaze following with the caregiver (see Fig. 4(b)).

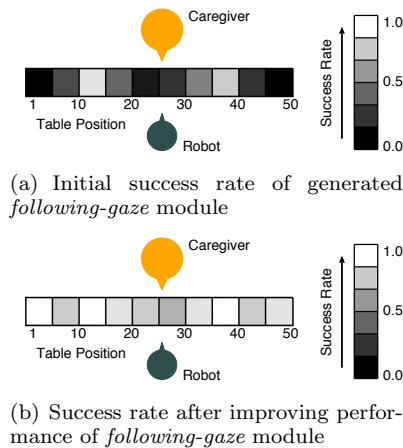
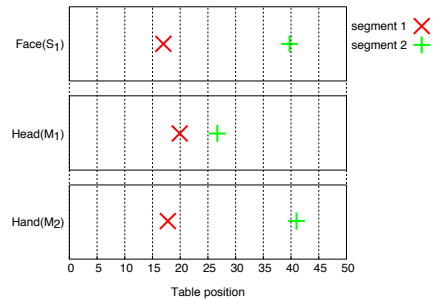
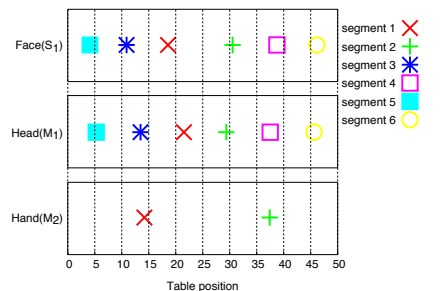


Figure 4: Changes in success rate of gaze following

The increase of the success rate of gaze following led $C_{1,3}^1$ to decrease gradually. This decrease made $C_{2,2}^1$ the next highest value, and the *returning (seeing-object)* module was generated. Using output from this module changed the contingency in the interaction and increased $C_{3,2}^1$ (red curve in Fig. 3(a)). This caused the generation of the *returning (no-condition)* module and enabled the robot to shift its gaze to the caregiver regardless whether it followed the caregiver's gaze. The robot alternately shifted its gaze between the caregiver and the object. This indicates that the robot acquired gaze following and alternation by finding a reasonable segment arrangement.



(a) Displacement of codebook vectors before learning



(b) Displacement of codebook vectors after learning

Figure 5: Transition of codebook vectors

4.3 Performance of adaptive segmentation

Such environmental features as table or object size affect how many segments are needed to achieve gaze following. We examined to what extent the robot can maintain a high gaze following performance in several different situations by arranging objects with different sizes.

We used the same experimental setting as in previous section, except for the size of objects. To show the advantage of the proposed mechanism, we also tested mechanisms without the arrangement and insertion processes; S_1 , M_1 , and M_2 were quantized into fixed segments (four, eight, or twelve segments) that were arranged at equal distance in advance.

Figure 6 shows the average success rate of gaze following in utilizing the *following-gaze* module for different object sizes. The proposed mechanism achieved a high success rate in every case, but the mechanisms without the arrangement and insertion processes had a low success rate, except when the number of segments was sufficient to achieve gaze following.

We also checked how many segments are arranged in S_1 or M_1 after learning. Figure 7 shows that the larger the object size is, the fewer segments is arranged in S_1 or M_1 . When the caregiver is looking at a large object, the robot can find the object by shifting its gaze roughly in her gaze direction. Therefore,

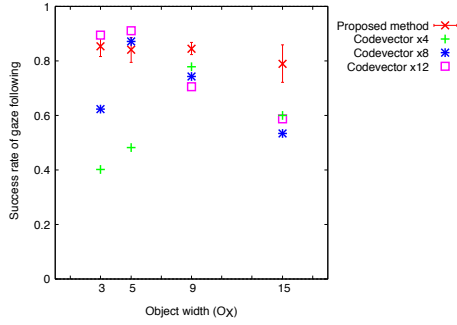


Figure 6: Change in success rate of gaze following for different object sizes

the results shown in Fig. 7 seems to indicate that the proposed mechanism found reasonable segmentation to achieve gaze following.

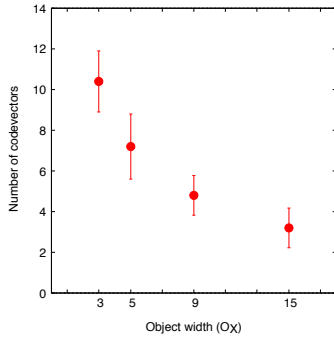


Figure 7: Number of segments in S_1 and M_1 for different object sizes after learning

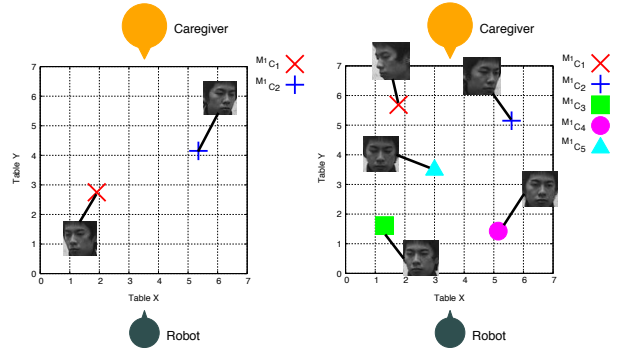
4.4 Segmentation in presence of high-dimensional information

In natural interaction with a caregiver, an infant must deal with high-dimensional information. In this case, designers have difficulty quantizing variables in advance. We tested whether a robot acquires gaze following when it obtains a camera image of a human face and takes actions on a square table.

We ran simulations where two objects, each of which is a square having four sections, were arranged on a square table with 49 sections. The robot observed one of the 18 40×40 pixel grayscale images indicating different directions of the human face. As codebook vectors for S_1 , 1600-dimensional vectors were used. The robot's actions were represented as 2-dimensional vectors indicating a position on the table. We set $(\zeta, \xi) = (1.0, 500, 8.0 \times 10^4)$. A new codebook vector was added on a point through two vectors of the existing vectors in the insertion process.

An average of 1.8 CMs was obtained. In over 80%

of the simulations, the *following-gaze* and *returning(after seeing an object)* modules were generated. In the simulations, S_1 and M_1 were quantized into an average of 5.2 segments. Fig. 8 shows the changes in the sensorimotor map from S_1 to M_1 constructed by the *following-gaze* module during a simulation. When the *following-gaze* module was generated, the robot coarsely shifted its gaze to where the caregiver was looking (Fig. 8(a)). Through the iteration of the and arrangement and insertion processes, however, it successfully acquired a sensorimotor map to follow the caregiver's gaze (Fig. 8(b)).



(a) Sensorimotor map in following-gaze module (b) Sensorimotor map in following-gaze module after simulation

Figure 8: Changes in sensorimotor map from S_1 to M_1

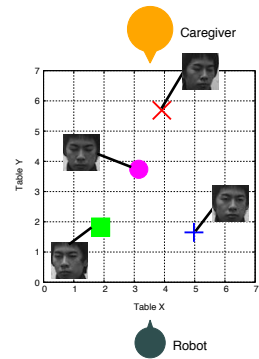


Figure 9: Sensorimotor map of following-gaze module in a robot with realistic visual field

4.5 Segmentation in presence of perspective correct visual field

In the above experiments, we assumed that the robot can see areas of fixed size despite the table position. However, this assumption is not reasonable in the real world. The size of the area seen by a human depends on the distance; the closer the area is to the human, the smaller is the area seen. Therefore, we investigated whether codebook vectors are arranged

based on the distance from a robot when it has a visual field depending on the distance.

We assumed a 0.5-meter tall robot with a 2.5-degree field of view based on the area that a human fovea covers (Fairchild, 2005). This means that the robot can simultaneously see about two or three sections on the table when shifting its gaze around the caregiver and it can see about one section when looking at the area around itself. We used the same experimental setting as in the previous section, except for the view field.

The average number of acquired CMs was 1.6. In about 60% percent of the simulations, the *following-gaze* and *returning(after seeing an object)* modules were generated. The number of segments in S_1 or M_1 averaged 3.8. Fig. 9 shows the changes in the sensorimotor map from S_1 to M_1 constructed by the *following-gaze* module after a simulation. S_1 and M_1 were quantized so that the codebook vectors were arranged by the distance. This result indicates that the proposed mechanism enabled the robot to find reasonable segmentation even when it had to segment a variable depending on the distance.

5. Conclusion and discussion

We proposed a learning mechanism that found reasonable segmentation to achieve joint attention behavior and incrementally acquired it by reproducing contingency in caregiver interactions. The robot autonomously categorized sensorimotor activity according to a contingency measure based on transfer entropy. We confirmed that a robot acquired gaze following and alternation and it found suitable segmentation to reproduce contingency in several conditions including several kinds of difficulty.

Developmental psychologists suggested that human infants gradually develop gaze following ability (Moore and Dunham, 1995); they only utilize another person's head orientation information to achieve joint attention and slowly realize that the person's eyes also direct his/her attention. In our experiment, the robot quantized sensory (and motor) variables to find stronger contingency and gradually quantized S_1 that represented the caregiver's gaze direction at higher resolution. Finding stronger contingency may explain how infants develop gaze following.

In our proposed mechanism, a derivative of C-saliency C was utilized to modulate the codebook vectors. We investigated how much C influenced this modulation. We ran another simulation using the same experimental setting as reported in Section 4.5, except that C was replaced by a constant value, $C = 1.0$. Compared to the result with adaptive C (Fig. 9), the codebook vectors of M_1 were distributed evenly on a table, although the segmentation in M_1 with adaptive C was optimized

depending on visual field. This illustrates that modulation based on the derivative of C-saliency promotes finding segmentation sufficient to reproduce contingency.

In the experiments, a few components in the variables such as f_r in S_1 were given in advance. However, a robot should quantize all variables without such a priori knowledge. The segmentation to reproduce the contingency of interaction with others may generate the components given in the experiments. As future work, we will investigate whether a robot can autonomously find suitable segmentation including f_r and f_ϕ in S_1 .

Acknowledgment

This work was supported by Grant-in-Aid for JSPS Fellows (20-5227).

References

- Asada, M., Hosoda, K., Kuniyoshi, Y., Ishiguro, H., Inui, T., Yoshikawa, Y., Ogino, M., and Yoshida, C. (2009). Cognitive developmental robotics: a survey. *IEEE Trans. on Autonomous Mental Development*, 1(1):12–34.
- Fairchild, M. (2005). *Color appearance models*. Wiley.
- Imai, M., Ono, T., and Ishiguro, H. (2001). Physical relation and expression: Joint attention for human-robot interaction. In *Proc. of 10th IEEE International Workshop on Robot and Human Communication*.
- Kaplan, F. and Hafner, V. (2004). The challenges of joint attention. *Interaction Studies*, 5:67–74.
- Moore, C. and Dunham, P., (Eds.) (1995). *Joint attention: It's origins and role in development*. Lawrence Erlbaum Associates.
- Nagai, Y., Hosoda, K., Morita, A., and Asada, M. (2003). A constructive model for the development of joint attention. *Connection Science*, 15(4):211–229.
- Schreiber, T. (2000). Measuring information transfer. *Physical review letters*, 85(2):461–464.
- Sumioka, H., Yoshikawa, Y., and Asada, M. (2008). Development of joint attention related actions based on reproducing interaction contingency. In *Proc. of the 7th Int. Conf. on Developmental and Learning*.
- Triesch, J., Teuscher, C., Deak, G., and Carlson, E. (2006). Gaze following: why (not) learn it? *Developmental Science*, 9(2):125–157.

Affordance learning from range data for multi-step planning

Emre Ugur^{1,2,3}

Erol Sahin³

Erhan Oztop^{1,2}

¹ NICT, Biological ICT Group
Kyoto, Japan

² ATR, Computational Neuroscience Labs.
Kyoto, Japan

³ METU, CENG, Kovan Lab.
Ankara, Turkey

Abstract

In this paper we present the realization of the formalism we have proposed for affordance learning and its use for planning (Sahin et al., 2007) on an anthropomorphic robotic hand. In this realization, the robot interacts with the objects in its environment using the programmed push and grasp-and-lift behaviors, and records its interactions in triples that consists of the initial percept of the object, the behavior applied and the observed effect, defined as the difference between the initial and the final percept. The interaction with the environment allows the robot to learn object affordance relations to predict the change in the percept of the object when a certain behavior is applied. These relations can then be used to develop multi-step plans using forward chaining. Our experiments have shown that the robot is able to learn the physical affordances of objects from 3D range images and use them to build symbols and relations that are used for making multi-step plans to achieve a given goal.

1. Introduction

There exists a representational gap between discrete symbols used in AI planning and the continuous sensory-motor experiences of a robot and the means to bridge this gap remains a long-standing problem in autonomous robotics. Learning of the mapping between the sensory-motor readings and these symbols is one approach that is a part of the so called symbol grounding problem (Harnad, 1990) and has been studied since the days of STRIPS. The learning studies in this context typically assume that the planning symbols are pre-coded, and only the relation of continuous sensory-motor reading to these symbols are learned (Klingspor et al., 1996).

On the other hand, (Sun, 2000) argued that symbols “are not formed in isolation” and that “they

are formed in relation to the experience of agents, through their perceptual/motor apparatuses, in their world and linked to their goals and actions”. In fact, these types of views are becoming common place in robotics as indicated by the increasing number studies with similar views. For example, symbol formation in a robot interacting with its world was studied in (Pisokas and Nehmzow, 2002), where self-organizing maps were used to cluster low-level sensory data and to form perceptual states. The planning is performed by successively predicting the next perceptual states. As will be described later on, prediction is also central to our approach although we believe that rather than learning the state-to-state transitions, learning the “change” in the current state could be more beneficial. (Geib et al., 2006) focused on planning and grounded the pre-defined high-level domain structures in the form of preconditions and effects. In our approach, we too address planning; but in addition, importantly we require that the affordance relations be learned bottom-up through interaction with the environment. The affordance notion we adopt has been adopted by other researchers as well. For instance, (Fitzpatrick et al., 2003) implemented a system where pushability affordances and the roll directions of the objects after the application of the push were learned. (Montesano et al., 2008) proposed a general probabilistic model based on Bayesian networks to learn the relationship between actions, objects, and effects through interaction with the environment. Given objects and actions (or any pair of components), the system had the ability to predict the effect (or the third component). In (Sinapov and Stoytchev, 2008) the affordances of the tools attached to the robot arm are learned by building a hierarchical models for behaviors and their observed outcomes. In (Griffith et al., 2009), the object affordances were learned through interaction for a task that requires categorization of container and non-container objects. Although these studies focused on affordance learning and prediction mech-

anisms through interaction with the environment, the use of learned/acquired knowledge has not been demonstrated for making multi-step plans.

In this paper we attempt to fill this gap by presenting a robotic system that interacts with its environment for learning the effects of its actions and representing affordance relations. After the learning phase, we show that the robot can make non-trivial multi-step plans involving push and grasp-and-lift behaviors based on the learned affordance relations. From a developmental point of view, this learning phase can be related to development of infants between 7-11 months, who explore the environment and learn the dynamics of the objects by hitting, grasping and dropping them and observing the results of their actions (Asada et al., 2009).

1.1 Affordances and Robot Control

According to Ecological Psychologist J.J. Gibson (Gibson, 1986) the organisms do not need to recognize the action-free meanings of the objects and make complex inferences over these meanings in order to act on them. For example we do not need to identify the objects when we need to interact with them. Instead, we look for a specific combination of the object properties taken with reference to us and our actions in order to detect their affordances. This introspection is also supported by neuroscientific findings. It is known that primate brain process visual information at in least two pathways: dorsal and ventral pathways. The ventral pathway appears to be responsible for object identification; whereas the dorsal pathway is more involved in perception for action (Culham and Valyear, 2006, Goodale, 2008, Goodale and Milner, 1992, Ungerleider and Mishkin, 1982). In particular, the anterior intraparietal area (AIP) appears to be the neural basis of manipulation related affordances as it is involved in computation of object features relevant for grasping (Sakata et al., 2005, Oztop et al., 2006).

Recently, we proposed a formalism (Şahin et al., 2007) for using affordances as a framework at different levels of robot control ranging from perceptual learning to planning. The formalism defines affordances as general relations that pertain to the robot-environment interaction, and represented them as triples of (1) the initial percept of the object, (2) the behavior applied and (3) the effect produced. For instance, the lift-ability affordance is represented as a relation between the (properties of an) object, the behavioral capabilities of the robot and the effects produced by the lift behavior.





In this paper we present the realization of this formalism on an anthropomorphic robotic hand and show that the robot interacts with the objects in

its environment and records its interactions as affordance relations and later use them to make multi-step plans for achieving given goals.

2. Experimental framework

An anthropomorphic robotic system, equipped with a range sensor, and its physics-based simulator is used as the experimental platform (Fig. 1). The robot platform is composed of a five fingered 16 DOF robot hand (Gifu Hand III, Dainichi Co. Ltd., Japan) and a 7 DOF robot arm (PA-10, Mitsubishi Heavy Industries). As for the range sensor, Swiss-Ranger SR-4000 infrared range finder, with 176x144 pixel array, 0.23° angular resolution and 1 cm distance accuracy was used. The simulator on the other hand is developed using Open Dynamics Engine (ODE) and mainly utilized in training and interaction phase because it is not feasible to make large number of exploratory interactions in the real robot.

The robot is equipped with three *push* behaviors and one *lift* behavior. For all behaviors, the hand is placed to a ‘reset’ position out of the view of the camera before and after behavior execution except for *lift*. The object position computed from the range finder is used as parameter by the behaviors to enable the robot interact with objects placed in different positions. The hand is wide-open initially for all behaviors, is clenched into a fist during *push-forward* execution, and remains open for other *push* behaviors. *push-forward*, *push-left*, and *push-right* behaviors first place the robot hand at the back, right and left of the object, respectively. Then, the hand moves towards object center and pushes the object in the appropriate direction. In *lift* behavior, the robot hand is placed at the back-right diagonal of the object first, then moved towards the object and while this move the fingers are closed to grasp the object. Afterwards, the closed hand is lifted vertically.

The robot interacts with three types of objects: boxes, cylinders and spheres, with different size and orientations. During the execution of *push* behaviors, the robot observes different consequences of its actions. For instance, when the robot pushes a box () or an upright cylinder (), the object is dragged during the execution of the behavior and stand still at the end of the action. However, when the robot pushes a sphere (), the object would roll away and fall down from the table, so at the end of the action the object disappears. The effect of a push behavior over lying cylinders () on the other hand depends on the relative orientation of the cylinder and direction of the *push*. The *lift* behavior would succeed in lifting an object, if the object is within the arm length of the robot and small enough to fit into the robot hand. However the consequences of the *lift* behavior execution is not limited to lifting the objects and can be complex. For exam-

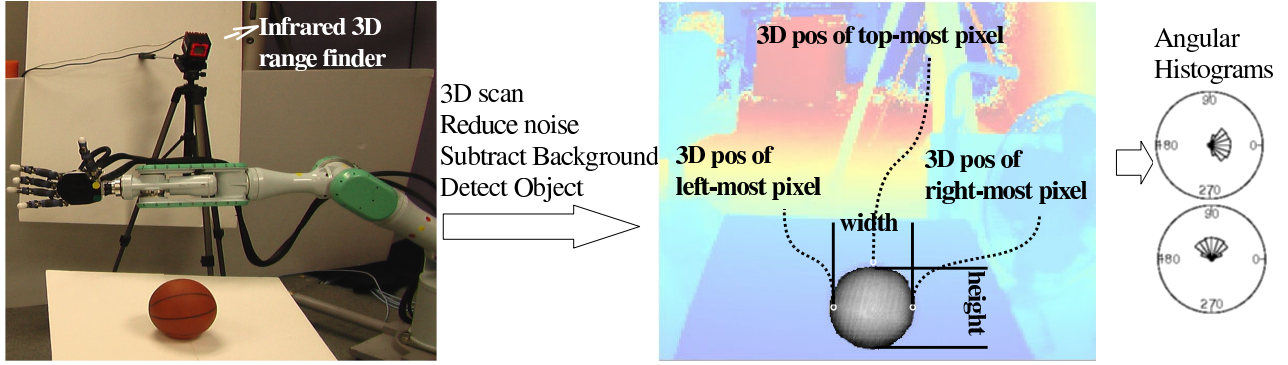


Figure 1: On the left, the 23 DOF hand-arm robotic platform, infrared range camera and a spherical object placed on the table are shown. On the right, the range image obtained from the 3D scan and a number of features computed from this image are given. Note that the subtracted background is blurred.

ple, some spheres can roll-away out of the view after an attempt to grasp and lift, and the large boxes are pushed away but remains in the view after *lift* behavior execution.

2.1 Perception

Pre-processing: The robot perceives the world through its 3D infrared range finder which provides the depth values in a range image and the 3D positions of the corresponding pixels. First, the range image is subtracted from the *background image* that was obtained from an object-free environment. The resulting image is segmented and the remained region is assumed to belong to an object. In the experiments reported in this paper only one object is presented to the robot. In order to reduce the effect of noise, the pixels at the boundary of the object are removed and then median and Gaussian filters with 5x5 window sizes are applied. Finally, the object features are computed using the depth values and 3D positions corresponding to the object pixels.

Object feature vector computation: The perception of the robot at time t is denoted as \mathbf{f}_o^t , where o is the object label and \mathbf{f} is a feature vector of size 53. Height, width and depth are used as dimension related features of the object. Closest, furthest, left-most and right-most points of the object are extracted and their 3D positions are included into the feature vector. The average distance of the pixels are used as the distance feature of the object. As shape related features, distribution of the local surface normal vectors of the object pixels are used. Specifically frequency histograms of normal vector angles in latitude and longitude are computed and used as follows.

The normal vectors of the local surfaces for all pixels are computed using any two neighbors of the corresponding pixel:

$$\mathbf{N}_r = (\mathbf{p}_{r_1} - \mathbf{p}_r) \times (\mathbf{p}_{r_2} - \mathbf{p}_r)$$

where r , r_1 and r_2 represent indexes of the pixel, and two neighbor pixels, and \mathbf{p} corresponds to 3D position. The direction of each normal vector is recorded in two base-dimensions, latitude and longitude. Two angular histograms are computed for each of these dimensions and the histograms are sliced into 18 intervals of 20° each. Frequency values of angular histograms obtained from normal vectors of the surface points in the region are used as 36 shape-related features. This representation encodes the distribution of the local surface normal vectors of the object.

In some situations (after execution of some behaviors) the object can move out of view. So we included a boolean *object visibility* feature in the feature vector to represent this qualitatively different situation.

Effect feature vector computation: For each object, the effect created by a behavior is computed as the difference between the final and initial features:

$$\xi_o^{b_i} = \mathbf{f}'_o - \mathbf{f}_o$$

where $\xi_o^{b_i}$, \mathbf{f}'_o and \mathbf{f}_o represents the effect, final and initial feature vectors, and b_i represents the behavior executed.

3. Learning of affordance relations

During the interaction phase, the robot interacts with the environment and in each interaction a *relation instance* of the form $(\xi_o^{b_i}, \mathbf{f}_o, b_i)$ is created. After interaction phase is completed, by using effect instances $(\{\xi_o^{b_i}\})$ that are obtained from all different objects, similar effects are grouped together to get a more general description of the effects that each behavior can create. This grouping is done using X-means clustering algorithm and for each cluster an *effect-id* is assigned. The associated *effect prototype* for the cluster is defined as the mean of the cluster, denoted as $\bar{\xi}^{b_i}$. Hence, for a given ξ , the correspond-

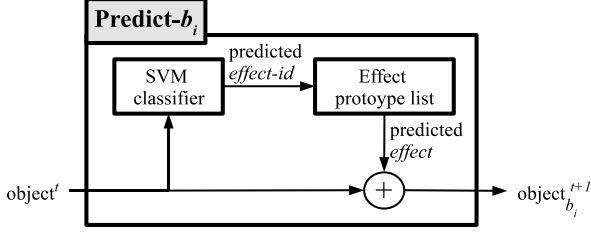


Figure 2: The **Predict-** operator that is trained to predict the next state of an object based on the predicted effect of applying behavior b_i .

ing $effect-id$ (c) can be found as:

$$c_{\xi} = \underset{1 \leq i \leq k}{\operatorname{argmin}} \|\xi - \bar{\xi}_i\| \quad (1)$$

where $1 \leq i \leq k$ is the cluster index.

Formally, the mapping between the object features and the effects created by a particular behavior b_i is learned by a classifier (χ^{b_i}) using the data set:

$$\mathbf{T}^{b_i} = \{(\mathbf{f}_o, c_{\xi_o^{b_i}})\}$$

where \mathbf{f}_o is given as the input feature vector to the classifier χ^{b_i} , and c is the corresponding target effect category. Specifically, we used a Support Vector Machine (SVM) classifier with linear kernel to learn this mapping for each behavior b_i using this training set¹. After training, predicted effect category can be found without applying behavior b_i to an object with perceptual features \mathbf{f}_o by:

$$c^{\text{predicted}}(b_i, o) = \chi^{b_i}(\mathbf{f}_o)$$

The predicted percept of the object after the application of the behavior can then be computed as (see Fig. 2):

$$\mathbf{f}'_o(\{b_i\}) = \mathbf{f}_o + \bar{\xi}_{c^{\text{predicted}}(b_i, o)}^{b_i}$$

4. Planning

The learned affordance relations can be used as operators for planning.

States A state is represented with the object feature vector that is perceived or expected to be perceived after execution a number of behaviors in t steps:

$$S_{\{b_i^1 \dots b_i^{t-1}\}}^t = \mathbf{f}_{o, \{b_i^1 \dots b_i^{t-1}\}}^t$$

where o corresponds to the perceived object, and $\mathbf{f}_{o, \{b_i^1 \dots b_i^{t-1}\}}^t$ is the expected percept after execution of the behavior sequence $\{b_i^1 \dots b_i^{t-1}\}$.

¹For this study, the LibSVM software is used. In (Uğur et al., 2007) we showed that the method is not constrained to batch learning and the training can be done in an online manner. The number of samples were minimized in online version by selecting the most interesting situations for interaction instead of random exploration.

Actions The pre-coded behaviors; namely the three *push* behaviors and the *lift* behavior, constitute the actions. Different from standard techniques, the actions do not have any pre-conditions and their description does not include pre-defined state transition rules. All actions are applicable in all states, where the next state depends on the learned effect prediction operators summarized in Fig. 2.

$$S_{\{b_i^1 \dots b_j^{t-1}\}}^t \xrightarrow{b_k^t} S_{\{b_i^1 \dots b_k^t\}}^{t+1}$$

Goals A goal is specified as a partial state, in terms of values of some object features within states. The user can define a goal based on feature values of the object. For example, the state that includes an object feature vector with $d_{mean} = 0.8m$ will satisfy the goal of *move object to 0.8m distance*. As another example, the goal of *pick-up a particular object* is satisfied in a state, where the closest-point z feature value of the corresponding object is large ($z_c > 0.3m$) in the range image.

Plan generation Forward chaining is used to generate totally ordered plans starting from the initial state. This process can be viewed as the breadth-first construction of a plan tree where the branching factor is the number of behaviors. The next states are computed using the prediction operator in Fig. 2. If the state in any time step satisfies the goal, the sequence of the behaviors which lead the initial state to the goal is accepted as a potential plan.

5. Experiments

The learning experiments are conducted in the physics based simulator and the results are tested in the real robot. As shown in Fig. 1, a table is placed in front of the robot both in simulation and real world. In the beginning of interactions, one random object o (among \square , \ominus , \boxplus , \square) is placed on the table, in a random orientation and size [20cm – 40cm]. The robot makes 3D scans before and after executing one of its behaviors (b_i) to compute the object (\mathbf{f}_o) and effect ($\xi_o^{b_i}$) feature vectors. After the behavior is applied, if the object is still visible and if there is change in the object features, another random behavior is applied. Otherwise, the object should have been fallen down the table or it is out of reach of the arm, so the object is removed and a new random object is placed. For all behaviors, approximately 1000 interactions are simulated. The resulting set of relation instances are then used in training.

5.1 Discovered effect categories for push

After robot-environment interactions are completed and affordance relations instances are collected, X-means algorithm found 5 effect categories for each

Table 1: EC_i^p represents i^{th} effect category of *push-forward* behavior. In (a), selected feature values of the corresponding effect prototypes are given. The magnitude of the arrow corresponds to the size of the change in feature value, whereas whether the feature is increased or decreased can be figured out by arrow's direction. In (b), in which situations such effect categories are formed is explained. The number of the object types appear during interactions are given in the first four column, the average real width and distance of the objects in those interactions are given in the last two columns.

	Visibility	Width	X Pos	Y Pos	Z Pos		⊖	⊞	⊠	⊡	Width	Dist
EC_1^p	-	-	-	-	↑	EC_1^p	0	45	40	65	17.0	86.3
EC_2^p	-	-	↓	-	↑	EC_2^p	0	0	0	55	23.1	90.8
EC_3^p	-	-	-	-	↑	EC_3^p	0	85	15	20	17.3	94.1
EC_4^p	↓	-	-	-	-	EC_4^p	125	10	175	15	17.5	88.8
EC_5^p	-	-	-	-	-	EC_5^p	50	90	170	40	18.1	124.4

(a) Effect features (b) Info. on objects

push behavior. In this section, the effect categories are interpreted by inspecting particular feature values of the corresponding effect prototypes and by identifying the situations in which these categories are generated. In all three *push* behaviors, similar categories are formed so here we present only *push-forward* behavior. Table 1(a) gives selected feature values from effect prototypes formed for *push-forward*, $\bar{\xi}_i^{\text{push-forward}}$ where $1 \leq i \leq 5$. In other words, the amount of change in different features is provided. Table 1(b) summarizes the situations in which effect categories are generated. The interpretation of effect categories is as follows:

- As seen in the Table 1(a), no effect in any feature is monitored in EC_5^p . When initial average distance of the objects is examined for EC_5^p from Table 1(b), it is found to be around 124.4 cm which corresponds to out of the range points. So the robot discovers interaction range of its *push* behaviors and represent it in EC_5^p .
- Visibility of objects drop from 1 to 0 in EC_4^p , which means the objects fall off the table in these situations. This can happen when the objects are pushed and rolled away out of the table. Indeed when the types of the object are inspected in Table 1(b), it is seen that significant number of spheres and lying cylinders create this effect category. Small number of boxes (10) and up-

Table 2: EC_i^l represents i^{th} effect category of *lift* behavior. For further explanation please see Table 1.

	⊖	⊞	⊠	⊡	Width	Dist
EC_1^l	60	10	85	0	16.7	90.0
EC_2^l	0	30	5	45	11.5	88.7
EC_3^l	55	75	160	105	17.8	122.4
EC_4^l	0	90	5	40	20.6	94.3
EC_5^l	65	25	95	50	20.8	88.7

(a) Effect features (b) Info. on objects

right cylinders (15) are also included in the EC_4^p because either they fall down from the edge of the table when pushed or they were at the robot hand as the result of previous *lift* behavior. When *push-forward* is activated, all robot angles including fingers are set to their initial positions, the hand will be opened and the object will drop from the hand and fall down to the ground. Note that the later effect of object drop is an emergent one, ie. the release behavior is not deliberately intended by the behavior designer. This emergent effect category will play a major role during planning in Section 5.3.

- In EC_1^p , EC_2^p and EC_3^p , the Z position of the objects is increased as the result of *push-forward* behavior and at the end of interaction the objects still remain visible as seen in the Table 1(a). When the object types are inspected, it is seen that these effect categories are produced mostly by boxes and upright cylinders. Such effects are not generated when the object is a sphere because spheres always roll-away when pushed, but some lying cylinders can lead to these effects because different orientations of lying cylinders afford either rollability or pushability.

5.2 Discovered effects categories for lift

Lift behavior, although not represented differently in robot's behavioral repertoire, is conceptually different from *push* behaviors, so the effect categories for *lift* behavior are interpreted separately. The feature values of effect prototypes and the situations in which effect categories are generated are given in Table 2 and the interpretation is as follows:

- EC_3^l describes effects that do not change significantly at all and corresponds to not-reachable

objects. This effect category is similar to EC_5^p for *push-forward* behavior.

- In EC_5^l , the objects become invisible during execution of *lift* behavior. Disappearing from the view was not expected and against our intention in *lift* behavior. There are two different type of situations for existence of such an effect. First, the object may be already in robot hand before execution of *lift* behavior so the initialization step of *lift* can result in falling the object off to the ground. Second, the rollable large objects cannot be grasped by robot hand, but they will be dragged and rolled down the table. The second reason explains the existence of large number of spheres and lying cylinders in EC_5^l .
- In EC_4^l , the height of the object (Y pos) does not change, but its position with respect to the robot changes. In other words, the object is not lifted, but dragged over the table. This effect is created by large ungraspable objects. Different from EC_5^l (previous item), they are not rollable. This claim is supported by large number of boxes (90) and lying cylinders (40) in EC_4^l , and relatively large average width of the corresponding objects (20.6 cm).
- In EC_1^l and EC_2^l , the height of the objects (Y Pos) are increased, so these effect categories correspond to situations where objects were actually lifted. One significant difference between two categories is on the change of perceived width of the objects. The perceived width of the object is decreased more in EC_2^l probably because it is better covered inside hand. Based on the actual average widths of the objects for these effect categories, smaller objects are seen to create EC_2^l . This is consistent with the explanation in decrease of perceived width because small objects are better grasped and tend to disappear inside hand.

5.3 Learning affordances and planning

After effect categories are discovered, the mapping from initial features to these categories are learned by training SVM classifiers χ^{b_i} for each behavior. 800 interactions are used in training and a separate set of 200 interactions are used in testing the classifiers. At the end, in predicting correct effect categories around 90% accuracy is obtained for different behaviors.

The planning capability, which is based on the learned affordance prediction system, is tested and demonstrated in the real robot platform. The robot, infrared range camera, and table are placed similar to the simulated interaction environment. A system with three main modules, namely *Perception*, *Planner* and *Execution Control*, are used for online verification of the approach. The *Perception Module*

is connected to the infrared range camera and is responsible for computing the object features and sending them to the *Planner Module*. Moreover, the *Perception Module* informs the *Execution Control Module* about the position of the objects for behavior parameterization. The *Execution Control Module* on the other hand receives a sequence of behaviors (a multi-step plan) from the *Planner Module* and executes the behaviors one by one using the positions received from the *Perception Module*. The *Execution Control Module* also informs the *Planner* when the plan is completed. When the plan completed signal is received, the *Planner Module* is responsible for sending a new multi-step plan to the *Execution Control* using the object features from the *Perception*. In fact, the *Planner Module* generates plans continuously but sends the plans only when plan completed signal is received. Continuous plan generation enables the *Planner* to monitor whether the execution of the original plan proceeds as planned or not. This system is tested with different objects and object placements for two different goals.

Keep the table clean The motivation of the first case study is to keep the table clean. In order to satisfy this goal, the desired value for *object-visibility* feature is set to be 0 (or false). So the planner needs to find a sequence of behaviors which leads to an object state with that particular feature value 0. The snapshots taken from this experiment are provided in Fig. 3. When a ball is placed in the middle of the table, the *Planner Module* selects *push-right* behavior and after the behavior is executed, the ball rolls away and falls off the table. When an upright cylinder is placed almost in the same place, the *Planner Module* generates a two-step plan (*lift* and *push-forward*). First *lift* behavior is executed and the object is lifted. Later, *push-forward* is activated, so the arm and hand joints need to move to their original (initial) position. During the initialization of the behavior, the hand opens and the object falls down. As we discussed earlier, this is rather an emergent behavior that was not planned by behavior designer but discovered by the learning system. In other situations when the object is placed at the edge of the table, *push-right* or *push-left* behaviors are selected and the object is pushed off the table. This experiment verifies that affordances related to physical characteristics of the objects (ball and upright cylinder). Moreover, the characteristics of the environment are also learned through interaction and system makes different plans based on different positions of the cylinder.

Bring the object to a target position The task in this case study is to bring the object to a desired position. The goal is defined over three feature values, that correspond to the 3D position of the ob-

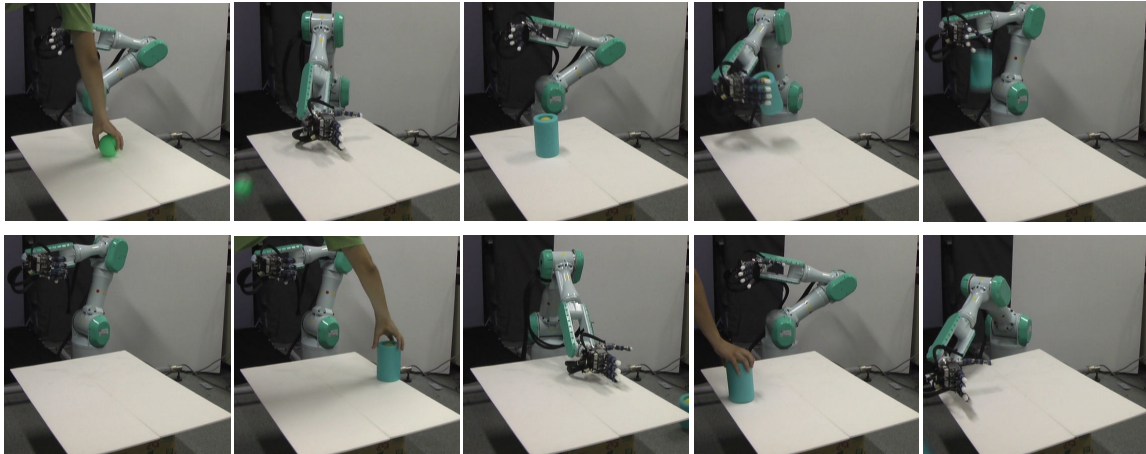


Figure 3: The table is kept clear by setting a desired state where *object-visibility* feature is 0. The corresponding movies can be downloaded from <http://www.kovan.ceng.metu.edu.tr/~emre/epirob09>.

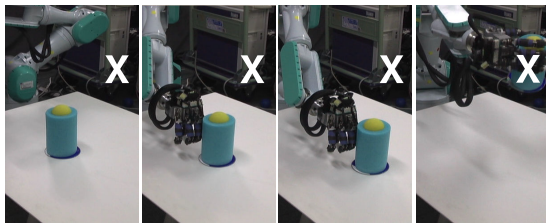


Figure 5: A plan is executed for the task of bringing the object to the target position represented by X.

ject's closest perceived pixel. In the first experiment, the target is set as a point on the table and is shown with a cross (X) in Fig. 4. The object is placed at the back and on the right side of the target from robot's view. In this case, the *Planner Module* generates a 4-step plan which is composed of *push-left*, *push-left*, *push-forward*, *push-forward*. After the plan is executed successfully, the same object is placed on the left side of the target from robot's view, closer to the robot when compared to previous case. The 4-step plan that is composed of one *push-right* and three *push-forward* behaviors is also executed successfully. In both cases, the exact order of behaviors is not important and in fact many different 4-step plans are generated with same behaviors arranged in different orders.

A more complex task description is given in Figure 5 where also desired height of the object is provided in the goal state. In this case, the *Planner* generates a 3-step plan composed of two *push-forward* and one *lift* behaviors. Different from previous case where target position is on table, only one plan with this particular order is generated because a *push-forward* behavior executed after *lift* behavior has the emergent effect of dropping the object from hand.

6. Conclusion

In this paper, we have shown that an anthropomorphic robotic hand can learn the physical affordances of objects from range images and use them to build symbols and relations that can be used in making multi-step predictions about the affordances of objects and achieve complex goals.

First, the robot is shown to discover different effect categories that represent qualitatively different set of situations in a completely unsupervised manner. Furthermore, some effects were not intended by the behavior designer but emerged during interactions. For example, although no 'release' behavior is implemented explicitly, the robot is shown to drop the object from hand during initial stage of the *push-forward* behavior if the the robot was holding the object in its hand.

The mapping between object features and effect categories are later learned by training classifiers which are used to form basic prediction operators. In two case studies, the knowledge that is acquired through learning in the simulator is directly transferred to the real robot. The robot generated multi-step plans for both table cleaning and object moving tasks and executed successfully. Because the robot formed the prediction operator based on its sensory-motor experience, it was able to make grounded and sometimes unexpected emergent plans for the same goal in different situations. These experiments verified that physical properties of the objects and characteristics of the environment are reflected in the learned affordances and generated plans. In future, the method will extended to multi-object environments and more complex non-discrete behaviors.

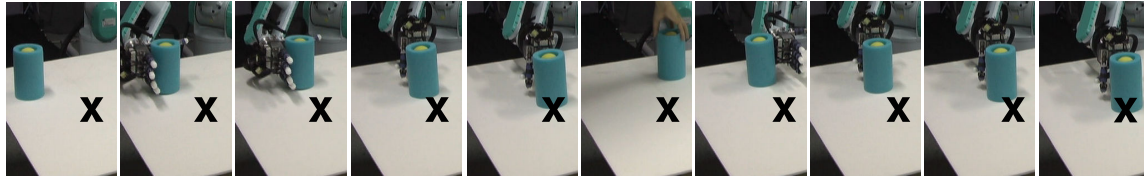


Figure 4: Different plans are executed in different situations for the task of bringing the object to the target position represented by X.

Acknowledgments

This work was partially funded by the European Commission under the ROSSI project (FP7-216125). We would like to acknowledge that the ideas presented in this paper partially shaped through discussions with Maya Çakmak and Mehmet R. Doğar.

References

- Asada, M., Hosoda, K., Kuniyoshi, Y., Ishiguro, H., Inui, T., Yoshikawa, Y., Ogino, M., and Yoshida, C. (2009). Cognitive developmental robotics: a survey. *IEEE Transactions on Autonomous Mental Development*, 1(1):12–34.
- Şahin, E., Çakmak, M., Doğar, M. R., Uğur, E., and Üçoluk, G. (2007). To afford or not to afford: A new formalization of affordances toward affordance-based robot control. *Adaptive Behavior*, 15(4):447–472.
- Culham, J. C. and Valyear, K. F. (2006). Human parietal cortex in action. *Current Opinion in Neurobiology*, 16:205–212.
- Fitzpatrick, P., Metta, G., Natale, L., Rao, A., and Sandini, G. (2003). Learning about objects through action -initial steps towards artificial cognition. In *Proc. of ICRA 03*, pages 3140–3145.
- Geib, C., Mourão, K., Petrick, R., Pugeault, N., Steedman, M., Krueger, N., and Wörgötter, F. (2006). Object action complexes as an interface for planning and robot control. In *Workshop: Towards Cognitive Humanoid Robots at IEEE RAS Int Conf. Humanoid Robots*. IEEE RAS.
- Gibson, J. (1986). *The Ecological Approach to Visual Perception*. Lawrence Erlbaum Associates.
- Goodale, M. A. (2008). Action without perception in human vision. *Cog Neuropsycho*, 25:891–919.
- Goodale, M. A. and Milner, A. D. (1992). Separate visual pathways for perception and action. *Trends Neurosci*, 15:20–25.
- Griffith, S., Sinapov, J., Miller, M., and Stoytchev, A. (2009). Toward interactive learning of object categories by a robot. In *Proc. of 8th ICDL*, Shanghai, China.
- Harnad, S. (1990). The symbol grounding problem. *Physica D*, 42(1-2):335–346.
- Klingspor, V., Morik, K., and Rieger, A. D. (1996). Learning concepts from sensor data of a mobile robot. *Machine Learning*, 23(2-3):305–332.
- Montesano, L., Lopes, M., Bernardino, A., and Santos-Victor, J. (2008). Learning object affordances: From sensory–motor maps to imitation. *IEEE Transactions on Robotics*, 24(1):15–26.
- Oztop, E., Imamizu, H., Cheng, G., and Kawato, M. (2006). A computational model of anterior intraparietal (aip) neurons. *Neurocomputing*, 69:1354–1361.
- Pisokas, J. and Nehmzow, U. (2002). Experiments in subsymbolic action planning with mobile robots. Technical report.
- Sakata, H., Tsutsui, K. I., and Taira, M. (2005). Toward an understanding of the neural processing for 3d shape perception. *Neuropsychologia*, 43:151–161.
- Sinapov, J. and Stoytchev, A. (2008). Detecting the functional similarities between tools using a hierarchical representation of outcomes. In *Proc. of 7th ICDL*.
- Sun, R. (2000). Symbol grounding: A new look at an old idea. *Philosophical Psychology*, 13(149–172).
- Ungerleider, L. G. and Mishkin, M. (1982). *Two cortical visual systems*, pages 549–586. Cambridge MA: MIT Press.
- Uğur, E., Doğar, M. R., Çakmak, M., and Şahin, E. (2007). Curiosity-driven learning of traversability affordance on a mobile robot. In *Proc. of ICDL’07*.

Posters

Using the interaction rhythm to build an internal reinforcement signal: a tool for intuitive HRI

Pierre Andry, Nicolas Garnault, Philippe Gaussier
ETIS UMR CNRS 8051, ENSEA, University Cergy Pontoise
F-95000 Cergy Pontoise, France
andry,garnault,gaussier@ensea.fr

Introduction

With the recent progress of learning by imitation, learning by doing or learning by demonstration, it now begins to be possible to teach to robots how to learn complex skills. Because of the complexity of the the skills, most of the technics use iterative or step by step procedures, allowing to refine the skill or the behavior through multiples demonstrations. In this scope, an important progress still needs to be done if we wish to obtain intuitive interactions, and confide the teaching of the robot to a non expert, or somebody that would not have to learn any explicit procedure or interface to indicate to the robot if it has succeeded or not to exhibit the right output. More fundamentally, we still have to understand what are the right mechanisms allowing the establishment of a "natural" turn taking between human and artificial systems, allowing to obtain robust interactions and an efficient support for long learning procedures. In this paper, we will show that a simple sensory-motor system detecting the rhythm of its own movements, is able to build an internal reinforcement signal that can, in turn change the weights of its own input-output associations. We will first explain our pluri-dicsiplinary motivation, and how developmental psychology highlights the importance of synchrony and rhythm detection in interactive situations. We will then show how a robot using rhythm detection can learn from a human a set of sensory-motor associations without any explicit reinforcement signal or teaching interface nor notion of agency included in the robot's architecture.

Pluridisciplinary motivation

Our work is inspired by noticeable studies in developmental psychology analyzing the young infants abilities to detect changes in the other's responses during face to face interactions. Since 1978, the Still face paradigm, introduced by Tronick et al has been widely used and studied (see (Nadel and (Eds.), 2005) for a review) especially during pre-verbal interactions. A Still face consist

in the production of a neutral, still face of the caregiver after a few minutes of interactions. Interestingly, this sudden break of the response and the timing of the interaction induce a fall of the infant's positive responses. The same responses were also measured with the more accurate Double Video paradigm allowing to shift the timing of the interaction using a dual display and recording system (Nadel et al., 2005). In this second paradigm, the content of the care giver's responses remain the same as in a normal interaction, but a more or less long decay can be introduced in the display of the mother's responses. The results highlight the importance of the timing of the response and shows how synchrony and rhythms are fundamental parameters of early interactions. Breaking the timing results in violating the infant's expectations, and produces the fall of positive responses with increasing negative ones.

Model

We are interested in studying how these changes can be detected and used to build internal values useful for learning and interaction. Our main working hypothesis is that during a face to face interaction composed of simple gestures (for example an imitation game) a constant rhythm should naturally emerge if the interaction goes well (that is to say, if the robot's responses correspond to the human's expectancies). Conversely, if the robot produces the wrong behavior or the wrong responses, we suppose that the human may introduce more breaks in the interaction, for example to take the time to restart the game, manifest his dissatisfaction (even if the robot is not able to process any signal concerning this dissatisfaction), or simply stopping the interaction. To illustrate how these hypothesis can be used, we propose a simple face to face interaction. The visual space of the robot is split in in different areas stimulated by the gestures of the human. When stimulated, one visual area triggers one gesture from a part of the robot's body. At the beginning of the experiment the connections between the Input layer (visual ar-

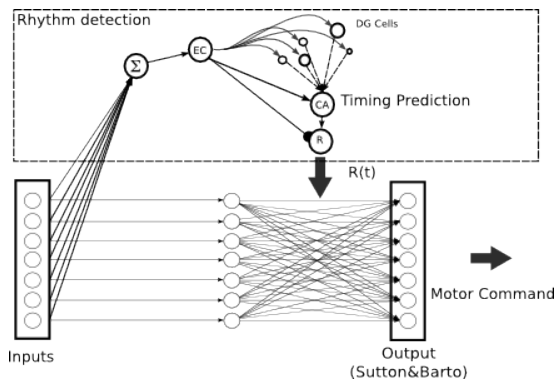


Figure 1: Model for rhythm detection and prediction. A reinforcement value is built when comparing the value of the input with the prediction activity of the rhythm

eas) and the Output layer (motor response) are random. This very elementary sensory-motor network ensures that the robot gives a response every time it is stimulated by the human : it bootstraps a simple turn taking interaction. The goal of the game is to teach the robot to perform an imitation without explicit reinforcement, that is to say to teach how to produce the motor responses that mirrors the human's gestures. During the interaction, the robot has to explore its motor repertory and reinforce the mirror actions. This reinforcement is done by a Sutton and Barto rule on the output group, allowing at the same time to explore and propose different Output actions and to strengthen the correct Input-output associations. Because no explicit reinforcement is given, the system builds from the interaction rhythm its own internal reinforcement value. Interestingly, the rhythm of the whole interaction can be extracted from the rhythm of self action. The system processes the sum of proprioceptive information from self gestures, always triggered by the human's gestures. Consequently, we can have a reliable information about the whole dynamics of the interactions by monitoring only the flow of self actions. A neural network learns the rhythm of the interaction and tries to predict the timing of the next motor action(see Fig 1). One EC neurons fires at the beginning of new actions (detects proprioception deviation). DG group decomposes the time elapsed between EC spikes, with multiple cells responding at different timing and with different time span (it simulates the response of cells of different sizes to the same EC stimulation). One CA cell learns the association between the state of the DG cells (the time elapsed since last action) and the new EC activation (the current action). After one learning, a sole activation of the DG cells produces an increasing activity of CA which tops at the instant corresponding to the predicted period of EC. At last, comparing CA and EC activities allows to obtain a continuous

prediction of the rhythm. This comparison is then used as the reward $R(t)$ for the Sutton and Barto group. We have tested this model according to two

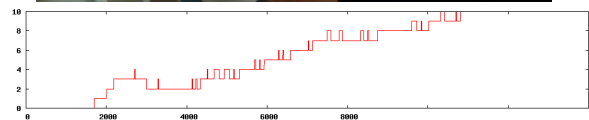


Figure 2: Up: Setup of the "mirror imitation" game with Aibo. Bottom: score corresponding to the learning of 10 Input-Output associations (that mean 100 different associations to explore). We can notice that the first association (score of 1) is the most difficult to discover. Once the first correct association is learned, it is often used as a basis for the interaction, thus allow the establishment of a constant rhythm and facilitating the discovery of further associations.

different experimental setup. A first setup was devoted to test the robustness and convergence of the of the architecture with large amounts of associations between key strokes and sounds (100 possible Input-Output associations). A second setup used a Sony Aibo robot, with the goal of discovering and learning 4 associations between the human gestures and the robot's responses (the imitation game). In future works, we are (1) extending the model to robots responses of very different temporality (how to extract the rhythm of the interaction when actions of the robot are of different lasting), and (2) comparing the convergence of different Learning algorithms (Sutton and barto *vs* probabilistic learning rule algorithms).

Acknowledgements

This work was supported by the French Region Ile de France, the Institut Universitaire de France (IUF) and the FEELIX GROWING european project.

References

- Nadel, J. and (Eds.), D. M. (2005). *Emotional Development*. Oxford University Press, Oxford.
- Nadel, J., Prepin, K., and Okanda, M. (2005). Experiencing contingency and agency : first step toward self-understanding ? *Interaction Studies*, 2:447-462.

The IM-CLeVeR Project: Intrinsically Motivated Cumulative Learning Versatile Robots

Gianluca Baldassarre¹, Marco Mirolli¹, Francesco Mannella¹, Daniele Caligiore¹, Elisabetta Visalberghi¹, Francesco Natale¹, Valentina Truppa¹, Gloria Sabbatini¹, Eugenio Guglielmelli², Flavio Keller², Domenico Campolo², Peter Redgrave³, Kevin Gurney³, Tom Stafford³, Jochen Triesch⁴, Cornelius Weber⁴, Constantin Rothkopf⁴, Ulrich Nehmzow⁵, Joan Condell⁵, Mia Siddique⁵, Mark Lee⁶, Martin Huelse⁶, Juergen Schmidhuber⁷, Faustino Gomez⁷, Alexander Foester⁷, Julian Togelius⁷, Andrew Barto^{1,8}

¹Istituto di Scienze e Tecnologie della Cognizione (ISTC), CNR

²Universita Campus Bio-Medico

³Adaptive Behavior Research Group, Department of Psychology, University of She eld

⁴Frankfurt Institute for Advanced Studies, Johann Wolfgang Goethe University

⁵School of Computing and Intelligent Systems, University of Ulster

⁶Department of Computer Science, Aberystwith University

⁷Dalle Molle Institute for Artificial Intelligence (IDSIA)

⁸Computer Science Department, University of Massachusetts - Amherst

Abstract

This short paper presents the core ideas of the IM-CLeVeR Project. IM-CLeVeR aims at developing a new methodology for designing robot controllers that can: (a) cumulatively learn new skills through autonomous development based on intrinsic motivations, and (b) reuse such skills for accomplishing multiple, complex, and externally-assigned tasks. This goal will be pursued by investigating three fundamental issues: (a) the mechanisms of abstraction of sensorimotor information; (b) the mechanisms underlying intrinsic motivations; (c) hierarchical architectures that permit cumulative learning. The study of these issues will be conducted on the basis of empirical experiments run with monkeys, children, and human adults, with bio-mimetic models aimed at reproducing and interpreting the results of such experiments, and through the design of innovative machine learning systems. The models, architectures, and algorithms so developed will be validated with experiments and demonstrators run with the simulated and real iCub humanoid robot.

1. Introduction

How can we create truly intelligent and autonomous machines and robots? This goal has both a huge technological and scientific importance. As a technology, autonomous intelligent machines can be exploited, for example, to perform repetitive tasks that humans do not like to carry out and conduct missions in hostile environments. On the scientific side, the ability to construct truly intelligent machines can shed new light on the mechanisms underlying learning and intelligence of humans and other primates, thus also enabling better treatments of psychiatric and neurological disorders.

The IM-CLeVeR project aims at developing a new design methodology for building autonomous intelligent robots based on intrinsically-motivated cumulative learning of skills. The central idea behind this new design methodology is that, instead of directly programming, training or evolving a set of specific skills in robots, we should endow them with developmental programs that allow an autonomous development of the needed skills on the basis of prolonged periods of interactions with the environment under the guidance of intrinsic motivations. Robots could then use the general abilities so acquired as building blocks for the solution of tasks that are relevant for the robot's users. Notice how these types of processes mark some of the most intelligent aspects of complex

organisms' behaviour, in particular human and non-human primates. For example, children at play carry out several activities driven only by intrinsic motivations such as curiosity. These activities allow them to acquire knowledge and skills exploited in later adult stages to pursue useful goals. The main objectives of the project will be pursued with these phenomena in mind.

2. Research issues

The central working hypothesis of the project is that cumulative, open-ended learning in artificial systems must be based on three fundamental principles:

1. Hierarchical architectures. Cumulative learning architectures for controlling robots should have the capability of developing sensorimotor and cognitive skills in an incremental hierarchical fashion. This requires: (a) acquiring skills and systematically increasing their complexity; (b) learning new skills using previously acquired skills as building blocks; (c) storing new skills without forgetting (and possibly improving) previously acquired ones.
2. Novelty detection and intrinsic motivations. A cumulative learning robot needs internal drives that focus learning on skills that: (a) are novel for the robot; (b) are within the robot's 'zone of proximal development' that is the robot has the drive to learn new skills that can be acquired on the basis of those already in its repertoire. To achieve this, the robot should be endowed with 'intrinsic motivations' that lead it to autonomously engage in activities that produce the maximum learning rate and/or information gain. Internal motivations differ from external motivations and rewards as the latter are associated with the practical outcomes that actions produce on the external world (e.g., food or sex in organisms or accomplishment of users' goals in robots). Intrinsically motivated learning must rely on 'novelty detectors', devices capable of monitoring and measuring the level of subjective novelty of action outcomes and learning rates so as to focus robot's activity on suitable experiences and boost learning speed.
3. Sensory abstraction and attention. Although sensory abstraction is a widely-investigated topic in cognitive sciences (e.g. in computer vision), the project will aim at isolating and studying the particular problems of abstraction related to the specific topics of the project, namely novelty detection and hierarchical architectures for cumulative learning.

3. Objectives

IM-CLeVeR has four main scientific and technological objectives:

1. To advance our knowledge about how cumulative learning is achieved in natural organisms. To this purpose, the project involves the implementation of empirical non-invasive experiments on intrinsically motivated learning in monkeys, children, human adults, and Parkinson patients, on the basis of novel experimental paradigms suitable for studying exploration, novelty detection, and the (cumulative) acquisition of novel actions .
2. To advance our knowledge about the mechanisms underlying intrinsically motivated cumulative learning in natural organisms. To this purpose, the project will develop bio-mimetic models (including both computer simulations and robotic experiments) aiming at reproducing and explaining the empirical findings provided by the aforementioned empirical experiments. In addition to its scientific value, this effort will also allow isolating new computational principles exploitable in robots.
3. To develop new machine learning techniques, architectures, and learning algorithms for the optimal design of cumulative learning robots. In particular, the project will aim at making substantial progress in the three distinct but related principles of the project working hypothesis: (a) hierarchical architectures, (b) intrinsic motivations, and (c) perceptual abstraction and attention.
4. To integrate the knowledge gained by the empirical experiments, the bio-mimetic computational models developed to interpret them, and the machine learning architectures and algorithms for building real robots demonstrating cumulative learning abilities. This challenge will involve the use of three iCub humanoid robotic platforms for the development of two demonstrators: CLEVER-K, a technologically-oriented demonstrator that will be tested in a kitchen scenario, and CLEVER-B, a demonstrator with which will be used to reproduce and interpret the results of the experiments carried out with monkeys and children.

Acknowledgements

IM-CLeVeR is supported by the European Commission under the 'FP7 Cognitive Systems, Interaction, and Robotics Initiative', grant no. 231722.

Emotion Non Verbal Behavior Modeling: Low and High Exhibitors

Stefania Balzarotti and Rita Ciceri

Laboratory of Communication Psychology, Università Cattolica, Milano
stefania.balzarotti@unicatt.it; rita.ciceri@unicatt.it

Abstract

The key role that emotion non verbal behavior may play in the interaction between human and artificial agents, including anthropomorphic robots, is today well-known. Within this broad research area, this study took into consideration the role of the individual style in emotion multimodal expression as a relevant feature to achieve an effective and natural emotional interaction between human and robot. We analyzed the multimodal emotional behavior of 163 human subjects who were watching an emotional disgusting film (Gross, 1998). All subjects were asked to answer three questionnaires to assess their dispositional regulatory strategies. Results showed: (1) different use of multiple behavioral categories; (2) strong inter-individual variability among low and high exhibitors; (3) significant differences with respect to individual styles of regulation and coping, when their measure is highly contextualized. All these factors should be considered when modeling emotion.

1. Introduction

The development of robots and artificial agents is driven today by new application domains where the ability of the embodied agents to interact with people as veritable partners may be crucial to achieve efficacy (Breazeal, 2003). Within this research field, it has been soon clear that emotional behavior plays a key role to regulate the HM interaction (Picard, 1997). Research has developed to provide computers with emotional skills: emotion *simulation* aims at the implementation of artificial embodied agents capable to reproduce human emotional expressions, e.g. robots and ECAs able to support the natural communication modalities of humans (Breazeal, 2003); on the other side, emotion *decoding* is meant to design agents able to recognize the user's emotional responses from real-time capturing of multiple signals.

Whatever the research goal is, modeling emotions and non verbal expression has revealed a demanding challenge and research is still far from supporting the complexity and the multimodal richness of human face-to-face communication. In this study, we considered two open issues. First, an essential property of human emotional non verbal behavior is *multimodality*: for this reason, computation research has based empirical investigation on the combination of multiple modalities

collecting various body measures in the attempt to reproduce the multimodal richness of the emotional process.

Second, emotion and non verbal expression emerge as a result of the interaction of an embodied system with a physical and social environment and are subjected to developmental processes. In other words, development and interaction give rise to *individual styles* of emotion expression: if the goal is to achieve the implementation of robots and artificial agents able to behave like humans – and whose behavior is governed by the same principles – individual differences in emotion expression should be considered in modeling those agents. Moreover, individual styles clearly appear as crucial in emotion recognition as well. For instance, it has been showed that non verbal expression is highly influenced by regulatory strategies (Gross, 1998).

Method

Sample: 163 nursing students (age: M=22,5; SD=4,47; M=28; F=135; 82,8% women) were recruited from the Faculty of Nursing Sciences of the Bicocca University in Monza and Lecco. The study employed nursing students since the surgery displayed by the stimulus belongs of their professional experience and learning.

Stimuli: A baseline clip that elicited very little emotion of any kind (a documentary); an emotional film clip (1 min) showing the amputation of an arm, which had elicited self-reported disgust in previous studies (Gross, 1998). Given our sample, a wider range of emotions could be expected, such as interest.

Dispositional measures: All subjects were previously asked to answer three questionnaires to monitor their regulatory and coping styles: two well-validated inventories (ERQ, COPE) tested general regulatory strategies and one questionnaire structured ad hoc for the study (Level of Distancing) assessed coping strategies in front of contextualized events belonging to a nurse experience (death of patients).

Behavioral measures: A camera recorded participant's face and upper body movements. Behavior was rated using the Behavioral Coding System (BCS, Ciceri and Balzarotti, 2008). The BCS is a multimodal system constituted by different macro-categories: face (divided into upper face, lower face and lip movements), gaze direction, posture, head movements, and vocal behavior for a total of 52 behavioral units scored. The 163 videos were rated by two coders (The Observer XT 7.0)

who received an extensive training in the use of this coding system to achieve adequate levels of inter-rater reliability (Cohen's $K = .89$).

Results

Frequency rates were extracted and computed for each macro-category (mean score). Considering the overall sample, significant differences between baseline and amputation clips were found for all categories [average $Z = -4.821$, $p < .001$], except for posture and head [average $Z = -.767$, $p > .05$].

Secondly, we focused on upper and lower face categories, where the greatest (positive) difference with respect to the baseline clip had emerged, and computed a global rate for face. Facial units were characterized by a highly-dispersed non-normal frequency distribution and the sample could be clearly divided into three groups. The first group (Non-Exhibitors, 30%) included subjects who didn't show any facial unit at all: this group totally suppressed facial behavior during the amputation clip showing significantly less facial behavior than during the baseline [$Z = -4.908$, $p < .001$]. The second group (Low Exhibitors, 42%) included subjects who showed a number of facial units below the mean ($M = 0.62$); baseline and amputation clips significantly differed for upper face [$Z = -3.743$, $p < .001$] but not for lower face [$Z = -1.153$, $p > .05$]. The third group (High Exhibitors, 28%) included subjects who showed a number of facial units above the mean; a significant difference was found between baseline and amputation [$Z = -5.178$, $p < .001$].

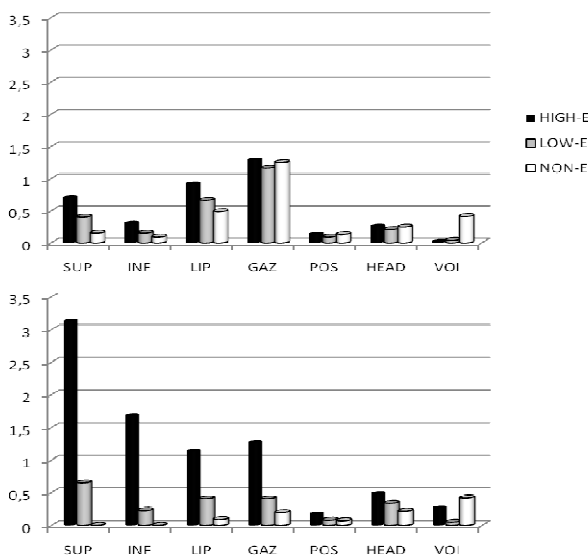


Figure 1 – Behavior rates for Non, Low and High Exhibitors

As to other behavioral categories, High Exhibitors showed significantly more head and vocal units [$Z = -4.908$, $p < .001$], whereas Non- and Low Exhibitors used significantly less gaze and lip movements during the amputation than during baseline clip.



Figure 2 – Examples of individual styles

Finally, we tested whether this variability in the use of non verbal behavior could be explained by individual regulatory styles. No significant correlations emerged with respect to standard questionnaires such as the ERQ and COPE; however, Non-, Low- and High Exhibitors significantly differed with respect to their Level of Distancing ($F_{2,161} = 4.674$, $p < .05$). In conclusion, our results showed: 1) a different use of multimodal categories (e.g. upper and lower face are the mostly used categories); 2) high inter-individual variability in the exhibition of facial behavior; 3) behavioral “freezing” rather than expression for certain individual styles (Non-Exhibitors); 4) the correlation with contextualized dispositional measures.

References

- Breazeal, C. (2003). Emotion and sociable humanoid robots. *International Journal of Human-Computer Studies*, 59, pp.119-155.
- Ciceri, R. and Balzarotti, S. (2008). From signals to emotions: Applying emotion models to HM affective interactions. In: Jimmy Or (Ed.) *Affective Computing: Emotion Modeling, Synthesis and Recognition*, I-Tech Education and Publishing, Vienna, Austria.
- Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., Taylor, J. G. (2001). Emotion recognition in human-computer interaction. *IEEE Signal Processing Magazine*, 18, 32–80.
- Gross, J. J. (1998). Antecedent- and response-focused emotion regulation: Divergent consequences for experience, expression, and physiology. *Journal of Personality and Social Psychology*, 74, 224-237.
- Kaiser, S. and Wehrle, T. (2001) Facial expressions as indicators of appraisal processes. In: *Appraisal processes in emotions: Theory, methods, research*, Scherer, K. R., Schorr, A., Johnstone, T. (Eds.), (pp. 285-300), Oxford University Press, New York.
- Picard, R.W. (1997). *Affective Computing*, The MIT Press, Cambridge, MA.

Proximo-Distal Competence Based Curiosity-Driven Exploration

Adrien Baranes, Pierre-Yves Oudeyer

INRIA Bordeaux - Sud-Ouest

351 Cours de la Liberation, 33405 Talence, France

{Adrien.Baranes, Pierre-Yves.Oudeyer}@inria.fr

Learning constraints and guiding mechanisms are involved since the very beginning of the infant development. Allowing a progressive and open-ended scaffolding of new skills, these mechanisms have been described as crucial, by psychologists and neuroscientists. Developmental heuristics presented here are directly inspired by the ability to control the growth of complexity of both exploration and learning in human children. More precisely, we focus on *intrinsic motivations* guiding mechanisms, responsible of spontaneous exploration, and on *maturational evolution* of the neural and muscular systems, that progressively allow the organism to control novel muscles, and thus, to increase its number of degrees of freedom (Lungarella and Berthouze, 2002). Therefore, we present a system using both self-motivation, and neuro-physiological maturation in an integrated computational mechanism, that aim to guide a robot, to gradually explore and learn its sensorimotor space.

1. Competence Based Intrinsic Motivation System

Previous work, presented in (Oudeyer et al., 2007), (Baranes and Oudeyer, 2009) introduced IAC and R-IAC as two knowledge-based (Oudeyer and Kaplan, 2008) computational models of intrinsic motivation in which a robot was motivated to explore sensorimotor subspaces where its predictions of the consequences of its actions increased maximally fast. These algorithms were shown to allow for self-organized developmental trajectories (Oudeyer et al., 2007) as well as efficient active learning of sensorimotor forward models. (Oudeyer and Kaplan, 2008) introduced the competence based intrinsic motivation framework, in which measures of interest are related to properties of the achievement of self-determined goals rather than to the properties of forward model predictions. In this poster, we introduce a competence based version of the R-IAC system (Baranes and Oudeyer, 2009). This system is composed of a forward model (to be learnt), a goal selection system which chooses goals with a probability proportional to the expected

progress in their mastery, and a controller/planner which allows the robot to reach a selected goal reusing the forward model. In analogy to R-IAC, the space of potential goals, i.e. of sensorimotor configurations to be reached, is split into subregions in each of which the mastery progress is monitored. Hence, the mastery progress, defined as the derivative of the evolution of errors in reaching goals in a particular subregion, is used as the measure of interestingness of given goals. Thanks to this measure of mastery progress, this algorithm allows the robot to explore and attempt goals of gradually increasing complexity. Additionally, we couple this mechanism with physiological maturation constraints as we will now explain.

2. Physiological Constraints

Over the first years, the physiological development of infants represents a very important constraint for the exploration and learning process. An important aspect of the maturation of the neural system is the myelination process, which leads to the extensive development of the corticospinal tract. This internal constraint allows a gradual development of the infants ability to control the distal musculature, from the trunk to hands (proximo-distal vector), and from the center of the body outwards (cephalo-caudal vector) (Kuipers, 1981). For instance, in the reaching task, the fact that young infants predominately use the musculature of the proximal arm and trunk, simplify the learning problem by reducing the functional degrees-of-freedom of the arm. In this poster, we propose to introduce such maturational constraints, progressively unfreezing the robot degrees of freedom, and their interaction with the intrinsic motivation system, described previously.

3. First Experiments

3.1 Competence Based Curiosity

Considering a robot controlled in a *configuration space* \mathbf{C} , and evolving in an *operational space* \mathbf{O} ,

monitoring the mastery of reaching goals can be treated at different level. Firstly, the *macro-level* considers the mastery to reach a precise goal state: in the case of an arm control task, a goal state can be described as a vector of precise value, in the *operational space* \mathbf{O} (for instance, the position of the arm extremity). Secondly, the mastery study can be performed at a *micro-level*, which is defined as the competence to perform micro-movements toward a precise goal in \mathbf{O} , from states in the operational space. Therefore, in the *macro-level* vision, we are able to monitor the competence level, to precisely reach a goal, and, in the *micro-level*, we are able to monitor the competence level, to perform micro/primitives actions.

In the first serie of experiment presented here, we analyse the behavior of the *micro-level* approach, while the full version of the poster will introduce the *macro-level*.

3.2 Proximo-Distal Evolution

The proximo-distal and cephalo-caudal maturation of humans can be described as the release of each controllable joint, following a sequence which depends on its morphology. This sequence can be implemented as a graph, whose each node represents an available joint, and each link is assigned with a weight representing a needed maturational level, to evolve to the next joint. Here, the needed maturational level is described as depending on two notions: (1) The *maturational* needed age, which has to be handcrafted, as a genetically coded value, it could be fixed proportionally to the number of learning experiments. (2) The *global competence level*, which depends on the global competence progress, allowing the passage to the next joint only if it is stabilized.

3.3 Experiment

The following experiment involved a simulated single *eye*, with pan/tilt rotations capabilities controled by joints velocities $(\dot{q}_{11}, \dot{q}_{12}) \in \mathbf{C}$, and a 2-joints *arm*, controled by $(\dot{q}_{21}, \dot{q}_{22}) \in \mathbf{C}$, possessing a visible extremity (simulating a hand). The couple $(x, y) \in \mathbf{O}$ represents the hand position in the *eye* referential, and $v \in \mathbf{O}$, a Boolean value meaning the presence of the hand, in the camera sight. The global system replies to the mapping $(q, \dot{q})_t \mapsto (v, x, y)_{t+1}$ with $q = (q_{11}, q_{12}, q_{13}, q_{14})$ and $\dot{q} = (\dot{q}_{11}, \dot{q}_{12}, \dot{q}_{13}, \dot{q}_{14})$. By choosing goal values $(v, x, y) \in \mathbf{O}$ and trying to reach it, the system is able to learn the forward model $(q, \dot{q})_t \mapsto \delta(v, x, y)_{t+1}$ (where $\delta(v, x, y)_{t+1} = (v, x, y)_{t+1} - (v, x, y)_t$), and motor skills (policies) $\pi(v, x, y, q) = \dot{q}$ of different complexities. In the studied configuration, we can point different kind of skill complexity, like the ones where the hand is not in the sight of the camera ($(v, x, y) = 0$), which can be con-

sidered as easy space subregions, or the ones where it is, which contains more skills to learn. The proposed experiment consists of observing the learning behavior of three approaches of goal selection $(v, x, y) \in \mathbf{O}$. The first one, which represents a uniform selection inside the whole space, is called *Random*. The second approach, called *RIAC Competence* represents the implementation of the Competence Based Intrinsic Motivation heuristic (without physiological constraints). Finally, the *Proximo-Distal Competence Based Curiosity* (PDCC) heuristic (using physiological constraints) is evaluated considering two stages, the system beginning by just moving its camera (values $(\dot{q}_{11}, \dot{q}_{12})$), and freeing its arm (values $(\dot{q}_{21}, \dot{q}_{22})$) on the second stage.

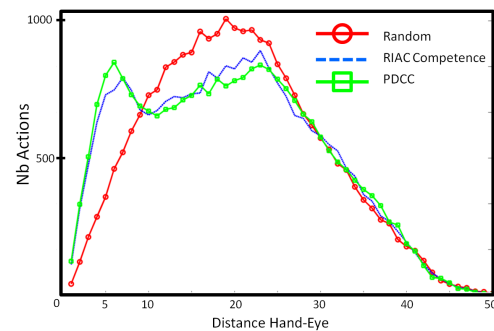


Figure 1: Histograms of Hand-Eye distances

The previous figure shows that RIAC Competence-based guides the eye to focus on the hand, more than the random guiding approach, and that the PDCC approach guides it toward the hand more than the two others. This allows us to argue that using both physiological constraint and competence based approaches can guide the system to avoid a too important focalization on too simple areas and guide it toward skills of intermediate or high complexity.

References

- Baranes, A. and Oudeyer, P.-Y. (2009). R-iac: Robust intrinsically motivated active learning. In *Proc. of the IEEE International Conference on Learning and Development*.
- Kuipers, H. (1981). *Anatomy of the descending pathways*. American Physiological Society.
- Lungarella, M. and Berthouze, L. (2002). Adaptivity via alternate freeing and freezing of degrees of freedom. In *Proc. of the 9th Intl. Conf. on Neural Information Processing*.
- Oudeyer, P.-Y. and Kaplan, F. (2008). How can we define intrinsic motivations ? In *Proc. Of the 8th Conf. On Epigenetic Robotics*.
- Oudeyer, P.-Y., Kaplan, F., and Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(2):pp. 265–286.

The role of internal value systems for a memory-based robotic architecture

Paul Baxter¹, Will Browne²

¹ Cybernetics Intelligence Research Group, University of Reading, U.K.
p.e.baxter@reading.ac.uk

² School of Engineering and Computer Science, University of Wellington, NZ
will.browne@ecs.vuw.ac.nz

Abstract

It is proposed that the value (or motivational) system of an agent provides the drive for the development of behavioural competences, but also imposes constraints on this process. This study provides an example of these proposed opposing effects in the context of a novel developmental Memory-Based Cognitive Framework: whilst a value system is necessary, its precise implementation may impose limitations on the acquisition of behavioural competencies through environmental interaction.

1. Introduction

Deriving principles from biology in the implementation of behaviourally flexible and autonomous robotic agents is now a well accepted methodology (Guillot and Meyer, 2001). Of particular importance for the present work is the principle of developmental learning (Weng, McClelland et al., 2001), and that cognition may be described as being fundamentally concerned with the manipulation and utilisation of memory (e.g. (Fuster, 1997)). Although autonomy has been proposed to result from self-sustaining processes, this is not currently possible in artificial agents, necessitating an emotion/value system to bridge the gap (Ziemke, 2008). This value system is proposed to supply the most basic drives of the agent (Franklin and Ramamurthy, 2006), which enables the functionally evaluative mechanism necessary to distinguish that which is beneficial for the agent from that which is harmful.

A number of types of value system have been proposed, e.g. homeostatic systems (Di Paolo and Iizuka, 2008) and intrinsic motivation (Oudeyer and Kaplan, 2007), but all emphasise this system as a driver of behaviour, without necessarily considering how such functionality constrains development. In this study, two types of value system are implemented in the framework of a novel cognitive architecture, with the particular aim of assessing how these influence the development of behaviour of a simple robotic agent.

2. MBCF/EMA and Value Systems

The Memory-Based Cognitive Framework (MBCF), and its corresponding computational architecture, the Embodied MBCF Agent (EMA), have been constructed based upon the previously described principles. In this context, memory is held to be associative, and is represented in the EMA by discrete objects (unlike a fully connected artificial neural network), where each of these objects (named 'cognits') explicitly link two elements from a lower level (figure 1) retrospectively, and have an activation value. Figure 1 describes the arrangement of groups of such cognits, where the arrows indicate the flow of activation over the course of one time step, and the action executed by the agent is simply the motor space element with the highest activation value. The manipulation of activation levels is thus important, since it is the totality of activation in the EMA layers which determines the executed action. It is proposed that the application of the term 'developmental' to this system is justified, since it exhibits not only changes in behaviour over time (i.e. learning), but also creates the structure of representation and action (i.e. the creation of the cognit structure, which *is* the control structure) (Meeden and Blank, 2006). Further details of the MBCF and EMA may be found in (Baxter and Browne, 2009).

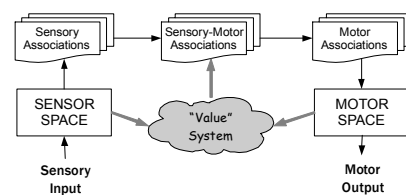


Figure 1: The EMA structure used in the present work. The Value system acts as a scalar of sensory-motor association activation levels, thus influencing the behaviour of the system.

The value system acts upon the activation levels of sensory-motor layer cognits. Each of these cognits has a 'value tag' (a float in the interval [0,1]) which acts a scalar for its activation level on every time-step. The assignment of this value tag is the subject of this paper. Two possible schemes are compared: (1) *static*, where

the assignment of an unchanging value tag is based on a manually designed transfer function; (2) *dynamic*, where the value tag is updated based on reinforcement learning principles. The former is thus only adaptive in terms of the number of cognits in the system, whereas in the latter case, both the number of cognits and the value tags thereof are adaptive. In addition to these schemes, a no-value system setup is implemented for comparison, where each of the value tags is initialised to 1.0, and remains at that value: i.e. there is no functional role for the value system.

3. Experiment and Results

For the present experiment, three versions of the EMA are implemented (the static, dynamic and no value system setups), and two benchmark controllers (a random walker, and a behaviour-based reactive obstacle avoidance controller). The aim for each of the agents was to learn the necessary sensory-motor coordination in order to move around freely in an open, bounded arena (i.e. obstacle avoidance). No cognits are present in the system at the start of each run. Figure 2 shows the results of this experiment.

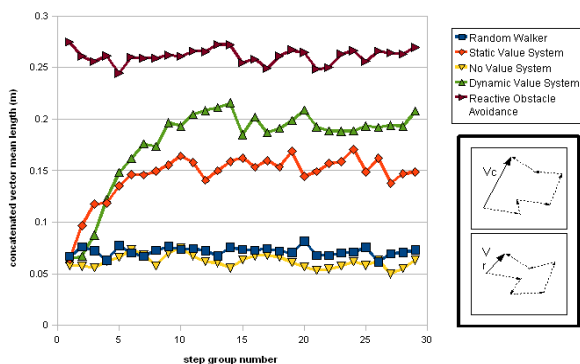


Figure 2: Concatenated vector mean length results (10 repeat runs, each of 5000 time-steps in a bounded open square environment) for five different setups: the development of behaviour of the static and dynamic value system setups is clear, whereas the no-value system EMA has a performance comparable to a random walker. (*inset*) derivation of the concatenated vectors: $|V_c| \gg |V_r|$, even though the constituent vectors (each of which represents the movement of the agent over a single time-step) are the same length.

There are two points which are of particular interest. The first is that the EMA requires a value system implementation in order for any meaningful behaviour to be acquired: the setup with no value system exhibits performance comparable with that of the random controller benchmark. The second point is the difference in final performance between the two value system setups. The advantage that the dynamic value system setup exhibits suggests that it is able to better adapt to the task and environment than the static value system setup (as might be expected). This result may alternatively be interpreted as a constraint on the

development of behaviour: the particular instantiation of the static value system prevents a level of performance which could potentially be achieved by the EMA. Indeed, if the behaviour-based reactive obstacle avoidance controller is seen as the optimal behaviour in this environment given this task, then the dynamic value system setup is similarly constrained. Just as the definition of a fitness function is central to the success of an evolutionary robotics setup, this outcome indicates that the definition of a value system in a developmental robotic architecture requires similar attention.

4. Conclusions

These results have supported the proposed necessity of a value system in the progressive acquisition of behavioural competencies. Furthermore, the different performances of the two value system setups provides an indication of the limitations that value systems can impose. This suggests that the constraining effects on development of such a value system should, along with the motivational drives it provides, also be considered.

References

- Baxter, P. and W. Browne (2009). Memory-Based Cognitive Framework: a Low-Level Association Approach to Cognitive Architectures. *European Conference on Artificial Life (ECAL'09)*. Budapest, Hungary.
- Di Paolo, E. and H. Iizuka (2008). "How (not) to model autonomous behaviour." *BioSystems* **91**(2): 409-423.
- Franklin, S. and U. Ramamurthy (2006). Motivations, values and emotions: three sides of the same coin. *6th International Workshop on Epigenetic Robotics*. Paris, France, Lund University Cognitive Studies. **128**.
- Fuster, J. M. (1997). "Network Memory." *Trends in Neuroscience* **20**(10): 451-459.
- Guillot, A. and J.-A. Meyer (2001). "The Animat contribution to Cognitive Systems Research." *Journal of Cognitive Systems Research* **2**: 157-165.
- Meeden, L. A. and D. S. Blank (2006). "Introduction to Developmental Robotics." *Connection Science* **18**(2): 93-96.
- Oudeyer, P.-Y. and F. Kaplan (2007). "What is Intrinsic Motivation? A Typology of Computational Approaches." *Frontiers in Neurobotics* **1**(6).
- Weng, J., J. McClelland, et al. (2001). "Autonomous mental development by robots and animals." *Science* **291**: 599-600.
- Ziemke, T. (2008). "On the role of emotion in biological and robotic autonomy." *BioSystems* **91**(2): 401-408.

Gesture recognition as a prerequisite of imitation learning in human-humanoid experiments

Florian A. Bertsch and Verena V. Hafner
Cognitive Robotics Group
Department of Computer Science
Humboldt-University Berlin, Germany
{bertsch,hafner}@informatik.hu-berlin.de

Abstract

Behavior recognition is one of the skills necessary for imitation learning. We present an approach that shows that real-time learning of visually observed dynamic gestures is possible, and outline how it could be used for imitation learning experiments.

1. Introduction

Imitation learning is an important but difficult skill that develops in early infancy. The prerequisite for imitation, which goes beyond a reflex-like behavior, is the recognition of the behavior of another person, being also one of the main prerequisites for communication. Gesturing is a good example of a behavior that is relatively simple but contains a large amount of information.

Robotic experiments on imitation learning (see (Dautenhahn and Nehaniv, 2002) for a review) have focused on different aspects of imitation learning. (Saunders et al., 2007) focus on self-imitation as a first step towards the imitation of others, since it can be learned using one's own sensorimotor feedback in a self-supervised manner. (Hafner and Kaplan, 2005) studied the development of body maps and interpersonal maps as a method for the recognition of body space and (interaction) behavior. Imitation is important for interpersonal interaction, learning and the ability of behavior recognition, and it is an important prerequisite for social interaction (Meltzoff and Gopnick, 1993).

In our approach, we presented a method for learning and recognizing visually observed dynamic gestures of humans. The method is applied on the humanoid robot Nao and tested in a real-time human-robot interaction experiment. We outline how this experimental setup and method can be used for natural and intuitive human-robot interaction and for the study of imitation learning and aspects of it such as the body correspondence problem.

2. Gesture Recognition

Choice of gestures and data acquisition

To ensure that our method can be calculated in real-time, we focus on gestures that can be described by the hand movements of a human within the image plane. This condition restricts the set of gestures we may use to such cases where the gesticulating person uses "large-scale" movements of the hands. While people typically use mimic and finger gestures during a conversation, they use "large-scale" movements when they gesticulate over large spatial distances. Therefore, we chose eight sample gestures out of a set of gestures used by construction workers to instruct vehicle drivers (see Fig. 1). We captured a set covering these gesture types, which contained 212 single gestures performed by 9 different persons (see Fig. 2).

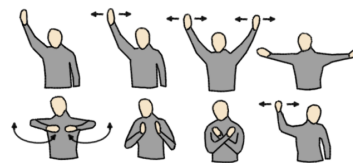


Figure 1: Set of sample gestures

The segmentation of the video stream into segments that correspond to single gestures is done automatically in a way that can be used in online experiments as well. Therefore, a resting posture (both arms are hanging down beside the torso) is recognized and each movement between leaving and returning to the resting posture is assumed as a gesture.

Recognizing a fixed gesture set

In a feature extraction step we localized and tracked the face and hands of the gesticulating persons. The extraction methods are based on a Viola-Jones-Detector and an adaptive color model. Based on these features we compared linear discriminant analysis (LDA), support vector machines (SVM) and hidden Markov models (HMM) for the recognition

task using cross-validation in a “leave-one-person-out” manner. As result we obtained a similar recognition rate for the different approaches of approx. 0.9 (in comparison to 1/8 for a random guess). We observed that the result mainly depends on the way the features are preprocessed. Following these findings, we propose an approach which relies on a simple recognition method based on histograms and avoids complex and costly methods.



Figure 2: Sample pictures from the video capturing of 9 persons performing 212 gestures of 8 different types.

Learning unknown gestures

In addition to the approach described in the last section, we developed a method to learn unknown gestures by observation that can be applied to a human-humanoid interaction scenario. When presenting a sequence of gestures unknown to the humanoid, it can learn new gesture types by grouping similar gestures together. This approach aims at developing methods that enable a humanoid to learn new behaviors by observing a human who not necessarily pays attention to the humanoid. To construct an appropriate method for this approach, we performed a comparative analysis of different feature representations and clustering methods. As result we obtained an online clustering method that is based on a specific distance measurement between observed gestures. Therefore the movement of each hand is decomposed into directed line segments and the distance of two gestures is defined as the sum of the smallest distances between the pair-wise most similar line segments. To evaluate this approach we applied it to gesture sequences which were generated by randomly choosing 5 gestures of each gesture type. As result we obtained an average adjusted Rand index of 0.7, a value indicating a successful grouping of the real gesture types.

3. Interaction game

To show the possibilities of the described approaches when used as basis for human-humanoid interaction and imitation, we arranged a gesture-based interaction game between a human and the humanoid robot Nao (see Fig. 3). The game consisted of alternating gesture recognition and presentation tasks for both participants to demonstrate the humanoid’s gesture recognition skills as well as its ability to use its human-like shape to perform gestures by itself.

This experiment uses a fixed predefined gesture set which is recognized using the method describe in section 2. and performed by the robot using a predefined movement pattern. The interaction game can be considered as an attempt to organize a setting which is convenient for future advanced imitation experiments based on visual gesture recognition and learning skills.

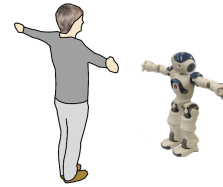


Figure 3: An interaction game demonstrates a gesture-based mutual human-humanoid interaction with a Nao.

4. Future imitation experiments

The presented approach of learning by observation allows learning to recognize unknown gestures without an explicit training session. It would be desirable to extend this skill in a way that the humanoid is not only able to recognize the new gestures but also to perform them itself. This would be a typical human-robot imitation task which seems to be easily achievable when using the gestures we focused on. It seems to be promising to set up a relation between the observed hand positions and the humanoid’s posture which should lead to the ability to repeat observed gestures in the desired manner.

References

- Dautenhahn, K. and Nehaniv, C. (2002). *Imitation in animals and artifacts*. MIT Press, Cambridge, MA.
- Hafner, V. and Kaplan, F. (2005). Interpersonal maps and the body correspondence problem. In Demiris, Y., Dautenhahn, K., and Nehaniv, C., (Eds.), *Proceedings of the Third International Symposium on Imitation in animals and artifacts*, pages 48–53, Hertfordshire, UK.
- Meltzoff, A. and Gopnick, A. (1993). The role of imitation in understanding persons and developing a theory of mind. In S. Baron-Cohen, H. T.-F. and D.Cohen, (Eds.), *Understanding other minds*, pages 335–366. Oxford University Press.
- Saunders, J., Nehaniv, C. L., Dautenhahn, K., and Alissandrakis, A. (2007). Self-imitation and environmental scaffolding for robot teaching. *International Journal of Advanced Robotics Systems, Special Issue on Human - Robot Interaction*, 4:109–124.

Designing a Turn-taking Mechanism as a Balance Between Familiarity and Novelty

A. J. Blanchard, J. Nadel

Abstract

Novelty is a main source of exploration and learning. However, an ever-changing environment may hinder the anticipation of an action effect and lead to emergent behaviors that are detrimental to learning. In this paper we present a model where the exploration of the physical or social environment is related to a balance between seeking novelty and familiarity. From this emerge turn-taking in a social environment, with correlated benefits of learning and communication.

Exploration and familiarity Familiarity plays an important role to redirect attention toward exploration. It provides the ingredients to stabilize the relationship between the environment and the agent's behavior. But researching for novelty is also very important in order to learn (Oudeyer and Kaplan, 2004).

Within this perspective, an autonomous robot needs both to handle familiar elements and to find novelty in its environment. Applying this principle in (Blanchard and Cañamero, 2006) made the robot oscillate between phases of exploration and phases of familiarization. The principle is to inhibit the motor command as a function of the perceived novelty. If we plot the successive values of executed movements and perceived novelty on two orthogonal axes, we find graphs following the structure shown in Fig. 1.

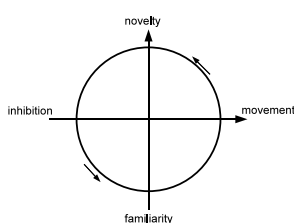


Figure 1: Perception and motor command of an agent on two orthogonal axes, we observe a cyclic dynamic.

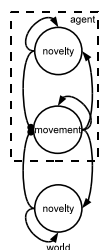


Figure 2: Oscillator generated by the modulation of movements from external and internal inhibition.

However this interesting behavior can become unstable if the consequences of its own movement are

delayed or if the perception is temporary disabled. Without perception of novelty, the motor command would increase infinitely. In addition, even if the level of novelty is constant, it can be useful to make the motor command vary and repeat same actions (Bolland and Emami, 2007). This allows the agent to experience different effects of the same movements in slightly different situations and to detect, via the repeated search of causal links between perception and action, whether the link was not due to random co-occurrence of events. Therefore we add an internal inhibition of the motor command in order to limit its amplitude but maintain minimal variations even when the perception of the world does not change (see Fig. 2).

New dynamic in a dyad We propose the above model to explain how oscillations needed for turn-taking (Prepin and Revel, 2007) are emerging. Although mainly quoted in its learning function, imitation is also a developmental means of communication where children take turns by alternating the roles of imitator and model (Nadel and Butterworth, 1999). Model and imitator form a new dynamic system, an evolving system of similarities built on the basis of two different individual repertoires from the interaction of which emerge new possibilities for both agents.

As a demonstration of the principle, we have set a simple virtual world implemented using “Pure Data” (Puckette, 2009). An agent can generate a movement M which makes its sensation S proportionally change—typically this represents an agent moving in the space where M is its velocity and S its position. Due to the advantages of simplicity, biological plausibility and stability of behavior in limit cases (specially for continuity at starting time), we reuse the detection of familiarity proposed in (Blanchard and Cañamero, 2006) to build the architecture presented in Fig 3.

We present the different kinds of exploration of only one agent for different parameters of the architecture in Fig 4.

In Figure 5 we present the movements of the two agents facing each other. We see that even if they start at the same time they synchronize their movement in anti-phase. This is the kind of behavior

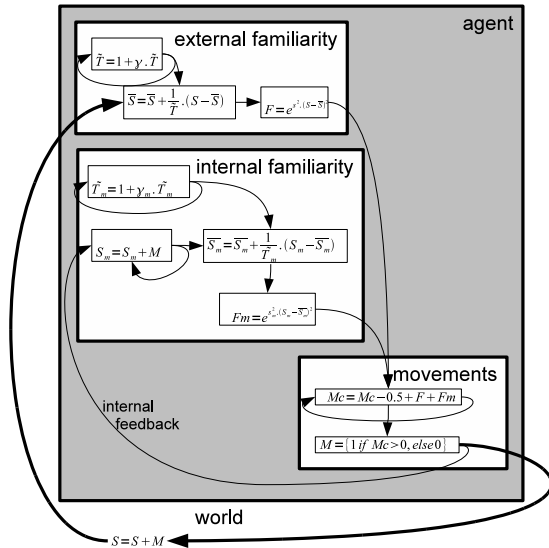


Figure 3: Implementation of one agent exploring a simple world. One part is controlling the internal familiarity of its motor command and another part is controlling the external familiarity of the world.

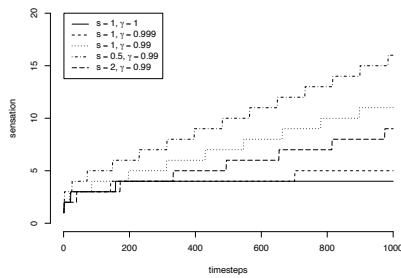


Figure 4: Sensation of the agent as a function of time for different values of sensitivity to novelty (s), and rate of habituation (γ).

we expect for turn-taking (Prepin and Revel, 2007, Revel and Andry, 2009).

Discussion We have proposed a new way to approach a difficult problem raised by the use of oscillators to simulate the alternation of novelty and familiarity. Indeed this alternation is needed to achieve an imitative turn-taking when the agent is in presence of a partner, but is also fruitful to achieve an optimal learning when an agent alone explores a physical environment (Bolland and Emami, 2007, Oudeyer and Kaplan, 2004).

We were first inspired by an original design using an attachment model where stability is a condition for exploration (Blanchard and Cañamero, 2006). After each novel stimulus leading to a novel motor response, or each novel movement leading to a novel perception via a novel stimulus, stability was taking

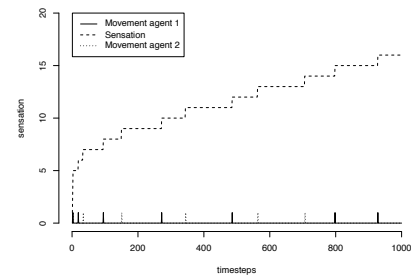


Figure 5: In a face to face setup, the agents automatically synchronize their movements in oppositions.

place.

We were guided to use oscillators as a way to produce synchrony and turn-taking by the collaborative work of Gaussier's team and Nadel's team (Nadel and Butterworth, 1999, Prepin and Revel, 2007, Revel and Andry, 2009). We joined the two options in this study, using oscillators to allow the emergence of turn-taking during interactive imitation, and inspired by the stability/novelty paradigm to synchronize behaviors.

Acknowledgement This research was funded by the European Project Felix Growing FP6ISI-045169 coordinated by Lola Cañamero.

References

- Blanchard, A. and Cañamero, L. (2006). Modulation of exploratory behavior for adaptation to the-context. II:131–139.
- Bolland, S. and Emami, S. (2007). The benefits of boredom: an exploration in developmental robotics. In *ALIFE*, pages 163–170.
- Nadel, J. and Butterworth, G., (Eds.) (1999). *Imitation in infancy*. Cambridge University Press.
- Oudeyer, P.-Y. and Kaplan, F. (2004). Intelligent adaptive curiosity: a source of self-development. In *Lund University Cognitive Studies*, pages 127–130.
- Prepin, K. and Revel, A. (2007). Human-machine interaction as a model of machine-machine interaction: how to make machines interact *Advanced Robotics*, 21(15).
- Puckette, M. S. (2009). <http://puredata.info/>.
- Revel, A. and Andry, P. (2009). Emergence of structured interactions: From a theoretical model to pragmatic robotics. *Neural networks*, 22(2):116–125.

Towards a new social referencing paradigm

S. Boucenna¹, P. Gaussier^{1,2}, L. Hafemeister¹, K. Bard³

¹ETIS, CNRS UMR 8051, ENSEA, Univ Cergy-Pontoise, ²IUF, ³Portsmouth University
 {boucenna,gaussier,hafemeister}@ensea.fr, kim.bard@port.ac.uk

How can a robot learn more and more complex tasks? This question is becoming central in robotics. In this work, we are interesting in understanding how emotional interactions with a social partner can bootstrap increasingly complex behaviors, which is important both for robotics application and understanding development. In particular, we propose that social referencing, gathering information through emotional interaction, fulfills this goal. Social referencing, a developmental process incorporating the ability to recognize, understand, respond to and alter behavior in response to the emotional expressions of a social partner, allows an infant, or a robot, to seek information from another individual and use that information to guide his behavior toward an object or event (Klennert et al., 1983).

Gathering information through emotional interaction seems to be a fast and efficient way to trigger learning. This is especially evident in early stages of human cognitive development, but also evident in other primates (Russell et al., 1997). Social referencing ability might provide the infant, or a robot, with valuable information concerning the environment and the outcome of its behavior, and is particularly useful since there is no need for verbal interactions. In social referencing, a good object or event is identified or signaled with an emotional message. The emotional values can be provided by a variety of modalities of emotional expressions, such as facial expressions, sound (a scream), gestures, etc. We choose to explore the facial expressions since they are an excellent way to communicate important information in ambiguous situations but also because they can be learned autonomously very quickly (Boucenna et al., 2008). Our idea is that social referencing as well as facial expression recognition can emerge from a simple sensori-motor system. All the work is based on the idea of the perception ambiguity: the inability at first to differentiate its own body from the body of others if their actions are correlated with its own actions. This perception ambiguity associated to a homeostatic system are sufficient to trigger first facial expression recognition and next learn to associate an emotional value to an arbitrary object. Without knowing that the other is an agent, the robot is able to learn some complex tasks. Hence we advocate the idea that the social referenc-

ing can be bootstrapped from a simple sensori-motor system not dedicated to social interactions.

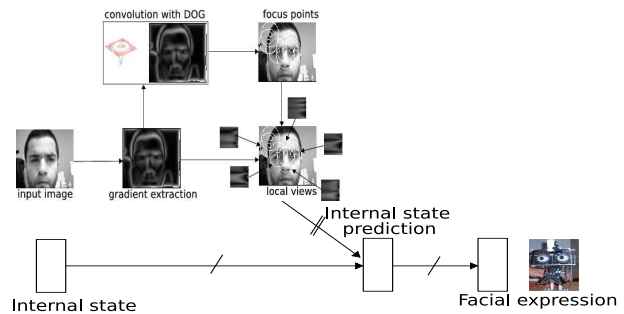


Figure 1: The global architecture to recognize facial expressions and imitate. A visual processing allows to extract the local views sequentially. The *internal state prediction* learns the association between the local views and the robot's internal state.

In our social referencing experiment, the set-up is the following: a robot head learns to recognize the facial expressions, an arm learns to reach an object in the workspace and an other camera views the workspace. Thanks to this set-up, the robot (head plus arm) can interact with the environment (human partner) and grasp objects. In the developed architecture, the robot learns to handle positive objects, and learns to avoid the negative objects as a direct consequence of emotional interactions with the social partner.

The robot head learns to recognize emotional facial expressions autonomously (Boucenna et al., 2008). The robot produces facial expressions (sadness, joy, anger, surprise and neutral face) and if the human mimicks correctly the robot expression, the robot learns to associate its proprioception (internal emotional state) with the human's facial expression. After few minutes of online learning (typically less than 3 minutes), the robot is able to recognize the human facial expressions as well as to mimick them (fig. 1).

After a visuo-motor learning, several positions in the workspace can be reached by the robot arm (Andry et al., 2001). One visual position corresponds to one or several motor configurations (e.g attractors). These attractors pull the arm in an attraction basin (the position target). This control is performed with a dynamical system in the aim

of smoothing the trajectory (Fukuyori et al., 2008). This dynamical system also uses a reinforcement signal in the aim of attaching a lot of or little importance to some attractors, for instance a reward can be given if the arm follows the right direction, otherwise a punishment. The reinforcement signal can be emotional (e.g happy facial expression is a positive signal and an angry facial expression is a negative signal).

As soon as the facial expression learning is performed (i.e the human partner must imitate the robot head between 2 and 3 min, then the robot is able to recognize and display the human facial expressions), the human can interact with the robot head to associate an emotional value to an object (positive or negative). The neural network (N.N) tries to correlate signals from the robot's internal state with external informations (e.g facial expressions or object attributes). The N.N does no distinction between the internal state and the facial expression recognized on the partner's face. In the absence of the internal state, the facial expression recognized induces an internal state which is associated with the object (a simple conditioning chain: figure 2).

Classical conditioning can perform this association between the emotional value that the human transmits and some areas of the image. The attentional process used in this model is very simple, the robot focuses on colored patches or textures. When focusing on an object, the robot extracts some focus points and associates the recognition of the local view surrounding these focus points with the emotional value of the robot. For instance, if the robot is in a neutral emotional state and the human displays a happy facial expression in the presence of an object, the robot will move to a joy state and will associate a positive value to the object. On the contrary if the human displays an anger facial expression, the value associated to this object will be negative. As soon as this learning is finished, the robot arm can handle or avoid the objects according to their associated emotional value. In other words, the emotional value associated to the object is the reinforcing signal that the arm uses so as to move (exactly as the human facial expression or an internal reward). Besides, the human partner can always provide an emotional value on the robot's behavior, there is a competition between the object emotional value and the partner's facial expression. The competition is performed with a simple WTA (Winner Take All). The robot could handle negative objects if the human partner displays a joy facial expression or it could avert positive objects if the human displays an anger facial expression (the object's emotional value is not overwritten).

We think this approach can provide new interesting insights about how humans can develop social

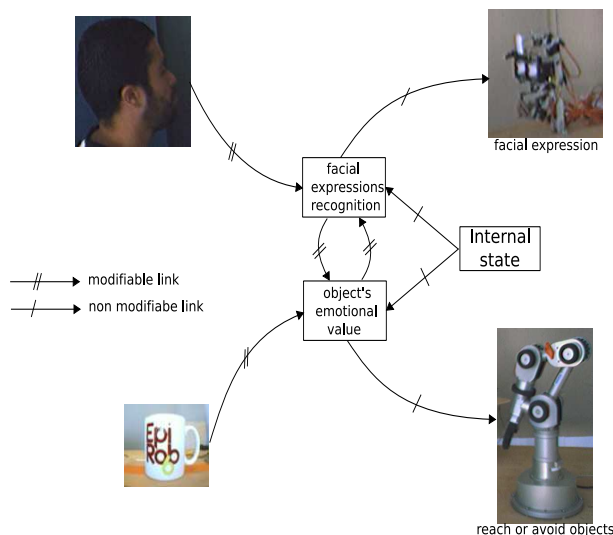


Figure 2: experimental set-up for the social referencing: The robot arm reaches the positive object and averts the negative object.

referencing capabilities from sensorimotor dynamics. In contrast to current developmental theory considering that social referencing is a complex cognitive process of triadic relations, the current work suggests 1) the primacy of emotion in learning, 2) the simple classical conditioning mechanisms by which another's emotional signal assumes identity with internal emotional states, and 3) a simple system of pairing internal emotional state with object-directed behavior.

Acknowledgments

The authors thank: J. Nadel, M. Simon and R. Soussignan and P. Canet for the design and calibration of the robot head. L. Canamero for the interesting discussions on emotion modelling. This study is part of the European project "FEELIX Growing" IST-045169, the French Region Ile de France "DIG-ITEO" and the Institut Universitaire de France.

References

- Andry, P., Gaussier, P., Moga, S., Banquet, J., and Nadel, J. (2001). Learning and communication in imitation: An autonomous robot perspective. *IEEE transactions on Systems, Man and Cybernetics, Part A*, 31(5):431–444.
- Boucenna, S., Gaussier, P., and Andry, P. (2008). What should be taught first: the emotional expression or the face? *epirob*.
- Fukuyori, I., Nakamura, Y., Matsumoto, Y., and Ishiguro, H. (2008). Flexible control mechanism for multi-dof robotic arm based on biological fluctuation. *From Animals to Animats 10*, 5040:22–31.
- Klennert, M., Campos, J., Sorce, J., Emde, R., and Svejda, M. (1983). The development of the social referencing in infancy. *Emotion in early development*, 2:57–86.
- Russell, C., Bard, K., and Adamson, L. (1997). Social referencing by young chimpanzees (pan troglodytes). *journal of comparative psychology*, 111(2):185–193.

Should I worry about my stressed pregnant robot?

David Bowes Lola Cañamero Rod Adams Volker Steuber
Neil Davey

Science and Technology Research Institute, University of Hertfordshire
Email: {D.H.Bowes, L.Cañamero, R.G.Adams, V.Steuber, N.Davey}@herts.ac.uk

1. Introduction

Since (Braitenberg, 1984), there has been a growing interest in the study of how to develop simple neuro-controllers for robots. This field of work now encompasses many different technical areas such as spiking neural networks, neural network evolution and neural network development (Floreano, 2005). Recent work has developed the idea of chemical signals having a role in the modulation of behaviours generated by artificial neural networks (Cañamero et al., 2002). Neural networks have been grown to produce 3D models of a known biological nature (Adams et al., 2004) so the idea that a genetic algorithm can produce a controller for a robot is not unknown. (Federici and Downing, 2006) demonstrated that including an embryonic development stage to the evolution and production of a robot controller increased the scalability of genetic algorithms. (Roggen et al., 2007) also conclude that morphogenesis provides an efficient mechanism for encoding the phenotype of an individual. (Miorandi and Yamamoto, 2008) summarises the research into bio-inspired systems, focusing mainly on the production of architectures using the phylogeny and ontology, and ignores epigenetic organization as proposed by (Sipper et al., 1997).

(Jablonka and Lamb, 2006) describe the role of epigenetic mechanisms which can have a long term impact on the behaviour/phenotype of an organism. They describe the main mechanisms by which the long term behaviour may occur such as gene switching, DNA methylation, physical organization and non-DNA mechanisms which can affect the expression of genes and the subsequent phenotype. (Tanev and Yuta, 2008) studied the impact that histones can have on gene expression and the adaptability of an organism.

Biological studies (Clarke et al., 1996, Laplante et al., 2008, Mastorci et al., 2009) show that pre-natal stress factors can affect the development and behaviour of post-natal offspring. It should be noted that the mechanisms by which the behaviour is attenuated is not understood and some results could be caused by embryo selection by the mother, rather than epigenetic effects (Ideta et al., 2009).

Recent work investigating the evolution of neural networks shows that repeating patterns of connectivity occur. In particular (Bowes et al., 2009) show that symmetrical lateral inhibition of neurons improved the ability of a robot to perform phototaxis in a variety of light conditions. (Oros et al., 2009) have also shown that bilateral symmetry improves the evolvability of neural controllers. Both studies include repeating patterns which may be coded using a GRN and developmental system as proposed by (Roggen et al., 2007).

2. Proposal

The overarching aim of this research is to produce and analyse mechanisms for creating robot controllers incorporating simulated neurons which have been produced using a GRN. The spiking neurons being affected by simulated chemicals which attenuate the simulated post synaptic membrane. This system is analogous to biological development where neural tissue is created in a pre-natal phase and the post-natal individual is responsible for providing a suitable environment for the development of the pre-natal individual. We will therefore study the relationship between simulated pre-natal development and simulated post-natal 'life'

The aim of this aspect of the research is to study mechanisms where the post-natal experience can affect the analogy of pre-natal development. This will require the creation of a post-natal robotic 'organism' which can survive in the equivalent of a 'hostile' environment, with desires for different resource. Having such an analogous 'organism' in the form of a robot, we intend determining a possible artificial genetic mechanism which would grow an appropriate neural network (Roggen et al., 2007). This regulatory genetic mechanism would be adapted to take into account the simulated chemicals from the post-natal environment and thus different instantiations of the robot would be produced depending on the environment available during the initial 'pre-natal' development.

The external environment that the robot will experience will consist of a light gradient which coincides with temperature and 'available food' in the form of simulated glucose. The robot will also have

an internal environment to represent the chemicals produced by the GRN and other chemicals such as glucose absorbed from the environment and waste products from metabolism.

The performance of the robots will be measured in terms of their ability to maintain a homeostatic stable environment in a range of environmental conditions which will be created by having varying light gradients in the robots living environment. A control group of robots will be used which do not have epigenetic mechanisms which we suggest will be able to cope when the environment does not fluctuate frequently. The epigenetic robots which experience environmental stress may produce offspring which are better adapted to the environment than the parent has just experienced. Hopefully, we should not be worried when the parent robot is stressed.

References

- Adams, R., Boekhorst, R., Rust, A., Kaye, P., and Schilstra, M. (2004). Design of spatially extended neural networks for specific applications. *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, 4:3101–3106 vol.4.
- Bowes, D., Adams, R., Canamero, L., Steuber, V., and Davey, N. (2009). The role of lateral inhibition in the sensory processing in a simulated spiking neural controller for a robot. In *Artificial Life, 2009. ALife '09. IEEE Symposium on*, pages 179–183.
- Braitenberg, V. (1984). *Vehicles: Experiments in Synthetic Psychology*. The MIT Press.
- Cañamero, L., Avila-Garcia, O., and Hafner, E. (2002). First experiments relating behavior selection architectures to environmental complexity. *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, 3:3024–3029 vol.3.
- Clarke, A. S., Soto, A., Bergholz, T., and Schneider, M. L. (1996). Maternal gestational stress alters adaptive and social behavior in adolescent rhesus monkey offspring. *Infant Behavior and Development*, 19(4):451 – 461.
- Federici, D. and Downing, K. (2006). Evolution and development of a multicellular organism: Scalability, resilience, and neutral complexification. *Artif. Life*, 12(3):381–409.
- Floreano, D. (2005). Evolutionary robotics. A tutorial.
- Ideta, A., Hayama, K., Kawashima, C., Urakawa, M., Miyamoto, A., and Aoyagi, Y. (2009). Subjecting holstein heifers to stress during the follicular phase following superovulatory treatment may increase the female sex ratio of embryos. *J Reprod Dev*.
- Jablonka, E. and Lamb, M. J. (2006). *Evolution in Four Dimensions : Genetic, Epigenetic, Behavioral, and Symbolic Variation in the History of Life*. The MIT Press.
- Laplante, D. P., Brunet, A., Schmitz, N., Ciampi, A., and King, S. (2008). Project ice storm: Prenatal maternal stress affects cognitive and linguistic functioning in 51/2-year-old children. *Journal of the American Academy of Child and Adolescent Psychiatry*, 47(9):1063–1072.
- Mastorci, F., Vicentini, M., Viltart, O., Manghi, M., Graiani, G., Quaini, F., Meerlo, P., Nalivaiko, E., Maccari, S., and Sgoifo, A. (2009). Long-term effects of prenatal stress: Changes in adult cardiovascular regulation and sensitivity to stress. *Neuroscience and Biobehavioral Reviews*, 33(2):191–203.
- Miorandi, D. and Yamamoto, L. (2008). Evolutionary and embryogenic approaches to autonomic systems. In *ValueTools '08: Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools*, pages 1–12, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- Oros, N., Steuber, V., Davey, N., Canamero, L., and Adams, R. (2009). Evolution of bilateral symmetry in agents controlled by spiking neural networks. In *Artificial Life, 2009. ALife '09. IEEE Symposium on*, pages 116–123.
- Roggen, D., Federici, D., and Floreano, D. (2007). Evolutionary morphogenesis for multi-cellular systems. *Genetic Programming and Evolvable Machines*, 8(1):61–96.
- Sipper, M., Sanchez, E., Mange, D., Tomassini, M., Perez-Uribe, A., and Stauffer, A. (1997). A phylogenetic, ontogenetic, and epigenetic view of bio-inspired hardware systems. *Evolutionary Computation, IEEE Transactions on*, 1(1):83–97.
- Tanev, I. and Yuta, K. (2008). Epigenetic programming: Genetic programming incorporating epigenetic learning through modification of histones. *Inf. Sci.*, 178(23):4469–4481.

Retro-projected faces effectiveness on gaze reading

Frédéric Delaunay Joachim de Greeff Tony Belpaeme

School of Computing and Maths
University of Plymouth, UK
Email: frederic.delaunay@plymouth.ac.uk

Introduction. Gaze reading is an essential part of HRI as it supports, among others, joint attention and non-linguistic interaction. While most work has focused on implementing gaze direction reading on a robot, little is known about how a human partner is able to read gaze direction from a robotic face.

Cognitive psychology shows how gaze direction reading is essential in joint visual attention (Langton et al., 2000) or how gaze avoidance is used in social communication (McCarthy et al., 2006).

This suggests that it is not only important to have machines that can read gaze direction, but that robots should be able to accurately display gazing behaviour that can be correctly interpreted by human users. We questioned the influence of the physiognomy of an agent's face and eyes on the user's ability to infer where it is looking. Thus, we performed a series of experiments in which human participants were asked to infer the gaze direction of four different types of faces as seen on fig 1 from two different viewpoints.

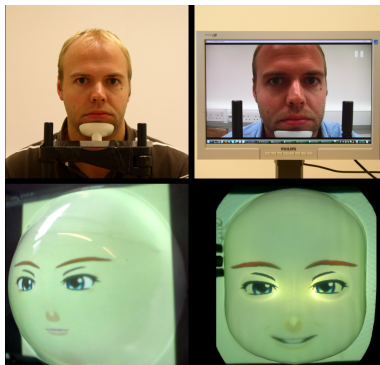


Figure 1: The 4 display-faces used in the experiment.

A real human face served as the null-hypothesis. We assume that a real human face works best for assessing gaze direction. Next to this, we evaluate three artificial faces. The first one is a recording of the human face displayed on a monitor; this condition serves to assess in how far the lack of 3D structure influences gaze reading. The second is a robotic face implemented as a back-projected 3D face; this is a new technology currently being explored in HRI (Delaunay et al., 2009) which is cheaper and more

flexible than existing facial animation technologies. The third is the same robotic face, this time projected into a semisphere; this serves to evaluate the technology of (Hashimoto and Kondo, 2007), who evaluated a similar robotic setup.

Methods and Results. In this process, we considered two points: as several contributions about 3D avatars displayed on flat screens have been published (Picot et al., 2007) we focus on the effect of flat screens on human gaze-guidance; and given that adult face proportions don't fit the dome and mask displays, these combinations would probably lead to the uncanny valley effect because of a squeezed face.

The viewpoint effect was explored by placing the participants either directly in front of the display-face (0 degrees condition), or at an angle of 45 degrees on the right (45 degrees condition). Twenty four participants participated in a sequence of four sessions. This yielded 96 records, which gives 12 records for each condition. To account for any habituation effect, i.e. performance increase of participants over the sessions of a sequence, we shuffled the order of sessions for a pair of participants. This way it was ensured that the number of times a display face would be experienced 1st, 2nd, 3rd or 4th was equal.

Between the participants and the display-face there was a transparent grid of 50x50 cm, evenly divided in 100 squares each displaying a number from the sequence 0 to 99 from top left to bottom right (center area of the grid are the numbers 44, 45, 54 and 55). The grid stands upright between the participants and the evaluated face so that the distance from eyes of the face to the numbers of the grid would increase evenly from the center of the grid. The position and size of the grid also ensured eyelids could not hinder the interpretation of gaze direction when gazing at the bottom of the grid (number 70 to 99).

A single session consisted of the display-face looking at a sequence of 50 randomly generated numbers, switching to the next one after a fixed delay. As numbers are pseudo-randomly generated, we instructed the participants that the same number can appear multiple times in a number sequence; allegedly each number would occur fairly over all sequences.

Once a number was gazed at, an auditory signal

was played indicating to the participants that they could perform their observation. A delay of 5 seconds was long enough to give the (human) display-face enough time to find the proper number and for the participants to write down their observations afterward. When the display-face was a human (one of the examiners), the number sequence was played over earphones worn by the examiner so it could not be heard by the participants. In the case of the video, the display-face consisted of a prerecorded sequence of the same examiner looking at a number sequence. In the two cases of the animated faces, the number sequence was generated on the fly and fed into the animated face control module. The same auditory signal was played once the display-face was looking at the next number to ensure consistency among sessions.

Distance between the participants' written sequence of numbers and actual sequence was calculated as follows: horizontal and vertical errors were counted separately, with each cell in either horizontal or vertical direction counting as 1 while diagonal errors were given a factor of 1.5.

A first confirmation of our expectations is that all participants performed best at guessing human gaze (the control) and also dramatically above chance for all other faces.

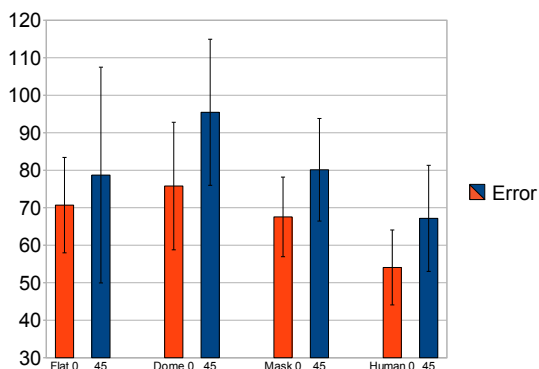


Figure 2: Mean errors per display from 2 angles.

When examining the difference in performance between the two different viewpoints, it is clear that it is much easier for participants to determine the gaze direction when they are facing the face, as opposed to a side view at 45 degrees (Figure). This difference between viewpoints was significant for the human, mask and dome, but not for the flat screen, due to the large variance in performance.

Finally, participants were asked to subjectively rate their experience (using a 7-point Likert scale) in terms of effectiveness. We asked them to describe how effective they found each of the four different faces in conveying information about gaze direction.

Results indicate participants find the human to be the most effective in terms of gaze information, followed by mask, flat-screen and dome. The difference between human and all other faces is significant, as is the difference between mask and dome, however it is not between flat and mask nor flat and dome.

Discussion. As expected, inferring gaze direction from a real human is easiest and most accurate. Overall though, it can be concluded that a 3D mask with a projected animated face embodies a reasonably setup for which participants are still rather apt at inferring the gaze direction. We hypothesize that although an animated face is missing some human characteristics, and hence this may impair the ability of participants to infer it's gazing direction, the 3D structure of the mask counters this effect. This is reflected in the fact that performance for the dome is significantly lower.

Comparing the mask and the flat-screen video, participants perform more or less equally well (difference in performance is not significant). A flat-screen video of a human face is also relatively well interpreted, although especially seen from the side the variance in performance is rather large.

References

- DeBoer, M. and Boxer, A. M. (1979). Signal functions of infant facial expression and gaze direction during mother-infant face-to-face play. *50(4):1215–1218*.
- Delaunay, F., de Greeff, J., and Belpaeme, T. (2009). Towards retro-projected robot faces: an alternative to mechatronic and android faces. In *Proceedings of the IEEE Ro-Man 2009 conference, Toyama, Japan*. IEEE.
- Hashimoto, M. and Kondo, H. (2007). Effect of emotional expression to gaze guidance using a face robot. In Tamatsu, Y., (Ed.), *Proceedings of the 17th IEEE International Symposium on Robot Human Interactive Communication (Ro-Man 2008)*, page 95101.
- Langton, S. R. H., Watt, R. J., and Bruce, V. (2000). Do the eyes have it? cues to the direction of social attention. *Trends in Cognitive Sciences*, 4(2):50 – 59.
- McCarthy, A., Lee, K., Itakura, S., and Muir, D. W. (2006). Cultural display rules drive eye gaze during thinking. *Journal of Cross-Cultural Psychology*, 37(6):717–722.
- Picot, A., Bailly, G., Elisei, F., and Raidt, S. (2007). Scrutinizing natural scenes: Controlling the gaze of an embodied conversational agent. In *IVA*, pages 272–282.

How internal modeling arises when ‘the world is not enough’: an evolutionary robotics study

Onofrio Gigliotta* Giovanni Pezzulo** Stefano Nolfi*

*Istituto di Scienze e Tecnologie della Cognizione - CNR
Via S.Martino della Battaglia, 44 - 00185 Rome, Italy

**Istituto di Linguistica Computazionale “Antonio Zampolli” - CNR
Via Giuseppe Moruzzi, 1 - 56124 Pisa, Italy

Abstract

The aim of this study is showing that a simulated robot trained in a navigation task with a genetic algorithm can develop an *internal model*, and rely on it to fulfill the same task adaptively even in (partial) absence of external stimuli, or when the robot is temporarily ‘blindfold’. We found that evolved internal models have dynamical and anticipatory aspects. In our experiments the key condition is unreliability in sensory stimulation¹.

1. Introduction

The idea that cognitive agents act on the basis of internal models of their tasks instead than purely on the basis of the stimuli they receive from the external environment can be considered foundational in cognitive science. The structure and functioning of internal models is however much more debated.

Recently, after years of little interest culminating with Rodney Brooks’ claim that “the world is its own best model”, the idea of internal modeling is gaining consensus anew, as numerous researchers in cognitive psychology, neuroscience, and robotics have (re)integrated the ideas of internal modeling and representation in an ‘embodied’ view of cognition loosing their classical symbolic centered status and, at the same time, emphasizing that *anticipation* is a key element of internal models’s functioning (Grush, 2004; Wolpert et al., 1995). However, it is less clear why and how did internal models originate. To tackle this problem, in this paper we adopt an *evolutionary robotics* methodology (Nolfi and Floreano, 2000), which permits to verify whether an internal model can evolve and eventually which are the prerequisites for its evolution.

The primary goal of this paper is to investigate whether artificial embodied agents, that are trained

for the ability to exhibit a given behavioral skill, develop and use an internal model that allow them to anticipate forthcoming stimuli to overcome the problems caused by the fact that sensory stimulation is incomplete or noisy. The work described in this paper represents one of the first attempts to demonstrate this hypothesis experimentally (after the pioneering study of Ziemke et al. 2005) and the first demonstration that an internal model can indeed arise spontaneously without being rewarded or incorporated explicitly in the system.

2. Methods, scenario, and results

To test our hypothesis, we set up an experimental scenario in which an embodied and situated agents should develop an ability to display a simple behaviour and keep producing it also when the sensory information is temporarily missing.

The agents consists of a simulated eye provided with a single photoreceptor located in front of a screen showing an 500x500 pixel image generated by the combination of a blue and red gradient ranging from 0 to 1 along the left-right and the top-down dimensions, respectively. Each time step, the photoreceptor detects the intensity of the blue and red in the pixel corresponding to the current position of the eye. The agent is also provided with two motors that allow it to move left-right and/or top-down, with respect to its current position, up to a maximum of ± 5 pixels along each axis.

The task of the agent is that to navigate on the image by turning around the center of the image. For the purpose of measuring agent’s ability to exhibit such behaviour, the image has been ideally divided into 36 sectors located around its centre.

The agent’s controller consists of an artificial neural network (see fig. 1) with two sensory neurons, eight internal neurons, two motor neurons, and two additional output neurons that are used to set the state of the sensory neurons when visual information is missing. The two motor neurons (M1, M2) determine the amplitude of the eye movement along the

¹Research supported by the EU’s 7th FP, grant agreements ITALK (ICT-214668) and HUMANOBS (ICT-231453).

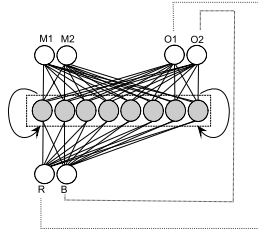


Figure 1: The architecture of agents' neural controller.

left-right and top-down dimension within a range of $[-5,5]$ pixels. Sensory inputs are simply rely units, internal neurons are leaky integrators motor neurons are standard logistic neurons activated.

The architecture of the neural network is fixed. The connection weights and biases and the time constant of the internal neurons are encoded in free parameters and evolved.

During the evolutionary process each individual is tested for 20 trials. At the beginning of each trial the eye is placed randomly in one of ten possible positions around the center of the image. The agent is then allowed to interact with the environment up to 4000 time steps. For each trial the agent experiences a succession of phases in which sensory information is available (normal phases), and phases in which it is missing (blind phase).

During all the normal phases, the state of the two sensory neurons is set on the basis of the colour of the current portion of the image perceived by the agent. During all the blind phases, the state of the two sensory neurons is set on the basis of the state of the additional output neurons (O1 and O2) at time $t-1$. The performance (fitness) of the individual has been evaluated by computing the number of subsequent sectors of the image visited by the eye.

By analysing the behaviour of evolved individuals we observed that in 17 out of 40 replications of the experiment, the best individual succeed in circling around the centre of the image both in normal and blind phases during which sensory stimulation is temporary missing and in which the state of R and B sensory neuron is replaced with the state of the two additional motor neurons O1 and O2. These individuals manage to compensate the lack of sensory information by self-sustaining their internal dynamic in two different ways. Agents belonging to the first 'family' (13 out of 17) solve the problem by developing two qualitatively different strategies for normal and blind phases, and trigger the first or the second strategies during the two corresponding phases. Interestingly, although almost all these agents anticipate incoming stimuli during the normal phases with their neurons O1 and O2 (anticipation measured through a cross-correlation analysis), their dynamics are different during the blind phases (like in Ziemke et al., 2005). Agents belonging to

the second 'family' (4 out of 17), instead, keep reacting to the experienced sensory states in similar ways during normal and blind phases and compensate the lack of sensory information with the self-generation of equivalent information and by anticipating how the state of the sensors would vary as a result of the execution of the planned action. That is, the agents use a predictive strategy based on internal modeling. Agents belonging to the second 'family' are, on average, more effective than agents belonging to the first family.

3. Conclusions

The central hypothesis that motivated our design methodology is that a (temporary) deprivation of external stimuli can make it favorable, from an evolutionary perspective, the development of a robot's internal model even in absence of any explicit reward for prediction. Indeed, once the robot has learned a reliable behavioral strategy and an associated dynamical representation of its task, it could be favorable to maintain the same strategy, and at the same time learning to self-maintain the same dynamics via self-generated (predicted) inputs, rather than evolving two separated strategies to deal with the presence or absence of external stimuli. Our experimental results show that, under opportune environmental conditions, a robot can spontaneously develop an internal model that has anticipatory aspects, can be (temporarily) detached from the current sensorimotor flow, and endogenously reactivated by self-generated signals. This result supports the idea that internal modeling capabilities could have arose for these reasons, and not for cognition, although they could have been successively exapted for increasingly complex cognitive uses (Grush, 2004; Pezzulo and Castelfranchi, 2007).

References

- Grush, R. (2004). The emulation theory of representation: motor control, imagery, and perception. *Behavioral and Brain Sciences*, 27(3):377–96.
- Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics*. MIT Press.
- Pezzulo, G. and Castelfranchi, C. (2007). The symbol detachment problem. *Cognitive Processing*, 8(2):115–131.
- Wolpert, D. M., Gharamani, Z., and Jordan, M. (1995). An internal model for sensorimotor integration. *Science*, 269:1179–1182.
- Ziemke, T., Jirnhed, D.-A., and Hesslow, G. (2005). Internal simulation of perception: a minimal neuro-robotic model. *Neurocomputing*, 68:85–104.

Experimental setup for studying the development of tool-use on the example of object throwing

Verena V. Hafner

Cognitive Robotics Group
Department of Computer Science
Humboldt-University Berlin, Germany
hafner@informatik.hu-berlin.de

Werner Sommer

Cognitive Psychophysiology Group
Department of Psychology
Humboldt-University Berlin, Germany
sommer@cms.hu-berlin.de

Abstract

In this paper, we present an experimental setup currently built at Humboldt-University Berlin where various research questions aimed at understanding the development of throwing skills can be answered. Both EEG data and the physical motion of the thrower are measured and visual feedback is provided.

1. Introduction

Throwing¹ is a difficult skill that involves a precise interplay of sensory information and motor commands. Throwing an object at a moving target in particular requires an exact eye-muscle coordination, as well as the coordination of shoulder, elbow, wrist and finger muscles. Another difficulty is the precision of the timing. The launch window for an accurate throw is only 1 to 10 ms wide (Calvin, 1983).

Investigating into the underlying processes for the development of throwing skills is both interesting from an evolutionary and from a developmental point of view. Throwing is possible by just using hand and arm without any specialised tool. Other forms related to throwing that require tool-use are archery, slingshot, or shooting. The earliest dedicated throwing tools in human history can be dated back to the paleolithic (Rhodes and Churchill, 2009). Many animal species show signs of throwing and other tool-use (Westergaard et al., 2000).

During human infant development, throwing appears shortly after the skill of reaching and grasping at around the age of 12 months (Piaget, 1962). A skilled hunter, however, can improve his learning skills until the age of about 40 years. This shows that aiming at a target requires much more than just physical force, and a large amount of experience is

¹By “throwing”, we mean aiming at a target during the throw, and not only launching an object. Even a long-throw fulfils this prerequisite since the angle at the point of release as well as the speed determine the point of collision of the object with the target.

necessary for this coordinated sensory-motor interaction. Calvin even argues that the development of throwing skills is a possible forerunner for the development of speech (Calvin, 1993) because it fostered the evolution of planning ahead.

We will present here a technical solution for an experimental setup respecting the complex physiological and sensorimotor prerequisites to study the development of throwing skills. One of the questions to be answered is what can be learned by pure self-supervised learning or learning by trial-and-error, and what can be learned by observing a skilled thrower (Demiris and Billard, 2006).

2. Experimental Setup

The experimental setup for the throwing experiments is derived in close cooperation between the cognitive robotics group and the cognitive psychophysiology groups at Humboldt-University Berlin.

In order to extract the relevant information and to provide the thrower with sensory feedback on his or her throw, the following data have to be collected:

- EEG data synchronised with other information
- Motion trajectories and acceleration of the object during the throw and other positions on the body
- Time of fixation of the target
- Time of starting the throw
- Time of releasing the object.

The following feedback and information need to be provided to the thrower, either as tactile or visual feedback:

- weight and size of the object
- size of and distance to the target
- trajectory of the flying object
- point of collision of the object

Simultaneously with the motion and feedback information, event-related brain processes supporting the motor action and cognition will be studied by means of EEG analysis. The EEG will be registered from 64 active electrodes (actiCAPs), band-pass filtered from DC to 100Hz and sampled at 1000Hz. EEG dynamics may provide insights into the brain's

central role in optimising the outcome of complex behaviour (Makeig et al., 2009) such as object throwing. By applying ICA, the EEG data can be separated into brain processes and non-brain artifacts (e.g., movement-, electromyographic and eye-blinks artifacts).

For the measurement of the motion, a sensor derived from Nintendo's *WiiMote* controller is used as our prototype. This choice of sensor hardware has several advantages. The *WiiMote* has a 3-axis accelerometer, an infrared camera, several buttons as well as vibrational and sound feedback. The data can be read and sent via Bluetooth at a rate of approx. 100Hz using open source drivers. The feasibility of using the *WiiMote* as a tool for robotics experiments has been shown in our previous research on behaviour recognition (Hafner and Bachmann, 2008). Only recently (June 2009), an additional sensor (*Wii MotionPlus*) was released that can be connected to the *WiiMote* and measures the rotational acceleration in three axes.



Figure 1: The sensor equipment for the experiments is partly derived from a Nintendo *WiiMote*.

The subject throws a real object towards a target that is presented on a projection screen, but as soon as the object leaves the hand (measured by using a *WiiMote* button release) it will be diverted and simulated instead. The visual feedback derived from the acceleration data of the throw is presented to the thrower on screen in real time.

3. Research Questions

The following research questions will be addressed using the presented experimental setup.

- Does the EEG activity reflect the brain's control of throwing, are there differences between different natural movements such as high-precision and low precision throwing, complex movement sequences, or simple movements (e.g. button presses)?
- How do phases of activity look like during pre-movement, post-movement, and effect monitoring?
- Are there differences in brain activity for skilled throwers and novices, men and women, long-distance and short distance targets, light and heavy objects?
- Does the result of the previous throw influence

the performance of the current throw?

- Can the parameters be extracted and learned to create a skilled robotic thrower?
- What are the differences in self-supervised learning and learning from observation or demonstration in a throwing task?

4. Future Work

The features and physiological properties of a skilled thrower derived with the above setup can be tested in a robotic setup with repeatable experiments and real physics (in comparison to a simulation). Variable parameters are temporal resolution, force, weight of the object, and precision of the motors. In these experiments, we would also like to investigate if the throw can be based on experience about the sensorimotor interplay or whether an internal simulation of a physical model is required in the robot.

References

- Calvin, W. (1983). A stones throw and its launch window - timing precision and its implications for language and hominid brains. *Journal of Theoretical Biology*, 104:121–135.
- Calvin, W. (1993). The unitary hypothesis - a common neural circuitry for novel manipulations, language, plan-ahead, and throwing. *Tools, Language and Cognition in Human Evolution*, pages 230–250.
- Demiris, Y. and Billard, A. (2006). *Special Issue on Robot Learning by Observation, Demonstration and Imitation*. IEEE Transaction on Systems, Man and Cybernetics.
- Hafner, V. and Bachmann, F. (2008). Human-humanoid walking gait recognition. In *Proceedings of Humanoids 2008, 8th IEEE-RAS International Conference on Humanoid Robots*, pages 598–602.
- Makeig, S., Gramann, K., Jung, T.-P., Sejnowski, T., and Poizner, H. (2009). Linking brain, mind and behavior. *International Journal of Psychophysiology*, 73:95–100.
- Piaget, J. (1962). *Play, dreams and imitation in childhood*. Norton Press, New York.
- Rhodes, J. A. and Churchill, S. E. (2009). Throwing in the middle and upper paleolithic: inferences from an analysis of humeral retroversion. *Journal of Human Evolution*, 56:1–10.
- Westergaard, G. C., Liv, C., Haynie, M. K., and Suomi, S. J. (2000). A comparative study of aimed throwing by monkeys and humans. *Neuropsychologia*, pages 1511–1517.

Learning Affective Landmarks

Antoine Hiolle and Lola Cañamero

Adaptive Systems Research Group

School of Computer Science & STRI

University of Hertfordshire

College Lane, Hatfield, Herts AL10 9AB, UK

{A.Hiolle,L.Canamero}@herts.ac.uk

Abstract

This poster presents early work on the effects of arousal and its regulation on learning about the environment, particularly affective memories associated with places that can be used to safely guide exploration.

1. Introduction

As part of our ongoing research on the development of attachment bonds between “baby” autonomous robots and human caregivers, which has previously investigated, among other aspects, the role of the caregiver in arousal regulation (Hiolle and Cañamero, 2008), this paper presents the first steps of a longer-term study regarding the effects of arousal and arousal regulation—one of the key elements in the development of emotional intelligence—on early learning of the environment. In particular, arousal modulation is used to associate simple affective experiences to the memory of scenes and places. This is an important aspect of cognitive development in infants, and in our case it permits the robot to easily learn the relevance of novel scenes and places—for example to be able to recall and find places of interest in terms of learning—where it was able to discover sensorimotor associations—and places of “comfort”—associated to its human caregiver. This is important to develop “safe” exploratory behavior alternating between locations of interest and a “secure base”.

2. Architecture

The architecture depicted in Figure 1 permits the robot recall and recognize several views of a scene associated with one of the following events: (1) the presence of the human caregiver; (2) a moderate level of predictability in the perceptions of the robot; or (3) a high level of unpredictability in its perceptions. The robot thus divides locations into three categories: places where a human caretaker has interacted (visually or using contact) with it, those where it has found interesting and predictable features to

learn, and those where features were unpredictable. This learning is mediated by the level of arousal of the robot—a variable indicating the overall level of internal activity: when the robot maintains a moderate arousal level, it will stop and associate visible landmarks with this state. Let us briefly consider the different elements of the architecture.

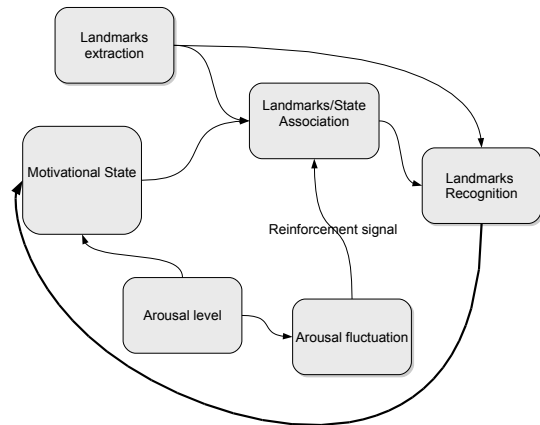


Figure 1: Architecture for learning affective landmarks.

Arousal level. The level of arousal is here a function of the sensorimotor learning system. This learning system consists of a simple sensorimotor coupling between the position of the joints and the current feature of the image from the robot’s camera, computed using the landmark extraction system described below. This process is decoupled from the landmark/state association one, i.e., it is stored in another group of neurons. Its purpose is to assess the stability and predictability of the location the robot is at, and this information is fed onto the arousal system. The prediction of the sensorimotor system—where the robot “believes” to be—is compared to the actual set of landmarks observed, and the perceptual error provides input to calculate the arousal, which varies as a function of the error. This means that the arousal increases when the robot discovers a new scene, and then decreases when the robot has learned the correct association. These fluctuations of

the arousal are then used to trigger the learning of a scene or place. Interaction with the human caregiver has a soothing effect on the level of arousal, which decreases exponentially when s/he “comforts” the robot either by touching a contact sensor or by appearing in its visual field; that scene would thus be learned as a “place of low arousal”.

Extraction of landmarks. To extract the landmarks that the robot will in turn learn, we first apply a Sobel filter over the gray-scale image received from the robot. The gradient image is then filtered with a Difference of Gaussian kernel in order to extract low resolution focus points. Then, a log-polar transform of a small size image centered on the focus point is used to code for the landmark. This method has been inspired by (Giovannangeli et al., 2006).

Landmark learning. For the robot to be able to learn the landmarks, they are projected onto a layer of neurons, the activities of which correspond to the intensities of the pixels of the landmark. The landmarks are then associated with the current state using a one-shot Hebbian rule modulated by the fluctuations of the arousal as follows: $w_{ij} = F_{Ar} \cdot I_{ik} \cdot S_j$, where w_{ij} is the weight between the state j and a pixel k with intensity I_{ik} , F_{Ar} is the absolute value of the first derivative—fluctuation of the arousal—and S_i is the activity of the neuron representing the state i . It is important to note that only one set of landmarks can be associated with one state, giving a one-slot memory per state to the robot.

State and Action Selection. The possible states S_i of the robot are either low arousal, medium arousal or high arousal based on predefined thresholds. A state of low arousal in a location can result from the fact that the human caregiver was detected, or that s/he touched the contact sensors of the robot, or that the landmarks present are highly predictable, indicating that the perceptual variability is very low. If the arousal fluctuation is high, the robot will look for stable landmarks by turning its head slowly, and if the visual scene becomes stable enough, the landmarks will be learned. In order to avoid blocking the robot in a position where it will wait for stable landmarks, the arousal fluctuation F_{Ar} decays at every time step. Once the learning of the scene has been achieved, the next state i is chosen by switching to the next upper state in a circular manner (low \rightarrow medium, medium \rightarrow high, high \rightarrow low). If a location has been learned already for this state, the robot will trigger a search behavior, until the recognition is achieved. If no landmarks have been associated with this state, the robot’s default behavior—exploring in a random walk—starts.

3. Results and Discussion

We tested the architecture using a humanoid Nao robot from Aldebaran Robotics. In a small un-

populated (apart from the experimenter) room, we used 5 landmarks for each location, extracted from a 160×120 -pixel image. The algorithm presented here does permit the robot to move between locations of known arousal outcomes. It functions adequately in a simple setting where a human is alone with the robot, where the robot will eventually discover, for example, the location of a play mat and the location of a human caregiver. Although the robot only remembers one set of landmarks—one location—for each state, this seems sufficient to model very early stages of baby-caregiver interaction since, given that the interventions and appearances of the human are continually updated, the robot has a way to retrieve its “secure base” provided by the caregiver (Bowlby, 1969). However, relying on stable landmarks in the scene could cause problems for more complex interactions and environments.

4. Future Work

This early architecture could be enhanced in a number of ways. For example, to improve the robustness of landmark recognition, the robot could associate the orientation of its head as additional information for coding the landmark, and this would allow it to choose the angle that provides the best accuracy when perceiving the landmarks. We would also like to add a learning system which allows the robot to learn the action leading from one state to another, avoiding the problem of “blind exploration” when trying to retrieve a location. Finally, we would like to extend the architecture to permit the robot differentially associate landmarks with individual caregivers as a function of their interaction history.

Acknowledgements

This research is supported by the European Commission as part of the FEELIX GROWING project (www.feelix-growing.org, FP6 IST-045169).

References

- Bowlby, J. (1969). *Attachment and loss*, vol. 1: Attachment. New York : Basics Books.
- Giovannangeli, C., Gaussier, P., and Banquet, J. (2006). Robustness of visual place cells in dynamic indoor and outdoor environment. *Int. J. Advanced Robotic Systems*, 3(2):115–124.
- Hiolle, A. and Cañamero, L. (2008). Consentious caretaking for autonomous robots. In Schlesinger, M., Berthouze, I., & Balkenius, C., eds., *Proc. 8th Intl. Conf. Epigenetic Robotics*, pp. 45–52. Lund Univ. Cognitive Studies.

Implementing inhibition of return; embodied visual memory for robotic systems

Martin Hülse

Sebastian McBride

Mark Lee

Dept. of Computer Science, Aberystwyth University, SY23 3DB, Wales, UK

Abstract

Based on the biological phenomenon of inhibition of return, we introduce an architecture developed for an active robotic vision system where continually updated global information is used to modulate the action selection process for saccadic camera movements. This facilitates, in an extremely efficient way, the fundamental process of avoiding re-saccading to objects previously visited and, thus, is considered to have a wide-ranging application within active vision systems.

Inhibition of return (IOR) refers to the suppression of stimuli (objects and events) processing where those stimuli have previously (and recently) been the focus of spatial attention (Lupianez et al., 2006). In this sense, it forms the basis of attentional (and thus visual) bias towards novel objects. Although the neural mechanism underpinning IOR is not completely understood, it is well established that the dorsal frontoparietal network, including frontal eye fields (FEF) and superior parietal cortex are the primary structures mediating its control (Mayer et al., 2004). These are some of the many modulatory and affecting structures of the deep superior colliculus (optic tectum in non-mammals), the primary motor structure controlling saccade. Although visual information from the retina starts at the superficial superior colliculus, and there are direct connections between the superficial and deep layers, the former cannot elicit saccade directly (Stein and Meredith, 1991). This information has to be subsequently processed by a number of cortical and sub-cortical structures that place it in context of 1) attentional bias within egocentric saliency maps (posterior parietal cortex) (Gottlieb, 2007), 2) the aforementioned IOR inputs from other modalities (Stein et al., 2002), 3) overriding voluntary saccades (frontal eye fields) (Stein et al., 2002) and 4) basal ganglia action selection (McHaffie et al., 2005). Thus, biologically there exists a highly developed, context specific method for facilitating the most appropriate saccade as a form of attention selection. All of the above saccade-affecting attributes have valuable robotic application but inhibition of return is potentially the most useful in the earlier stages of constructing a saccade system that is attention rather than visual-input driven. For example,

within the most basic of active vision system tasks where static objects of the same shape and color are systematically saccaded to (i.e. brought to the centre of image), there is a consistent need for a mechanism whereby objects already scanned are ignored (i.e. inhibition of return). The primary issue here is that similar image data can emerge in very different image locations, thus the only way of knowing whether an image feature has previously been saccaded to or not, is to store that information at the global level. In the following we introduce an architecture developed for a robotic active vision system where that architecture enables the system to integrate and update global information which can in turn modulate the action selection process for saccadic camera movements.

The active vision system consists of two cameras (both provide RGB 1032x778 image data) mounted on a motorised pan-tilt-vergence unit. Three degrees of freedom (DOF) are used: one verge movement for each camera and one tilt which moves both cameras. Each motor is controlled by determining its position in radians (*rad*) where the state of the active vision system is fully determined by the motor positions of the tilt, left and right verge axis, $(p_{tilt}, p_{vL}, p_{vR})$.

The overall computational architecture is illustrated in Figure 1. It consists of three main parts implementing: 1) filtering image data, 2) action selection and execution and 3) the operation of the visual memory. The latter is the central feature of this architecture and main objective of this paper. Without the visual memory, action selection and the resulting saccadic eye movements are determined solely by the current retina image data. Hence, similar visual inputs (RGB image) lead to the same saccade, no matter how often this specific saccade has been executed before. With a visual memory in place, however, specific motor positions $(p_{tilt}, p_{vL}, p_{vR})$ resulting from a successful saccadic camera movement can be stored. This information can then be used to merge the camera image data with the data representing the items present in the visual memory (i.e. those previously saccaded to). The inhibition of return process can then be simply carried out by subtracting the latter from the former, essentially transforming the original camera input into a 'retina-based saliency map' where, objects in the visual memory have been inhibited leaving unsaccaded objects to compete for

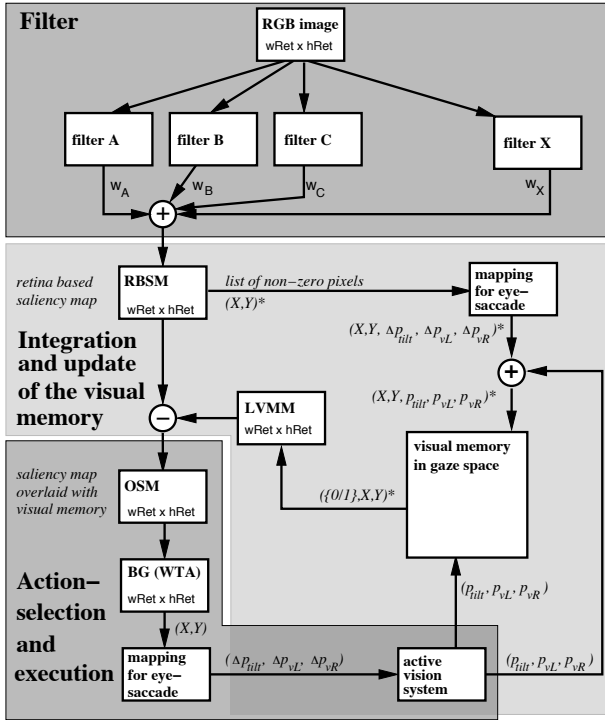


Figure 1: Architecture for embodied visual memory.

the next saccade. In the following the core function of this architecture shall be described in more detail.

A visual buffer (*local visual memory map* or LVMM), and the mapping for the saccadic eye movement (*retina based saliency map* or RBSM) are the essential elements necessary to create the so called *overlaid saliency map* (OSM), see Figure 1. The OSM then feeds into an action selection process: *Basal Ganglia*, (BG). The LVMM represents stimuli which have corresponding entries in the visual memory. The creation of the LVMM is, thus, a crucial part of the architecture. This process starts with RBSM where, for each no-zero pixel in RBSM, the corresponding Δ values (Δp_{tilt} , Δp_{vL} , Δp_{vR}) are derived. These Δ values are learnt beforehand through a mapping process previously described (Lee et al., 2007). Hence, for each non-zero pixel in RBSM we get the relative motor positions (Δp_{tilt} , Δp_{vL} , Δp_{vR}) which drives the particular pixel into the image center. The result of this step is stored as a list where each entry is written as: $(X, Y, \Delta p_{tilt}, \Delta p_{vL}, \Delta p_{vR})$. Notice, in Figure 1 an asterisk signifies a list. Adding these Δ -values to the current absolute motor positions (p_{tilt} , p_{vL} , p_{vR}) provided by the active vision system delivers the final absolute motor positions of the active vision system if a saccade movement was executed. This is again represented as a list: $(X, Y, p_{tilt}, p_{vL}, p_{vR})$. Thus, the Δ -values are replaced by the final absolute motor positions. With this global information the system can now easily ask if a specific pixel (X, Y) in the current RBSM has a corresponding item in the visual memory. If the derived absolute motor positions of pixel (X, Y) can be found in the visual memory

then this pixel is labelled with value of 1 otherwise it is labeled as 0. Thus, all list entries appear as: $(X, Y, \{0, 1\})$. From this list we can then create the LVMM which has the same dimensions as RBSM. Since LVMM contains all previously saccaded to pixels (value 1.0), subtraction from RBSM results in the aforementioned ‘retina-based saliency map’ and an accurate mapping of objects that have not yet been saccaded to.

Although several computational models of inhibition of return of have been put forward e.g. (Alexandre, 2009), an actual robotic implementation of such a process has, until now, not been fully described. In this context, the system provides good real-time performance and, thus, has the potential to be functional within the most demanding of visuomotor paradigms.

Acknowledgment

Thanks for support from EC-FP7 projects IM-CLeVer and ROSSI, and EPSRC, grant EP/C516303/1.

References

- Alexandre, F. (2009). Cortical basis of communication: Local computation, coordination, attention. *Neural Networks*, 22, 126-133.
- Gottlieb, J. (2007). From thought to action: The parietal cortex as a bridge between perception, action, and cognition. *Neuron*, 53, 1, 9-16.
- Lee, M., Meng, Q., and Chao, F. (2007). Developmental learning for autonomous robots. *Robotics and Autonomous Systems*, 55 (9), 750-759.
- Lupianez, J., Klein, R., and Bartolomeo, P. (2006). Inhibition of return: Twenty years after. *Cognitive Neuropsychology*, 23, 7, 1003-1014.
- Mayer, A., Seidenberg, M., Dorflinger, J., and Rao, S. (2004). An event-related fmri study of exogenous orienting: Supporting evidence for the cortical basis of inhibition of return? *Journal of Cognitive Neuroscience*, 16, 7, 1262-1271.
- McHaffie, J., Stanford, T., Stein, B., Coizet, W., and Redgrave, P. (2005). Subcortical loops through the basal ganglia. *Trends in Neurosciences*, 28, 8, 401-407.
- Stein, B. and Meredith, M. (1991). Functional organization of the superior colliculus. In A.G., L., (Ed.), *The neural bases of visual function*, pages 85-100. Macmillan, Hampshire.
- Stein, B., Wallace, M., Stanford, T., and Jiang, W. (2002). Cortex governs multisensory integration in the midbrain. *Neuroscientist*, 8, 4, 306-314.

Distal place recognition based navigation control inspired by Hippocampus - Amygdala interaction

Ansgar Koene*

Gianluca Baldassarre**

Francesco Mannella**

Tony J. Prescott*

*Department of Psychology, University of Sheffield, Sheffield S10 2TP, UK

**Istituto di Scienze e Tecnologie della Cognizione, Consiglio Nazionale delle Ricerche I-00185 Roma, Italy

Abstract

We present a novel robot navigation system based on distal place and value recognition. The navigation control system is inspired by the hippocampus - amygdala circuit that is involved in place learning/recognition and stimulus value association.

1. Introduction

A computational model of the hippocampus - amygdala circuit was developed focusing on the ability to recognize not only the current location where the robot is but also surrounding locations that are currently visible to the robot. This *distal* location recognition relies on the property that, in the absence of occlusions, place defining stimulus configurations change in a gradual manner as the robot moves from one location to another. The difference between the current sensory inputs and the templates associated with known locations therefore increases gradually as a function of distance to the perceived locations. *Distal* recognition of value associated places allows our system to navigate towards goals without exploration of the intermediate space. Further more, navigation behavior naturally becomes contingent upon the stimulus state of the target location (stimulus configuration changes when target light is ON or OFF) providing our controller with added flexibility for dealing with state changes in the environment.

This model was integrated into a robot control system that was previously published in (1) without the new hippocampus and amygdala implementations.

2. Distal place recognition & localization

Figure 1 shows a diagram of the hippocampus model for place recognition. The perceived stimulus configuration forms the sensory input that is matched to heading direction specific templates (i.e. *view cells* (2)). The activity of all *view cells* associated with a particular location is summed in the *distal recog-*

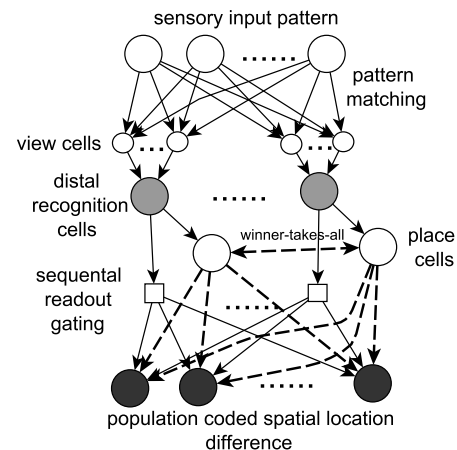


Figure 1: Hippocampus network. Solid: excitatory connections; Dashed lines: inhibitory connections

nition cells where the magnitude of activation indicates the recognition confidence. Winner-takes-all competition between the *distal recognition cell* outputs yields the current location estimate in the *place cells* (3). *Place cells* and *distal recognition cells* are associated with spatial locations by their connectivity to output *grid cells* (4) that use a population code representation of spatial coordinates. By subtracting the current location estimate from the *distal recognition cell* associated location the output encodes the required displacement for reaching distally recognized locations. In order to do this with a single set of *grid cells* however it is necessary to process the *distal recognition cells* sequentially.

3. Place value association

The amygdala provides association of values with basic sensory stimuli (e.g. target lights) or stimulus configurations encoded via hippocampal *distal recognition cells*. Whenever an innately rewarding/punishing input is received any stimulus (configuration) that predicted the reward becomes associated with the rewarding input. In order to trace

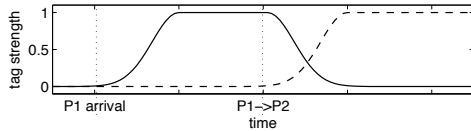


Figure 2: Solid line: *tag* related to place P1; Dotted line: *tag* related to place P2

which stimulus configuration (i.e. *place*) predicted reward delivery we tag *place cell* activations with a signal that gradually increases to a saturation level as long as the *place cell* is active and gradually decreases when *place cell* activation is removed (see figure 2). For basic stimuli tagging is triggered when the perceived stimulus strongly changes (e.g. light goes on or off). For further details, see (5).

4. Experiment

To test our rat brain inspired navigation system we used a *differentially rewarded plus-maze* task (1).

First the robot explored the maze guided simply by attraction to unmapped visible locations. When the first reward location was encountered the corresponding place and sensory stimulus (target light) were associated with the reward value. Subsequently, exploration behavior was overruled by target light approach behavior whenever the robot was able to see target lights. Once all maze arms were visited the robot recognized the valued locations and visited them in order of learned reward magnitude.

The robot produced a sparse map of the environment with a majority of place cells in maze corner areas where small movements dramatically changed the visual inputs. Analysis of the visual pattern templates (view cells) revealed that the values were successfully associated with stimulus configurations where the target object light is on even though each maze arm end is also mapped to a visual configuration where the light is off.

The robot successfully recognized not only its current location in the maze but also produced gradually reducing recognitions for distal locations in the current field of view (figure 3).

5. Conclusion

Distal recognition of value associated places produces flexible navigation without requiring full exploration of the movement space. The resulting navigation behavior is intrinsically contingent upon the stimulus state of the target location enabling the controller to cope with state changes in the environment.

Based on the combination of *distal* place recognition and value association the hippocampus-amygdala network successfully guided the robot towards visible locations that were learned to be most

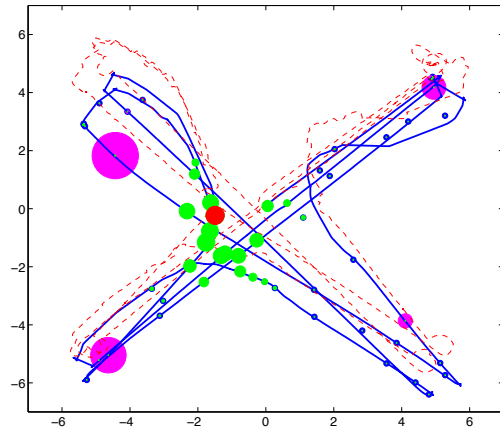


Figure 3: Solid blue line: estimated movement by robot; Dashed red line: actual movement; Red disk: estimate of current location; Blue disks: place cell locations; Green disks: activity of *distal recognition cells* (bigger = more active); Magenta disks: values associated with place cells (bigger = higher value)

rewarding.

Acknowledgements

This work was supported by the European Union Framework 6 IST project 027819 (ICEA project: www.iceaproject.eu).

References

- Koene A., Prescott T.J.: Hippocampus, Amygdala and Basal Ganglia based navigation control. In: Artificial Neural Networks – ICANN 2009. Lecture Notes in Computer Science, 5768, 267-276 (2009)
- Rolls, E.R., Stringer, S.M.: Spatial view cells in the hippocampus, and their idiothetic update based on place and head direction. *Neural Networks* 18(9), 1229-1241 (2005)
- OKeefe, J., Dostrovsky, J.: The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely moving rat. *Brain Research* 34, 171-175 (1971)
- Fyhn, M., Molden, S., Witter, M.P., Moser, E.I., Moser, M.B.: Spatial representation in the entorhinal cortex. *Science* 305, 1258-1264 (2004)
- Mannella F., Koene A., Baldassarre G.: Navigation via Pavlovian Conditioning: A Robotic Bio-Constrained Model of Autoshaping in Rats. Proceedings of the Ninth International Conference on Epigenetic Robotics (2009)

Learning paths as a sequence of sensori-motor associations

M. Lagarde¹, P. Andry¹, P. Gaussier^{1,2}

(1) ETIS, ENSEA, Univ Cergy-pontoise, CNRS UMR 8051 (2) IUF
F-95000 Cergy Pontoise, France
{lagarde,andry,gaussier}@ensea.fr

Our long term goal is to design a control architecture allowing a robot to learn, as autonomously as possible, complex behaviors. A human can achieve a task using different strategies. For example, we can follow a particular path or directly reach a particular position. Both strategies allow to go to a place. This raises a question on the human development : how a child learns a task? What and how to order different strategies? In robotics, a behavior can be learned as a particular trajectory [Calinon and Billard, 2006] or as the reaching of a target (position, object, ...) [Girard et al., 2005]. Most of the time, the learning of the trajectory is highly dependent of the parameters defining the motor dynamics of the robot or the effector(s). Timing, speed, and acceleration matter. Conversely, if the robot has to reach a particular target, then how the learning process anchors this target in the sensory-motor space is of high importance (and the choice of this space also). Here, we present a model trying to merge these two aspects (learning the timing of the trajectory as a sequence of motor transitions [Andry, 2002] or as visuo-motor associations during the moves of the robot). Our model is composed of two sensori-motor loops allowing a robot to learn the temporal and visuo-motor properties of self behavior when guided and corrected by a human experimenter (figure 1). The experiments raise the question of synchronization and action selection between the different responses of the sub-structures.

We have developed a controller for mobile robots which is able to associate visual information (a panorama of the environment) with self orientation (using a compass representing the direction of the actual movement). This controller is designed as a sensori-motor loop based on a neurobiological model testing some of the spatial properties of the hippocampus. Interestingly, fewer researches also highlight the temporal properties of the hippocampal loop and the fact that populations of cells can also learn the timing of the transitions between input events. From these studies we have



Figure 1: Setup allowing a human to teach paths to a mobile robot. The control architecture is designed learn changes in the robot's own motor dynamics (strong differences in the proprioceptive flow of the wheels). Thanks to a leash, the human can pull on a sensing device, thus guiding the robot. The robotic arms are not used in this study.

designed a control architecture allowing a robot to learn complex sequences (i.e. sequences containing many occurrences of the same "unit") of sensori-motor transitions [Lagarde et al., 2007]. Next, it is important to distinguish how a "behavior" can be learned: on one side the different steps can be anchored in the environment. A coding linked to the environment is obtained where each step, each association is dependant of the environment (and the recognition of this environment). On the other side, this behavior can be learned independently of the external environment, "blindly", for example by anchoring the proprioceptive changes according to an internal timeline. Of course, it is interesting to notice that both solutions seem to complete each other: figure 2 is a simplified schema of the global architecture where both dynamics (temporal and spatial) are learned by the same neural network (NN) in two sensori-motor loops.

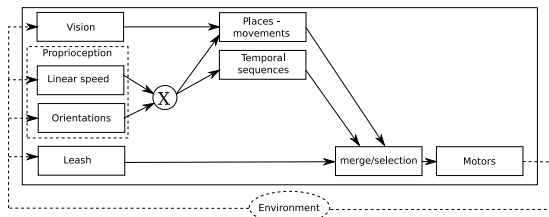


Figure 2: Model of 2 sensori-motor loops (i) place-movement associations learning and (ii) temporal complex sequence learning.

With such a NN, we show that spatial and temporal sensori-motor loops can be complementary. For example (figure 3), during a navigation task, the robot learns the path as a succession of temporal movements in addition to a collection of spatial attractors. This learning proceeds in two steps. First, the robot learns different independent place-movement associations. Second, the robot is kidnapped and put on a place. While the robot follows the trajectory from the succession of place-movement associations, the system learns the timing of its own movements. The result is a robot able to navigate autonomously reactively (spatial attractor) and proactively (temporal prediction). We present an experiment where an encoding can compensate another one. Here, the temporal encoding compensates the place-movements associations when the vision is blocked. This raises questions on the process of action selection. To continue our works on the complementary of both sensori-motor loops, we will study if the place-movement associations and temporal sequences can be fused to strengthen each other and deliver a coherent behavior. Futur work aims at understanding how the different strategies develop and cohabit during the human development. How a robot can self evaluate to select one strategy when the different loops proposes contradictory movement?

Acknowledgments

This work was supported by the French Region Ile de France, the Institut Universitaire de France (IUF) and the FEELIX GROWING european project. (FP6 IST-045169)

References

- Andry, P. (2002). Thèse : Apprentissage et interactions via imitation : application d'une approche développementale à la robotique autonome.
- Calinon, S. and Billard, A. (2006). *Learning of Gestures by Imitation in a Humanoid Robot*. Cambridge University Press, k. dautenhahn and c.l. nehaniv edition. in press.

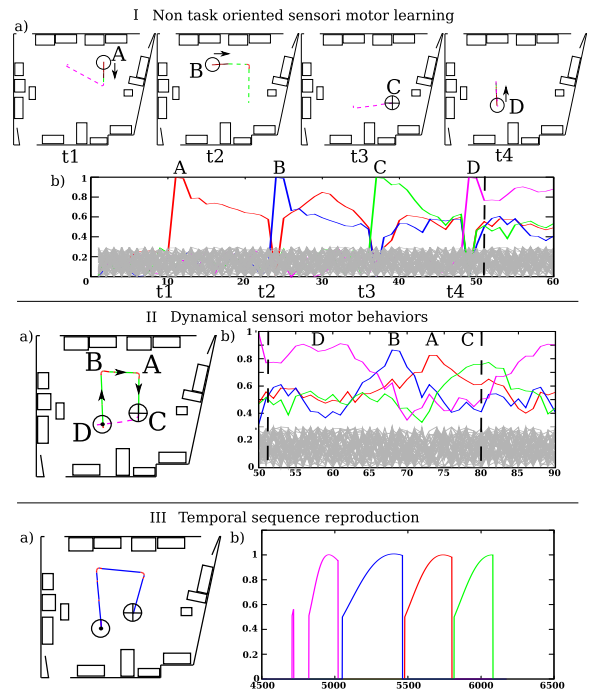


Figure 3: I.a - The operator guides the robot to different places of the room. The robot learns each visual place and associates its current movement - an orientation/linear speed couple (plain arrows). I.b - Activities of the place cells when learning the 4 places ("A", "B", "C" and "D".)

II.a - The robot is taken back to place "D" by the operator. Thanks to the place recognition, the robot predicts and executes the associated movement *arrow up* which leads it to the next recognized place, and so on. At the same time, the system learns the temporal sequence of its orientations/linear speed couple. II.b - The activities of visual places recognized.

III.a - The robot is taken back at the beginning of the temporal sequence by the operator. The operator hides the camera of the robot, consequently the system can not recognize the visual places. Thanks to the learned temporal sequence, the robot predicts the next movement and it reproduces the whole trajectory. III.b - Activities of the temporal predictions.

Girard, Filliat, Meyer, Berthoz, and Guillot (2005). Integration of navigation and action selection functionalities in a computational model of cortico-basal ganglia-thalamo-cortical loops. *Adaptive Behavior*, 13(2):115–130.

Lagarde, M., Andry, P., and Gaussier, P. (2007). The role of internal oscillators for the one-shot learning of complex temporal sequences. In de Sa, J. M., Alexandre, L. A., Duch, W., and Mandic, D., (Eds.), *Artificial Neural Networks – ICANN 2007*, volume 4668 of *LNCS*, pages 934–943. Springer.

Learning to Collaborate by Observation

Stephane Lallee¹, Felix Warneken², Peter Ford Dominey¹

¹*CNRS & INSERM U846, France, peter.dominey@inserm.fr, stephane.lallee@inserm.fr*

²*Max Planck Institute of Evolutionary Anthropology & Harvard University warneken@eva.mpg.de*

Abstract

Human infants display a remarkable capacity to learn collaborative behavior from a single demonstration, and to use this knowledge to take either agent's role in the collaborative behavior. They are able to extract individual's actions in terms of their object manipulation goals and attribute these to the appropriate agent, forming a "bird's eye view" of the collaborative action.

The current research exploits these concepts to allow the iCub humanoid to learn collaborative tasks via single observations of human demonstration. The tasks involve two agents performing coordinated, collaborative sequences of simple object manipulations. Action perception is organized around physical properties of objects – their appearance and disappearance. The robot has a pre-learned action repertoire that mirrors this perceptual capability for actions. During human demonstration our real-time action parser extracts the sequence of actions, including agent attribution. The human and robot then agree on "who goes first" and the shared plan is used by the robot to collaborate, taking the appropriate role in the learned action plan. We present results from 2 experiments in which distinct collaborative behaviors are learned in real-time. We argue that this approach provides a powerful compliment to existing programming by demonstration methods..

1. Introduction

Human children at 18-24 months display a remarkable ability to observe adults perform a collaborative task (with only 1 or two demonstrations) and then to engage themselves in that task, taking the role of either of the demonstrating adults (Warneken et al 2006a,b). Tasks typically involve retrieval of a toy from a physical device which requires both agents to manipulate it in temporally organized and synchronized manner. By definition, the goal-directed tasks require two agents to collaborate – as the physical constraints of the task are such that an individual agent cannot achieve the goal. The behavioral data indicate that the children have understood the task in terms of a coordinated succession of actions, rather than a set of specific motor trajectories. This research has identified

three principal characteristics for collaboration (1) agents are mutually responsive and coordinated, (2) they have a common shared action plan for the joint enterprise. (These provide a "birds eye view" of the collaboration and can be demonstrated by the agents' ability to reverse roles.), and (3) a mutual commitment to the goal (Warneken et al 2006a,b).

Based on these definitions, and standard collaboration scenarios, we have developed the "Get the toy" scenario, in which subjects must collaborate to achieve the goal. In this scenario, a two-handled box is covering the target object, a small toy. In the demonstration, User 1 lifts the box using both hands, allowing User 2 to take the toy. The robot should be able to observe the sequence of actions, form a shared plan (i.e. a plan in which actions are attributed to agents), and then use that plan to take either role in the collaborative action. This provides a framework for more cognitive learning related to the notion that to be grasped an object must be visible and/or not physically covered/obstructed. The observational learning capability shall extend to any scenario (of arbitrary length) consisting of actions that can be recognized and performed by the robot.

2. System Overview

System extends our previous work in the language-action grounding framework (Dominey et al. 2005, 2009).

Construction of shared plans via observation: In the current research, shared plans are to be constructed based on the robot's observation of two humans demonstrating the task to be learned. At the onset of a new interaction, the supervisor indicates to the human by spoken language that it is ready to observe a new interaction. Using the vision-based action recognition, the robot detects human generated action (extending Dominey & Boucher 2005). Once a delay of >10 seconds takes place with no further action, then the system determines that the collaborative interaction has been completed. The shared plan is then committed to the Knowledge base.

Engage in Learned Collaboration: Once the plan has been created and committed to the knowledge base, the system is then ready to engage in the collaborative interaction. The first step is to determine who goes first. The system thus asks the user "Who goes first, you or me?". Based on the users reply, the system

attributes roles to itself and the human.

The system then begins execution of the shared plan. For each action, the system recalls and states who does what. When it is the agent, it performs its action. When the human is the agent, the robot monitors the human performance to determine whether the action was completed.

Dialog management and spoken language processing (voice recognition, and synthesis) is provided by the CSLU Rapid Application Development (RAD) Toolkit (<http://cslu.cse.ogi.edu/toolkit/>). RAD provides a state-based dialog system capability, in which the passage from one state to another occurs as a function of recognition of spoken words or phrases; or evaluation of Boolean expressions.

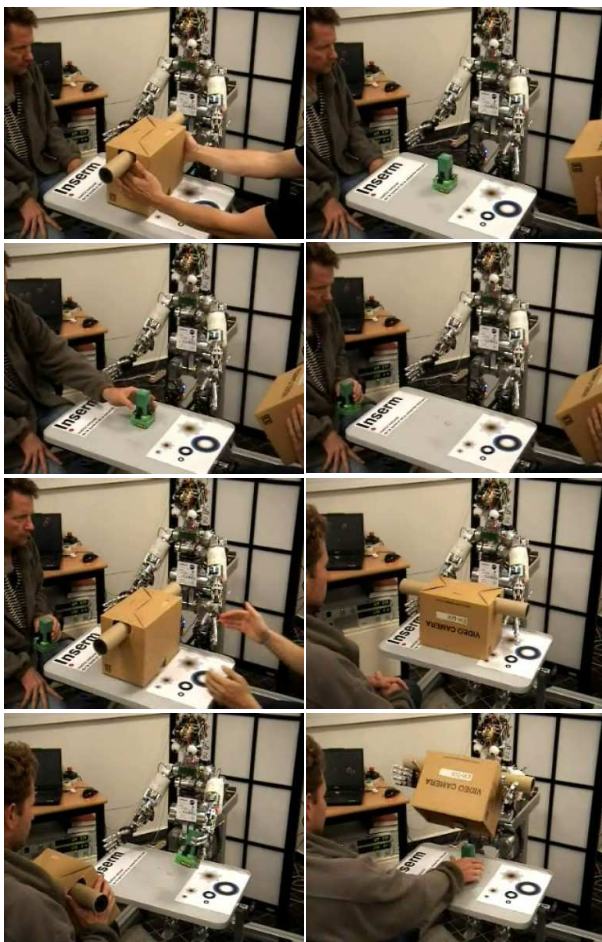


Figure 1. Get the toy interaction. (A) Larry (to the robot's Left) takes the box, revealing the toy (B). (C) Roboert (to robot's Right) takes the toy. (D-E) Larry replaces the box, finishing the game. (F) Now human and robot do the collaborative task. (G) Human takes the box and robot takes the toy. (H) Role reversal – Robot takes the box, and human takes the toy.

3. Experimental results

In order to evaluate the implemented system, we performed a series of experiments in which two humans demonstrated a collaborative behavior to the robot, and then the robot performed the task with one

of the humans.

As illustrated in Figure 1, two users to the left (Larry) and right (Robert) demonstrated the task. Larry lifts the box, revealing the toy. Robert takes the toy, and finally Larry replaces the box. After a delay of ~10 seconds with no action, the system determines that the interaction is finished, and requests verbal validation, which is confirmed. This results in a shared plan of the form: {larry take box, Robert take toy, larry put box}.

The system then asks the user about who should go first. Based on the response, the system identifies who was first in the shared plan, and replaces that person by "you" or "me" based on the user's choice. The shared plan is then ready for execution. In the current version the robot announces who does what before each action. This is optional. When the robot is the agent it performs its action. When the user is the agent the robot validates that the human has performed this action. This validation can be done via vision, or via verbal confirmation from the user. When the robot is the agent, it performs the required action using the capabilities defined in Table 1.

We tested a collaborative behavior that has a different and more extended temporal sequence: Robert puts the box on the table, Larry takes the box. Robert then places the toy on the table, and finally Larry covers it with the box. We performed this demonstration and the robot correctly generated and used the corresponding plan, demonstrating a generalization capability.

4. Acknowledgements

This work is supported by the FP7 IST – PROJECTS– CHRIS (No.215805) and ORGANIC (No.231267), and by the French ANR-07-ROBO-0004-04 Amorces, ANR-08-Blan-0003-01 Comprendre.

References

- Dominey PF, Boucher (2005) Learning To Talk About Events From Narrated Video in the Construction Grammar Framework, *Artificial Intelligence*, 167 (2005) 31–61
- Dominey PF, Mallet A, Yoshida E (2009) Real-Time spoken-language programming for cooperative interaction with a humanoid apprentice, . In Press, *Intl J. Humanoid Robotics*
- Warneken F, Chen F, Tomasello M (2006) Cooperative Activities in Young Children and Chimpanzees, *Child Development*, 77(3) 640-663.
- Warneken F, Tomasello M (2006) Altruistic helping in human infants and young chimpanzees, *Science*, 311, 1301-1303

Integrating a Need Module into a Task-independent Framework for Modeling Emotion: A Theoretical Approach

S.L. Lutfi, C. Sanz-Moreno, R. Barra-Chicote, J.M Montero
Speech Technology Group, Universidad Politecnica de Madrid, Spain
{syaheerah,csmoreno,barra,juancho}@die.upm.es

Abstract

This paper concerns emotion modeling for a task-independent agent by integrating a module of needs. Inspired by theoretical views with regards to human needs, we suggest that appraisals can be confined within various scopes of needs, and to demonstrate this, we propose an emotion framework which allows control over appraisals via pre-defined levels of needs, urgency or priorities.

1. Introduction

In biological systems, motivations are concerned with internal needs related to survival (Canamero, 1997) and psychological needs related to self-sufficiency. Motivation varies as a function of deprivation in a form of varying internal states, and the latter are postulated to explain the variability of behavioral responses (Canamero, 1997). But what is the relation between motivations and emotions? Thomkins (Tomkins, 1984) views emotions as the primary motivating mechanism. According to him, the affect system adds strength to drives as motives - "without its amplification, nothing else matters, and with its amplification, anything else can matter. It thus combines urgency and generality" (p.164). In another similar view, Zimmerman pointed out that the deficiencies of the various levels of need are actually experienced *emotionally* on a conscious level, but the individual may be unconscious regarding the *level of need* he or she is deficient of. In his words (Zimmerman, 2002)(p.3), "The deficiency in safety needs is experienced as *fear* by many people. When safety needs are met, fear disappears". As Maslow [(Maslow, 1999)] has pointed out, when a need is satisfied, a new 'higher' need emerges. In this case we might see the 'love needs' arise in which one needs to be *courageous*. In this sense, fear is replaced by courage". Thus, he coined the word "need-emotions" to relate need deficiencies as experiential emotions.

These theories served as inspiration to incorporate a need module in our affect model for a domain-independent multi-tasking agent. To demonstrate this, we propose an emotion framework which allows control over appraisals via pre-defined *levels of needs*, urgency or priorities. In other words, appraisal components derive information from the need

components, implicitly computed, which results to the elicitation of a suitable emotion response, represented in the agent's behaviour. Any changes in appraisals are dependent on the need level, which underlies the reasoning techniques that support the framework's cognitive process. The motivation framework is based on literature by Abraham Maslow (Maslow, 1999), describing a renowned motivational hierarchy explaining human needs from the most basic to reaching self-actualization. According to Maslow, human beings first gratify the most basic needs, before they are motivated to move on to the next level, thus, each level takes precedence over others. What makes this approach different from other appraisal-based approaches is the addition of the need-layers that function as a decomposer of task-specific events according to their importance and urgency.

Most work in computational modeling of emotion focused on appraisal-based approaches (Gadanh, 2003; Gratch & Marsella, 2004; Marcella & Gratch, 2006). Although we are perhaps the first to introduce the integrated computational account of needs, a similar architecture was acquainted in the eighties for modeling behaviour-based robots by Rodney Brooks (Brooks, 1986). Though the idea of task-decomposition into different layers is similar, our architecture differs in the way the layers handle inputs.

2. Proposed Model of Affect

As a pilot study, we have restricted our agent to fit the scope of domesticity. This means the agent is able to perform simple tasks such as turning on or off the lights, providing weather information, cleaning the floor facilitated by a vacuum cleaner etc. The agent can also act as a game partner and play board-games. These tasks are carried out in two ways: established adaptation to changing surroundings (i.e.: modifying a room environment according to user preference – brightness level in the room, timely preferred TV channel) and by verbal instructions.

In the proposed architecture (Figure 1), each task has a pre-fixed relation with one or more levels of Maslow pyramid. The manipulation of these needs is based on the agent's causal interpretation influenced by both *task-specific* and *general* events. Task-specific events directly relate to the task modules. In other words, events are induced by the module involving a specific task (game module, vacuum cleaner module

etc.) For example, in a game-playing task, events may be a good movement, a bad movement, agent cheating, partner cheating etc. General events are those that are task-independent, and those detected by the agent via speech and facial modalities - such as successfully detecting insults/threats by his partner, smile or frown or even something simpler, such as detecting words accurately. Simply put, these events are *inputs* that affect various needs 'satisfaction' in the scope of the Maslow pyramid, producing varying need values (termed M-values). As events change quickly, M-values vary on the same rhythm, also taking into consideration the values on *previous state* - producing dynamicity. These variations will be further taken as inputs by the Need Independent Features (NIF) Generator to appraise the needs in terms of *Relevance, Urgency, Desirability, Unexpectedness* and *Unfamiliarity*. Appraised needs are output as vectors, whereby each vector is mapped into an *emotion instance* of a specific type and intensity. To account for the prioritization of need (which indirectly projects the importance and urgency of a task), a constant-weight is added to each instance, depending on the need-level (lower level with greater weight). Finally, the dominating emotion obtained effects both the cognitive process and behavior-selection of the agent - similar to the conducts of humans. Our current model illustrates six types of emotions - Happiness, Sadness, Surprise, Anger, Fear and Neutral. These emotions are elicited via two modalities, speech and/or facial.

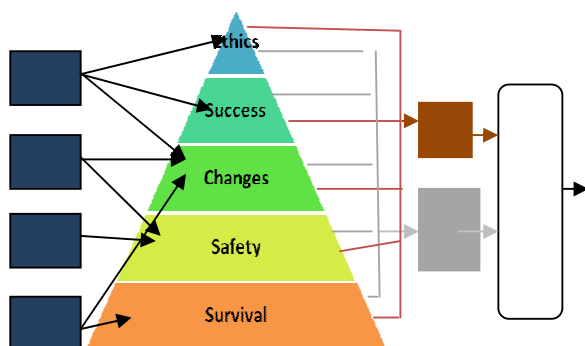


Figure 1 : Need-inspired Affect Architecture

2.1 Task Independency

As explained earlier, the agent's causal interpretation is influenced by both task-specific and general events. In the real sense however, the agent's interpretation is based on the *Maslow variation (M-values)* which is in turn modified by these events. He does not directly analyze the internal operations of each task. Thus, the behavior of the agent is *independent* of the existing task(s). Therefore the agent is made aware of his needs by connecting the cognitive module to the Maslow need pyramid rather than directly to the tasks and its situations. In this way, this module preserves the scalability of tasks, whereby the agent's tasks can be

added or appropriately changed to suit applications in different domains.

3. Conclusion and Ongoing Work

The deficiency of needs is experienced by people emotionally. Their beliefs, goals and plans are influenced by their needs, and the progression towards satisfying their needs predict their emotions over time. Emotion on past, present and future events can be altered by altering their needs. These requirements lead us to a computational framework of emotion that is tied to a causal interpretation of an individual's needs. We argue that the use of the Maslow need framework, which is evident in the nature of human beings, is a suitable technique for problems of prioritization in multi-tasking agents. Apart from that, this way allows flexibility in adding or modifying tasks according to various application domains. An initial demo of our early work can be accessed here (Robonauta GTH, 2009).

Currently we are testing the proposed algorithm in an Excel simulation, and the next step is to transfer this simulation to a formal evaluation.

References

- Brooks, R. (1986). A Robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation*, 2(1), 14-23.
- Canamero, D. (1997). Modeling Motivations and Emotions as a Basis for Intelligent Behaviour: ACM.
- Gadanhó, S. C. (2003). Learning Behavior-Selection by Emotions and Cognition in a Multi-Goal Robot Task. *Journal of Machine Learning Research*, 4(2003), 385-412.
- Gratch, J., & Marsella, S. (2004). A Domain-Independent Framework for Modelling Emotion. *Journal of Cognitive Systems Research*, 5(4), 269-306.
- Marcella, S., & Gratch, J. (2006). *Ema: A Computational Model of Appraisal Dynamics*. Agent Construction and Emotions (ACE 2006), Vienna, Austria.
- Maslow, A. H. (1999). *Toward the Psychology of Being* (3rd ed.). Canada: John Wiley & Sons.
- Robonauta GTH. (2009). *Robotic Assistant with Domestic Capabilities and Automatic Speech Recognition, Speech Synthesis and Emotional Behaviour*: <http://tinyurl.com/robonauta-demo1>
- Tomkins, S. S. (1984). Affect Theory. In K. R. Scherer & P. Ekman (Eds.), *Approaches to Emotion* (pp. 163-195). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Zimmerman, M. (2002). *Abraham Maslow, Emotional Literacy and Ortho-Education: How to Make the World a Better Place (Chap 6)*, 2002, Collective Works: <http://tinyurl.com/zimmerman-emotions>

Investigating the basis for conversation between human and robot

Caroline Lyon and Joe Saunders

University of Hertfordshire, Hatfield, UK, AL10 9AB

C.M.Lyon, J.1.Saunders@herts.ac.uk

Abstract

We investigate preliminary stages in enabling robots to talk with humans in a natural manner, and outline experiments. The process is inspired by language acquisition in infants, and by recent empirical evidence of neuronal organisation.

1. Introduction

In this paper we describe preliminary stages in enabling robots to communicate with humans, using natural language. This starts with babbling, analogous to the pre-linguistic infant in a proto-conversation with its carer, progressing to learning the meaning of utterances through mediated physical interaction (Saunders et al., 2009). This work is inspired by the acquisition of language by human infants, and by recent empirical evidence of neuronal organisation.

Participants in our experiments talk to the robot in natural, unrestricted language, about a blocks world with objects of various shapes. The robot must learn to “understand” this highly redundant natural language, in which the same concept can be expressed in a number of alternative ways (e.g. “push the red box”, “give the red box a push”). Its own productions may be more limited: asymmetrical development is typical of human infant language acquisition (de Boisson-Bardies, 1999, p 201-209).

2. Natural language, evolutionary baggage and neuronal organisation

Language has emerged by recruiting mechanisms that originally evolved for other purposes. For instance, in English, French, Japanese, Chinese and other languages there are many common homophones, ambiguous words such as

no/know to/two/too their/there

We disambiguate such words by taking them in context, processing short sequences of linguistic elements as coherent units.

The fact that we do not avoid ambiguous words but resolve their meaning by processing short sequences suggests that such serial processing methods are easily accessible. It seems likely that sequential processing is based on exaptations of faculties originally developed for different purposes. As Steels says: “the human language faculty is a dynamic configuration of brain mechanisms, which grows and adapts recruiting available cognitive/neural resources for optimally achieving the task of communication” (Steels, 2007).

As well as Wernicke’s and Broca’s areas in the brain other regions are involved in language processing. See, for instance, Lieberman (Lieberman, 2000), Dominey et al. (Dominey et al., 2003), Pulvermuller (Pulvermuller, 2002) on why serial processing is a key factor in the perception and production of speech.

Thus, there is significant evidence that dual systems are needed for language processing. On the one hand there is *implicit* learning of patterns and procedures, without intentional shared reference. On the other hand there is *explicit* declarative learning, in which there is joint attention between teacher and learner, and reference to objects, actions or relationships.

This dichotomy is also described as a dorsal pathway concerned with sub-lexical processing, object interactions and phonetic decoding, in contrast to a ventral pathway specialising in object identification and whole word recognition. This functional segregation is also characterised as a motor-articulatory system on the one hand and a conceptual system on the other (Hickok and Poeppel, 2004, Saur and Kreher, 2008). As described below, we adopt this dichotomy in a simplistic manner in the implementation of a language learning robot.

3. Implementation

Work is currently being undertaken influenced by the constructivist approach of Tomasello (Tomasello, 2003). The aim of this work is to enable the development of language capabilities in a robot through interaction with a teacher, an actual or

simulated human. Initial assumptions are:

- The robot has the intention to communicate
- Communicative ability is learnt through interaction with a teacher
- Perception and production of speech are based on simulated mirror neuron type structures, in which the same elements reflect components of perceived speech and generate synthesized speech
- Memory sites include distinct areas associated with implicit, pattern learning on the one hand and explicit word learning on the other

The process is based on turn taking. Initially the robot (a synthetic agent in preliminary experiments) produces simulated babbling while a teacher produces utterances in ordinary English, both represented as streams of phonemes. The agent's output starts as random syllables, but becomes biased towards the teacher's speech. The starting point is taken as analogous to the stage at which infants start canonical babbling (de Boisson-Bardies, 1999, p. 45-46). Babbling is thought to play a key role in early language development (Oudeyer, 2006, p. 148) (Pulvermuller, 2002, p. 50)

The robot or agent segments the teacher's utterance into short sections in a variety of ways, based on observed mechanisms. These include phonotactic constraints based on distributional evidence, taking the end of an utterance as a significant unit, taking identified words or holophrases as anchor points for further segmentation. Prosodic information plays a key role for humans, and we plan to use it in future. These segments join the robot's store of pre-lexical components, available for use in its productions. When the robot produces, by chance, syllables that can be concatenated to make a word, the teacher will give a positive reaction, metaphorically a "reward". The new word becomes latched in memory, a candidate for future production by the robot along with other syllables. Thus a lexical store is built up, and words will be produced embedded in a stream of non-words, ready to be given semantic reference.

The acquisition of meaning would in reality take place at the same time as speech segmentation described above occurs. However, we are investigating these processes separately initially in order to understand each strand better. Experiments in a blocks world, where a human interacts with the humanoid Kaspar2, are described in (Saunders et al., 2009). Our robots learn to extract the semantics of a series of shapes associated with perceived speech patterns (as strings of phonemes), visual and proprioceptive perceptions.

An associative approach usually requires a large number of learning episodes so that statistical regularities can be established. An alternative mecha-

nism, used here, is to have learning experiences biased through intentional reference, such as shared gazing, pointing and other types of feedback to reinforce the utterances of the teacher.

In developing the basis for conversation between human and robot we cannot avoid the evolutionary baggage that human language carries, and we need to understand our own neural language processors if we are to implement robotic systems that carry out similar tasks.

Acknowledgement

Work described in this paper is conducted within the EU Integrated Project ITALK, 2008-2012, funded by the EU Commission under contract number FP7-214668.

References

- de Boisson-Bardies, B. (1999). *How Language Comes to Children*. MIT.
- Dominey, P. F., Hoen, M., Blanc, J.-M., and Lelekov-Boissard, T. (2003). Neurological basis of language: Evidence from simulation, aphasia and ERP studies. *Brain and Language*, 86:207-225.
- Hickok, G. and Poeppel, D. (2004). Dorsal and ventral streams: a framework for understanding aspects of the functional anatomy of language. *Cognition*, 92.
- Lieberman, P. (2000). *Human Language and our Reptilian Brain*. Harvard University Press.
- Oudeyer, P.-Y. (2006). *Self-organization in the Evolution of Speech*. Oxford University Press.
- Pulvermuller, F. (2002). *The Neuroscience of Language*. CUP.
- Saunders, J., Lyon, C., Nehaniv, C. L., Dautenhahn, K., and Forster, F. (2009). A Constructivist Approach to Robot Language Learning via Simulated Babbling and Holophrase Extraction. In *IEEE ALife09 Conference*.
- Saur, D. and Kreher, B. W. (2008). Ventral and dorsal pathways for language. *Proceedings of the National Academy of Sciences*, 105(46).
- Steels, L. (2007). The Recruitment Theory of Language. In Lyon, C., Nehaniv, C., and Cangelosi, A., (Eds.), *Emergence of Communication and Language*. Springer.
- Tomasello, M. (2003). *Constructing a Language: A Usage-Based Theory of Language Acquisition*. Harvard University Press.

The Use of Emotions in an Autonomous Agent's Decision Making Process

María Malfaz
RoboticsLab.

Carlos III University of Madrid
28911, Leganés, Madrid, Spain
mmalfaz@ing.uc3m.es

Miguel A. Salichs
RoboticsLab.

Carlos III University of Madrid
28911, Leganés, Madrid, Spain
salichs@ing.uc3m.es

Abstract

The end goal of our research is to design an emotion-based decision making system for an autonomous and social robot. This means that the robot can interact with people and/or other robots and that it is the one who decides its own actions. For this reason, a motivational model has been developed for its decision making system. This model uses mechanisms inspired by emotions and their functionality in nature. The behaviours of the robot will be oriented to maintain its internal equilibrium. As a previous step, this model has been successfully implemented in virtual agents, as illustrated by the results given in this paper.

1. Introduction

Emotions and robotics have been recently combined. Many authors have stated the necessity of implementing emotions in robots in order to improve their capabilities. Some of them affirm that robots need emotions for the same reason humans do (Fellows, 2004). It has been proved that emotions influence attention, memory, decision making, and other areas that some years ago seemed not to be related to emotion at all (Picard, 1998). Some researchers stated that since emotions are essential in nature for survival, they should be useful for building autonomous robots (Cañamero, 2003).

The goal of our research is to construct a social and autonomous robot and, based on the ideas previously stated, the implementation of emotions seems to be ideally suited for our objective. Autonomy implies that the robot has to be able to decide its own goals, and then decide on its own behaviours in order to achieve these goals. We currently have a robotic platform developed for human-robot interaction: Maggie. This is a social robot developed by the RoboticsLab research team and is fully explained in (Salichs et al., 2006).

The motivational system proposed in this paper will enhance the autonomy of Maggie and it has been successfully implemented on virtual agents (Malfaz and Salichs, 2006).

2. Motivational Model

In Fig.1 the proposed motivational system is shown. Based on other works, (Cañamero, 2003), we consider that an autonomous agent selects its behaviours in order to maintain a stable internal equilibrium. In our case, this internal equilibrium is related to the optimization of its wellbeing. The agent will make its decision according to a motivational model based on drives (internal needs), motivations, and emotions. The wellbeing of the robot/agent is defined as a function of its drives and it measures the degree of satisfaction of its internal needs.

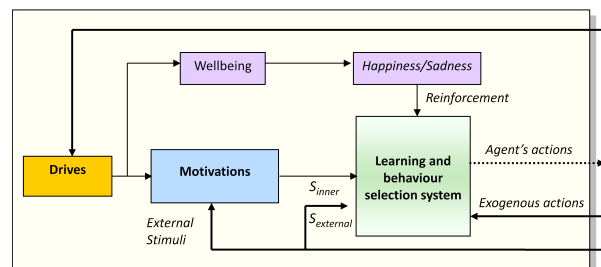


Figure 1: Motivational system

The Learning and Behaviour System is where the learning process takes place. The input received by the Learning and Behaviour System comes from the state of the agent. This includes the inner and external states and other exogenous actions not executed by the agent, but by other agents. Within this system, the agent evaluates and selects the action most appropriate for a certain state, and then executes the action selected. For this purpose, the agent uses a reinforcement learning algorithm. By using this algorithm the agent learns the long term value of executing an action in a certain state. The best suited action to execute will be that which has the highest

value.

In relation to emotions, based on the idea that those are states elicited by reinforcements (rewards or punishments) (Rolls, 2003), some emotions, such as happiness and sadness, are used in the reinforcement function.

Moreover, based on some theories that claim emotions can motivate behaviours (Breazeal, 2002) (Rolls, 2003), other emotions can also be a motivation, e.g. fear. According to Ortony (Ortony, 2003), fear appears when the possibility of something bad happening exists.

3. Experimental Results

In this section we present experimental results obtained from two cases: the agent with no Fear motivation and the agent with Fear motivation. In order to carry out these experiments the agent lives with two kinds of opponents: a neutral agent who randomly selects from a repertoire of non-aggressive actions; and a dangerous agent, who 95% of the time chooses its actions from a repertoire of non-aggressive actions, while the other 5% of the time it kicks.

When the agent is kicked it receives a significant negative reinforcement. Therefore, interaction with the second opponent may be dangerous although the agent, at the beginning of its life, will not be aware of it. When using this approach, Fear is related to the worst experience the agent had while interacting with an opponent.

As shown in table 1, the agent with no Fear interacts with the dangerous agent even though this opponent may kick it at times. In fact, during the experiment, the agent interacts with the neutral agent a total of 326 times and a total of 214 times with the dangerous opponent.

This happens because while the dangerous opponent treats the agent kindly the majority of the time, once in a while it behaves badly towards it. Therefore, the long term value of the social interactions learned through reinforcement learning is high.

Table 1: Number of interactions with both opponents

	With no Fear	With Fear
Neutral agent	326	376
Dangerous agent	274	4

On the other hand, when the agent had Fear as a motivation, the agent interacted with the dangerous agent a mere 4 times while it interacted with the neutral opponent a total of 376 times, see table 1.

The agent's consideration of the worst experience while interacting with the dangerous opponent gives the agent the ability to be able to detect a dangerous

situation. In doing so, it will learn what actions to select.

On this particular occasion, it is after being punished several times that the agent considers that being next to the dangerous agent is dangerous. Therefore, the agent learns to not interact with the agent that can harm it. It has, in fact, become afraid of being next to the dangerous agent.

Moreover, the agent learns that when it is scared the appropriate action is to escape. This escape action is not an *a priori* programmed action.

Acknowledgements

The authors gratefully acknowledge the funds provided by the Spanish Government through the projects called "Peer to Peer Robot-Human Interaction" (R2H), funded by MEC (Ministry of Science and Education) and the project "A new approach to social robotics" (AROS), funded by MICINN (Ministry of Science and Innovation).

References

- Breazeal, C. (2002). *Designing sociable robots*. The MIT Press.
- Cañamero, L. (2003). *Emotions in humans and artifacts*, chapter Designing emotions for activity selection in autonomous agents. MIT Press.
- Fellows, J. (2004). From human emotions to robot emotions. Technical report, AAAI 2004 Spring Symposium on Architectures for Modelling Emotion: Cross-Disciplinary Foundations. SS-04-02. AAAI Press.
- Malfaz, M. and Salichs, M. (2006). Emotion-based learning of intrinsically motivated autonomous agents living in a social world. In *ICDL 5: The Fifth International Conference on Development and Learning, Bloomington, Indiana*.
- Ortony, A. (2003). *Emotions in humans and artifacts*, chapter On making believable emotional agents believable, pages 188–211. MIT Press.
- Picard, R. W. (1998). *Los ordenadores emocionales*. Ed. Ariel S.A.
- Rolls, E. (2003). *Emotions in humans and artifacts*, chapter Theory of emotion, its functions, and its adaptive value. MIT Press.
- Salichs, M., Barber, R., Khamis, A. M., Malfaz, M., Gorostiza, J. F., Pacheco, R., Rivas, R., Corrales, A., and Delgado, E. (2006). Maggie: A robotic platform for human-robot social interaction. In *IEEE International Conference on Robotics, Automation and Mechatronics (RAM 2006)*. Bangkok. Thailand.

Multimodal Representation of Hand Grasping based on Deep Belief Nets

Masaki Ogino Takanori Nagura Minoru Asada^{*}
 JST ERATO Asada Synergistic Intelligence Project
 Yamadaoka 2-1, Suita, Osaka 565-0871, Japan
 Osaka University
 Graduate School of Engineering, Department of Adaptive Machine Systems,
 Yamadaoka 2-1, Suita, Osaka 565-0871, Japan

In human brain, different sensor information is thought to be processed in different area and integrated in parietal area. Fig. 1 shows a model of neural mechanism for grasping proposed by Oztop et al (Oztop et al., 2006). As shown in this figure, the information of hand and object is processed separately and important features for grasping are extracted in the hierarchical network.

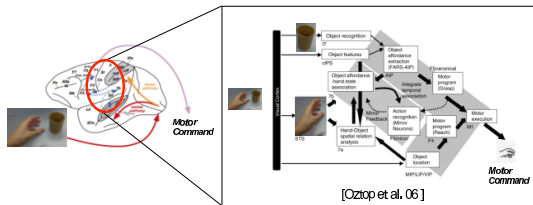


Figure 1: Neural mechanism for grasping proposed by Oztop et al. (Oztop et al., 2006)

In this paper, we aim to construct a hierarchical model for grasping like brain model. The hierarchical model is thought to be plausible as developmental model, because an infant learns its grasping skills gradually in the developing process (Case-Smith and Pehoski, 1992). For this purpose, we adopt deep belief network (DBN), proposed by Hinton, for representing the multimodal information in grasping, in which one modal information is self organized to extract statistical information of given data and different modal information are easily integrated in the hierarchical architecture (Hinton, 2007).

From grasping experiences, four kinds of multimodal information are extracted and input to neural networks for tactile sensing, joint angles, hand images, and object image, respectively. In each modal, raw sensor information are self organized using restricted Boltzmann machine (RBM) and input information is represented in tactile feature, hand feature and object image feature. In this model, it is assumed that information on hand posture such

as joint angle and hand image are integrated as hand feature before integration for grasping. Total integration are processed using information during grasping objects. After learning of integration, tactile senses and hand information are recalled from the object image by virtue of DBN properties (Fig. 2).

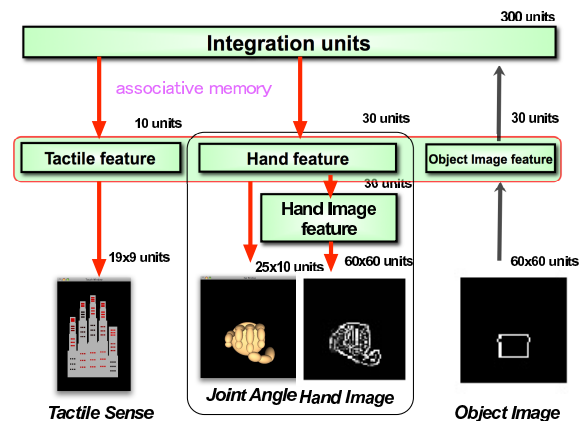


Figure 2: Reconstructing the tactile sensing and hand features from an object image

References

- Case-Smith, J. and Pehoski, C. (1992). *Development of Hand Skills in the Child*. Amer Occupational Therapy Assn.
- Hinton, G. (2007). To recognize shapes, first learn to generate images. In *Computational Neuroscience: Theoretical Insights into Brain Function*.
- Oztop, E., Kawato, M., and Arbib, M. (2006). Mirror neurons and imitation : a computationally guided review. *Neural Networks*, 19:254-271.

How Are Representations Affected by Scene Statistics in an Adaptive Active Vision System?

Dimitri Ognibene* Giovanni Pezzulo** Gianluca Baldassarre*

*ISTC-CNR Via S.Martino della Battaglia, 44 - 00185 Rome, Italy

**ILC-CNR Via Giuseppe Moruzzi, 1 - 56124 Pisa, Italy

{dimitri.ognibene,giovanni.pezzulo,gianluca.baldassarre}@istc.cnr.it

1. Introduction

One of the main claims of active vision (Ballard, 1991) is that finding data on demand, based on the requirements of the task, is more efficient than reconstructing the whole scene by performing a complete visual scan of it. This aids generalisation and a dramatic reduction of the needed visual computations. Using this strategy, however, generates the need to learn complex gaze control strategies dependent on the pursued goals and the properties of scenes and objects. For example, to be able to find an object in the environment an agent needs to learn to use several sources of information such as spatial relations of objects and bottom-up saliency of scene regions. In addition, if the system is genuinely autonomous it also needs to develop a representation of the objects themselves, for example of potential targets, cues and distractors, on the basis of generic reward signals to be maximized and the visual control policy used. Most of the models proposed in developmental robotics do not use *adaptive* visual control and so are ill suited to investigate these issues.

In a previous work (Ognibene et al., 2008) we presented a reinforcement-learning neuro-robotic architecture, based on neural population codes, which was able to *develop attention control* policies by interacting with the environment *based on a rewarded reaching task* it had to accomplish. In this paper the same architecture is used to investigate the types of *internal representations* that this same architecture develops when exposed to two classes of environments where objects are organised on the basis of *contrast-ing spatial relations* (Figure 1).

A recent view on neural population codes proposes that neural maps might be used to develop overall probability distributions of stimuli (Pouget et al., 2002). On the contrary, this study shows that active vision systems tend to develop actions which actively disambiguate the stimuli and acquire new evidence only when needed: as a consequence, the acquired representations do not reflect overall probability distributions related to stimuli but rather the contextual relationships between them.



Figure 1: Examples of environments used to test the model, drawn from two classes of environments **L** and **R**. In each trial, the specific environment was randomly drawn from **L** or **R** with a probability of 75% and 25%, respectively. Both classes of environments were based on 2 to 5 green cues forming a vertical line, one blue distractor, and one red target. The cues, distractor and target were located on the vertexes of a 5×6 matrix. In **L** environments, the target and distractor were located at a random position respectively at the left and right of the green line, whereas in **R** environments were located at a random position respectively at the right and left of it.

2. The model

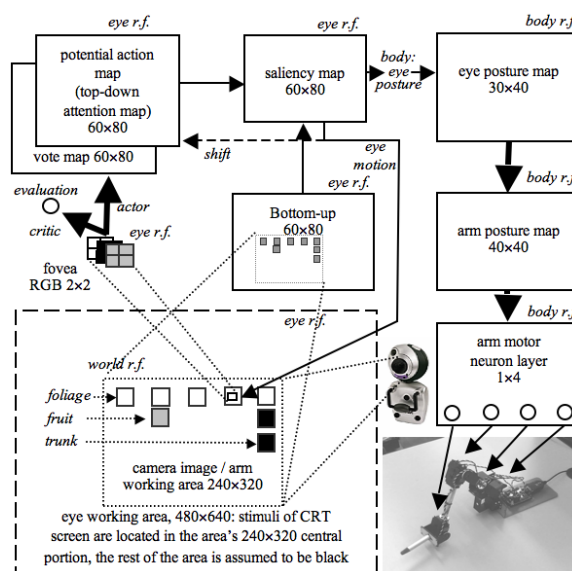


Figure 2: The architecture of the model.

The architecture and setup of the model (Figure 2.), used here in a simulated version, is now

briefly described but the reader should refer to Ognibene et al. (2008) for details. The robotic setup used to test the model is composed of a camera looking down to a robotic arm. The arm acts on a working plane consisting of a screen which shows the visual stimuli of the task.

The architecture of the model is formed by three main components:

(a) *Bottom-up attention component.* The input image is used to activate a *periphery map* which identifies high-contrast regions on the basis of suitable filters .

(b) *Top-down attention component.* The central part of input image (*fovea*) is used as input of an *actor-critic* model which learns to predict, by suitably activating the output map of the actor (*vote map*), the spatial position of the rewarded targets with respect to the foveated objects. A *potential action map (PAM)*, based on leaky neurons, accumulates evidence, furnished by the actor, on possible locations of the target while the fovea explores the scene objects. An overall *saliency map* integrates information from the periphery map and the PAM to select the next eye movement on the basis of a dynamic neural competition. All maps of the attention components use an eye-centered reference frame.

(c) *Arm-control component.* Each fixation point, encoded in a *eye posture map*, suggests a potential target to a *arm posture map*: when the eye fixates a location for enough time (3 time steps on average), the arm posture map triggers a related arm action on the basis of a second dynamic neural competition. If the reached object is the target, the system gets a reward of one, otherwise it gets a small punishment (mimicking energy consumption).

3. Results and Conclusions

The tests of the model show that it learns an exploration policy which initially assumes to be tackling an **L** environment, so first searches the green line and then, on this basis, the target on its left (two eye steps). In the presence of an **R** environment, this assumption fails and the agent searches the target directly on the right of the green line rather than exploring anew. This strategy allows the system to find the target with only one additional step.

Table 1 shows the activation of the vote map of two agents respectively trained with **L** environments or with both **L** and **R** environments (with a frequency of 75% and 25%, respectively), when the agents foveate either the cue or the distractor (a third agent trained only with **R** environments developed vote maps mirroring those of the **L**-trained agent: data not reported).

These results show that the representations underlying the gaze-control policies are not based on a combination of all possible policies needed to tackle

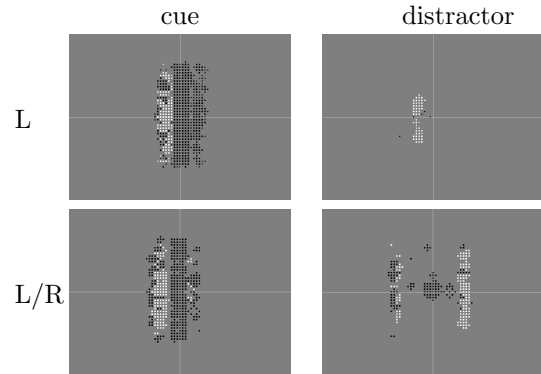


Table 1: Activation of the vote map when the model foveates the cue or distractor. L: agent trained only on **L** environments for 60.000 trials. L/R: agent trained on both **L** and **R** environments for 60.000 trials.

the two classes of environments. In fact in the latter case one would expect internal representations to be a combination of the vote maps needed to tackle the **L** or **R** environments in isolation (e.g., a sum or a max of the two). Instead, the internal representations encode the specific exploration routines best suited to solve the task at hand. This is especially evident if one considers the vote maps related to the distractor: when the system is trained with **L** environments, the map does not encode the position of target but only the action of foveating the green line, whereas when trained with both **L** and **R** environments the system encodes the action of going to the right of the green line as in this case the distractor becomes a good predictor of the target located there.

These strategies exemplify a general principle used by adaptive active vision system to tackle complex environments. When agents must learn to autonomously and adaptively solve tasks, the representations they develop reflect the actions that permit to interact with the environment in order to acquire new information and solve tasks given the information acquired that far, more than the overall statistics of scenes.

Acknowledgements Research funded by the EU project IM-CLeVeR (FP7-ICT-IP-231722).

References

- Ballard, D. (1991). Animate vision. *AI*, 48:57–86.
- Ognibene, D., Balkenius, C., and Baldassarre, G. (2008). Integrating epistemic action (active vision) and pragmatic action (reaching): A neural architecture for camera-arm robots. In *SAB’08*, Osaka, Japan. Springer.
- Pouget, A., Ducom, J. C., Torri, J., and Bavelier, D. (2002). Multisensory spatial representations in eye-centered coordinates for reaching. *Cognition*, 83(1):B1–11.

Self-motivated learning robot

Mohamed Oubbati

mohamed.oubbati@uni-ulm.de

Günther Palm

guenther.palm@uni-ulm.de

Institute of Neural Information Processing. 89069 Ulm, Germany

1. Introduction

Many psychological studies, starting from the classic paper (White, 1959) to more recent efforts (Dayan and Belleine, 2002), show that children are not passive learners but are intrinsically motivated to progress in learning. It is such a form of active learning which has fascinated roboticists and pushed them to think about new learning architectures (Weng et al., 2001, Oudeyer et al., 2007). Taking inspiration from development in neuroscience and psychology, numerous researchers (Weng et al., 2001, Scassellati, 2000, Marshall et al., 2004) have persuasively argued that a developmental approach could open new issues for designing intelligent robots. According to a developmental approach, a robot would be able to explore its environment not to fulfill predefined tasks but to learn a broad set of reusable skills.

In our work we want to explore new issues to support this field of research. Our objective is to develop basic models and techniques, enabling a robot to acquire new knowledge via self-motivated learning. We expect that the robot will be able to grow its cognitive capacity in an unlimited fashion to generate well co-ordinated actions and later accomplish meaningful tasks.

2. Methodology

In our opinion, self-motivation provides an agent with the desire to manipulate the world and discover new things. Interaction with the physical world has a crucial advantage for open-ended learning, since learning materials, i.e. training data, are basically unlimited. Gil (Gil, 1996) proposed a methodology for learning from the environment by experimentation. She described how it is possible to detect missing knowledge which leads to a need for learning. By experimentation it is meant that the learning system *probes* the physical world in order to fill the knowledge gap. For example, the agent might just randomly generate actions on its environment and observe the consequences of these actions. This is inspired from animals: “*curious animals, faced with a static environment, will go and perturb it, even at great risk to their safety*” (Grand, 1998). We will start from the assumption that **missing knowledge**

of the environment leads to a need for learning. That is, the robot is always *motivated* to understand the environment. It might need to model the environment in order to predict the consequences of its actions. Any unexpected behaviour of the environment leads to identify a knowledge gap which triggers the learning process. Although this developmental robot learns basically through self-motivation, it takes also the advantages of a human partner. When the world remains unpredictable, the agent will be “frustrated” and looks for the human-teacher. Reinforcement learning with external human rewards (Isbell et al., 2001) and learning by animal training techniques (Blumberg et al., 2002) are some starting points for us to define how human instructions can be incorporated in the learning process. Technically, these ideas and the algorithms for their realisation can be formulated in the broad framework of reinforcement learning (Sutton and Barto, 1998). However, we have to create different kinds of rewards: “external” rewards from the interaction with the physical environment and with humans, and “internal” rewards from an internal motivational system.

3. Learning system

An important idea in our work is to organize the learning process into successive learning problems or developmental stages that build on previously acquired sensori-motor organization and corresponding representations in a hierarchical manner. This development will be guided by a corresponding design of the environment. We will design a **motivational system** that analyses the accuracy of predictions made on the environment, and resorts to the learning process when additional information is needed. This leads to design a **learning system** which uses both *experimentation* on the environment and *interaction with humans* as learning mechanisms. The overall system architecture is depicted in Figure (1). The **motivational system** controls the overall *motivation* and *frustration* of the agent and triggers the **learning system** only when additional information is needed to accomplish the general goal. This shall be achieved by having continuous internal interaction with the world-model and weighting different reward signals coming from the environment in a

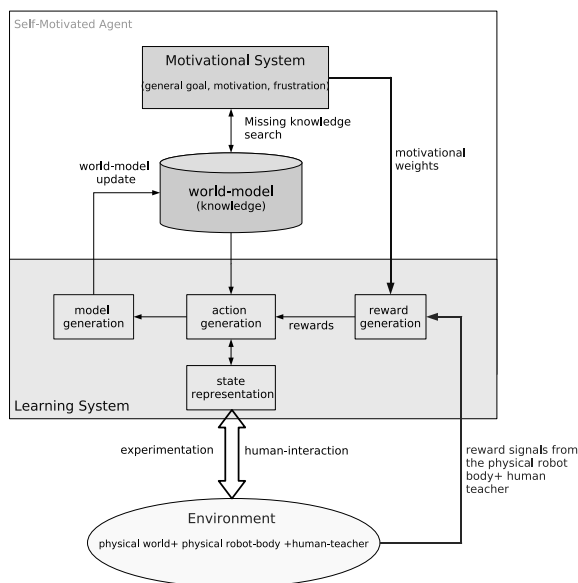


Figure 1: Learning through self-motivation

given situation. That is, our approach is based on a *continuous* and *selective* interaction with the environment (including a human partner) to design a self-motivated robot. The *model generation* module is a system which is trained to predict future environment states from previous action/state pairs, and to update the world-model.

4. Progress in Learning

In the following we briefly describe stages and progress in learning which allow measuring the evolution of the learning process.

Stage 1 (initial exploration) The emphasis here is on building and updating the internal representation of the environment. The robot sends out random actions and receives information back from the environment. This will generate mappings between the robot's actions, the robot's state, and the environment's state

Stage 2 (selective exploration and exploitation)

The robot should produce the "desire" to learn. Here the motivational system will play a major role: (i) it analyses predictions on the world model, (ii) identifies the knowledge gap, and (iii) triggers the learning process when new information is needed to accomplish a global goal. The motivational system could be seen as a Meta-learner which learns how to generate the desire to learn, i.e. how to trigger learning for improving the reliability of the predictions of the world-model. At stage 2 the robot is expected to conduct one fully embodied autonomous experimentation pushed by its motivational system.

Stage 3 (learning from human) The robot learns from humans when experimentation on the surrounding world can not fill the knowledge gap. This is what we call the *frustration* state of the robot. Frustration could be generated when the world remains unpredictable or remains completely predictable. The motivational system identifies the frustration state and triggers the learning system to engage in interaction with the human-teacher. At this stage, we expect that under human guidance the robot will improve its behavior into more complex sequential actions.

References

- Blumberg, B., Downie, M., Ivanov, Y. A., Berlin, M., Johnson, M. P., and Tomlinson, B. (2002). Integrated learning for interactive synthetic characters. *ACM Trans. Graph.*, 21(3):417–426.
- Dayan, P. and Belleine, W. (2002). Reward, motivation and reinforcement learning. *Neuron*, 36:285–298.
- Gil, Y. (1996). Planning experiments: Resolving interactions between two planning spaces. In *AIPS*, pages 102–109.
- Grand, S. (1998). Curiosity created the cat. *IEEE Intelligent Systems*, 13(3):2–4.
- Isbell, C. L., Shelton, C. R., Kearns, M. J., Singh, S. P., and Stone, P. (2001). A social reinforcement learning agent. In *Agents*, pages 377–384.
- Marshall, J., Blank, D., and Meeden, L. (2004). An emergent framework for self-motivation in developmental robotics. In *Proc. of ICDL 2004*, Salk Institute, San Diego.
- Oudeyer, P.-Y., Kaplan, F., and Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*.
- Scassellati, B. (2000). How developmental psychology and robotics complement each other. In *NSF/DARPA Workshop on Development and Learning*. Michigan State University, Lansing, MI.
- Sutton, R. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. The MIT Press.
- Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., and Thelen, E. (2001). Autonomous mental development by robots and animals. *Science*, 291:599–600.
- White, R. W. (1959). Motivation reconsidered: the concept of competence. *Psychological Review*, 66:297–333.

Emerging Attention: Reward Based Model

Vitaly Pimenov

Faculty of Applied Mathematics and Control Processes
Saint-Petersburg State University
Saint-Petersburg, 198504, Russian Federation
vitaly.pimenov@gmail.com

Abstract

Paper proposes a reward-centric model of attention emergence. Attention control is studied from the effectiveness point of view. Reward is suggested to be a measure of attention movement efficiency. Two scales of reward, epigenetic and genetic, are considered to explain emergence of top-down, agent driven, and bottom-up, stimulus driven, attention respectively. It is shown that reinforcement learning framework can be smoothly applied to build a computational implementation of the model.

1. Introduction

Study of attention in natural and artificial systems has seen impressive progress over past decades (see (Frintrop, 2006) for a review). Applications of attentional perception include very diverse domains: object recognition and manipulation, robotic navigation, human-robot interaction, motion analysis, visual search, image and video classification and other.

Due to practical demand and availability of efficient hardware modern research shows ever increasing interest to the problem of attention emergence; development of learning methods has been brought to forefront.

Majority of computational attention systems are based on supervised learning (e.g. (Frintrop, 2006, Rasozadeh et al., 2007)) intended to solve visual search tasks. Examples of search targets therefore become a learning experience in such systems.

Current work embodies an alternative approach that is based on learning optimal decisions about attention movement. Several computational attention models on this type implemented on the basis reinforcement learning have been already developed (Shariatpanahi and Ahmadabadi, 2007, Paletta et al., 2005, Ognibene et al., 2008, Mozer et al., 2006). Also research devoted to simulating shared attention and gaze following (Jasso and Triesch, 2007, Matsuda and Omori, 2001) should be noticed.

Existing approaches have significant differences and there is no single comprehensive framework for understanding attention emergence. In particular, there is a gap in understanding top-down attention control mechanisms and their integration with bottom-up processes.

The goal of current research is to introduce a model of attention emergence capable to explain both bottom-up and top-down attention emergence. It is claimed that such model can be developed on the basis of enactive system approach. “*The only condition that is required of an enactive system is effective action*” (Vernon, 2006).

Recent results in computer vision (Paletta et al., 2005), neuroscience (Deco, 2004) and psychology (Deubel, 2004) support evidence that attention involves decision making and thus can be seen from an effectiveness point of view. Following the reinforcement learning paradigm widely employed for robot learning it is proposed that attention control effectiveness can be evaluated in terms of delayed reward.

2. Model of Attention

Consider an agent perceiving the environment with sensors $P = \{p^1, \dots, p^k\}$. At each time step t agent perceives input $P_t = \{p_t^1, \dots, p_t^k\}$, where $p_t^i = (p_{t1}^i, \dots, p_{tn_i}^i)$ — measurements made by sensor p_i .

Attention movement is a choice of an input subset $\bar{P}_t = \bar{P}(P_t) = \{p_t^{j_1}, \dots, p_t^{j_a}\}$ for further processing. Efficiency of attention movement can be determined with a certain criterion J , such that optimal attention control maximizes value of J . It is proposed that J can be described as a reward $R(\bar{P}_t)$ related to attention movement \bar{P}_t . Reward can have different scales: two essential scales are proposed to explain emergence of top-down and bottom-up attention — epigenetic and genetic scale respectively.

Reward on an epigenetic scale is perceived by agent and serves as reinforcer by increasing the frequency of the action that results in reward. Such reward for example can explain development of gaze following in infants. As calculation of reward requires focusing attention and there is a temporal de-

lay in reward, in the scope of this proposal expected discounted reward is considered as a source of attention control decision. Thus for discrete time criterion J takes a following form:

$$J = E \left[\sum_{t=0}^{\infty} \gamma^t R(\bar{P}_t) \right]. \quad (1)$$

Reward on a genetic scale has an evolutionary meaning: survival of the fittest. This process caused known bottom-up attention mechanisms to appear presently. Such reward is not perceived by agent.

It can be seen that (1) corresponds to non-deterministic γ -discounted cumulative reward used in reinforcement learning. Therefore attention control problem can be easily reformulated as a reinforcement learning problem and can further be solved with known methods. Furthermore, attention control can be coupled with existing robotic architectures as (Taylor et al., 2009) within action-perception loop, similarly to the system described in (Ognibene et al., 2008).

It should also be noticed that concept of saliency, widely used in literature (Frintrop, 2006, Paletta et al., 2005, Shariatpanahi and Ahmadabadi, 2007), can now be understood as an approximation of reward.

Analogy between saliency and reward was already proposed in (Jasso and Triesch, 2007): in their model focusing attention on more salient points brings more reward. In current paper this analogy is reversed and following hypothesis is proposed: *saliency is an anticipation of reward*, i.e. the more reward was associated with a stimulus in the past the more salient for the agent it will become in the future.

Assuming above statements about nature of saliency, important conclusions concerning reward calculation can be drawn. On one hand, analyzing known sources of saliency it is possible to model existing reward structures. On the other hand, given a known reward structure new saliency measures can be designed for specific applications.

3. Conclusions and Future Work

The paper brings forward a model of attention emergence that extends previous work on learning methods. The novelty of research is a provision of a computational attention model that explains emergence of both bottom-up and top-down attention.

There are two major directions of further research. First, coherence of proposed model with modern theories of attention is to be proven: experiments should be carried out to show how reward structures can be induced from known saliency measures. Also a plausible neural implementation is demanded.

Second, since there are no limits on reward structure, it is possible to design complex rewards that

can depend on object recognition, logical reasoning and other high-level cognitive functions. Efficiency of such task-specific rewards is to be evaluated.

References

- Deco, G. (2004). The computational neuroscience of visual cognition: Attention, memory and reward. In *WAPCV*, pages 49–58.
- Deubel, H. (2004). Localization of targets across saccades: Role of landmark objects. *Visual Cognition*, 11:173–202.
- Frintrop, S. (2006). *VOCUS: A Visual Attention System for Object Detection and Goal-Directed Search*. Springer.
- Jasso, H. and Triesch, J. (2007). Learning to attend — from bottom-up to top-down. In *WAPCV*, pages 106–122.
- Matsuda, G. and Omori, T. (2001). Learning of joint visual attention by reinforcement learning. In *Proc. of the 4th Int. Conf. on Cognitive Modeling*, pages 157–162.
- Mozer, M. C., Shettle, M., and Vecera, S. P. (2006). Control of visual attention: A rational account. *NIPS*, 18:923–930.
- Ognibene, D., Balkenius, C., and Baldassarre, G. (2008). A reinforcement-learning model of top-down attention based on a potential-action map. In *The Challenge of Anticipation*, pages 161–184.
- Paletta, L., Fritz, G., and Seifert, C. (2005). Q-learning of sequential attention for visual object recognition from informative local descriptors. In *ICML*, pages 649–656.
- Rasozadeh, B., Tavakoli Targhi, A., and J.-O., E. (2007). An attentional system combining top-down and bottom-up influences. In *WAPCV*, pages 123–140.
- Shariatpanahi, H. F. and Ahmadabadi, M. N. (2007). Biologically inspired framework for learning and abstract representation of attention control. In *WAPCV*, pages 307–324.
- Taylor, J. G., Hartley, M., Taylor, N., Panchev, C., and S., K. (2009). A hierarchical attention-based neural network architecture, based on human brain guidance, for perception, conceptualisation, action and reasoning. *Image Vis. Comput.*, 27(11):1641–1657.
- Vernon, D. (2006). The space of cognitive vision. In *Cognitive Vision Systems*, pages 7–24.

Long Short-Term Memory for Affordances Learning*

Sergio Roa
sergio.roa@dfki.de

Geert-Jan Kruijff
gj@dfki.de

German Research Center for Artificial Intelligence / DFKI GmbH

Abstract

This paper addresses the problem of sensorimotor learning from the perspective of affordances learning of simple objects. We are developing a scenario where a robotic arm interacts with a polyflap, a simple 3-dimensional geometrical object. We perform experiments with a simulated arm using a physics simulator, but we plan to use also a real arm. The robot interacts with the object by pushing it in different ways. We use Recurrent Neural Networks to predict the arm and object poses during this interaction, given a discrete set of random actions that the robot can produce.

1. Introduction

Robots should be able to adapt and learn by interacting in dynamic environments, if we want that they acquire the kind of complex skills performed by humans and animals in general. In altricial animals (like humans) the development of complex motor skills is continuously improved after different stages of development. In these species (Sloman and Chappell, 2005), the interaction with the environment plays an important role for the acquisition of sensorimotor abilities, and for the hierarchical acquisition of more complex skills based on the ones previously acquired. This introduces us to the concept of affordance, which is for instance referred to learning about and from actions performed by an agent on an object. In (Gibson, 1977), a theory of affordances was developed. We can apply this theory of cognitive development to the field of robotics by employing, for instance, machine learning techniques that allow the robot to predict action consequences on certain objects. The interaction with objects and in general with different environmental aspects allow to shape the “mind” of the robot on the basis of its acquired experience.

Taking into account that the environment and the physical characteristics (embodiment) of a robot has a complex structure, we have to think of proper scenarios where we can test these techniques and theories. In (Sloman, 2006), simple scenarios using 3-dimensional objects called polyflaps were proposed.

*The research reported of in this paper is supported by EU FP7 IP “CogX” (ICT-215181)

The objective is to steadily increase the complexity of the space of actions and the structure of the environment. That would allow us to evaluate algorithms that can be useful for compositional (hierarchical) skills development.

It is also important to identify what kind of perceptions can drive learning for an autonomous robot. Based on the way children acquire learning skills at early stages of development, the works presented in (Oudeyer et al., 2007, Roa et al., 2008) describe a system in which the robot has an intrinsic motivation for learning, based on the interestingness of the situations it discovers. For these tasks, a simple intrinsic reward mechanism is employed, which is proportional to the increase of the error rate of some classifier trying to predict the consequences of the robot actions at a given time. The robot was able to identify *affordances* as correlations between its space and actions and its consequences in the environment. In this work, classifiers are used for prediction and the robot is equipped with real-valued sensors and actions comprising its sensorimotor space. After training, there are different classifiers specialized (biased) in some regions of the state space. A statistical mechanism to split the state space into regions is implemented to support the specialization of the classifiers.

2. Scenario

As already pointed out, we use a robotic arm which interacts with a polyflap in a simulated environment (Figure 1).

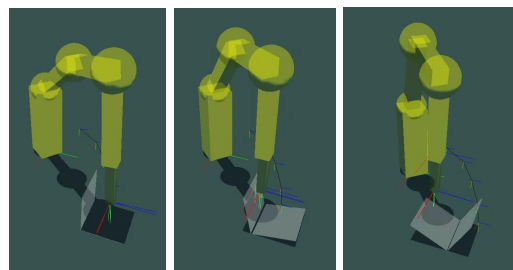


Figure 1: Learning scenario with a polyflap

We use a simulator that can track objects and returns an object pose. Objects that we consider are polyflaps and the arm body parts, which are simple

objects from which we can obtain 3D information. Thus, the task is to use machines that can predict spatio-temporal sequences, and this can be seen as a time-series prediction or regression problem. A sample $\mathbf{s} = [\mathbf{c}, \mathbf{s}_i]_{i=1, \dots, n}$ is then a whole sequence of feature vectors $\mathbf{s}_i = [\mathbf{v}_i, \mathbf{m}_i]$, where \mathbf{v} denotes a vector containing visual data of an object (pose in homogeneous coordinates), \mathbf{m} denotes motor information (joints pose, joint velocities) and i a time frame number up to the limit $n = 70$, together with a motor control command vector \mathbf{c} . In practice, the actions considered are pushing actions on a linear trajectory applying a velocity profile (a 4th degree polynom) to an online inverse kinematics solver and an horizontal direction angle. The values are normalized with mean 0 and standard deviation 1.0.

3. Learning Approach

The learning machines described in (Oudeyer et al., 2007, Roa et al., 2008) can predict short-term consequences of actions. They use an active learning mechanism which uses a measure of learning progress based on the error prediction to select next actions according to this interestingness measure. In this case we are facing a spatio-temporal prediction problem. Recurrent Neural Networks (RNNs), and more specifically Long Short-Term Memory (LSTM) machines (Hochreiter and Schmidhuber, 1997, Graves, 2008) have been shown to accurately predict sequences over extended periods of time. Another approach is the CrySSMEx algorithm (Jacobsson, 2006) which could either extract a probabilistic finite model (a substochastic machine) of the experiences learned by the RNNs (LSTM) or be used itself to analyze the sensorimotor space (as a dynamic system) over several periods of time, and finally extract a model. More importantly, these models should give us a categorization of different object behaviours and corresponding affordances, i.e., given similar objects (similar features) the predictions should be similar. By using these machines, it is possible to evaluate the certainty of the machine to predict action consequences over several periods of time. This mechanism would afford to simulate a kind of mastery driven action selection (if the RNN successfully predicts action consequences) or curiosity driven action selection (if the RNN is failing to predict action consequences and there is learning progress). Other kinds of drives might be novelty (unpredictable action consequence), surprise (unexpected outcome) or interactive (based on a human reward/punishment signal). A feature vector in a frame i is processed at a time step t . The RNN should then predict the corresponding feature vector in the next frame $i + 1$ at some time $t + \delta$, till $i = n$. Initially, we use gradient-based methods

for offline learning and in online experiments this knowledge might also be used as a kind of knowledge transfer method. In general, a LSTM is composed of input units, special units (gate units, memory cells) or conventional hidden units. The weights w are learned by using a modified gradient descent algorithm, that together with the special units avoid the problem of exponentially decaying error (Hochreiter and Schmidhuber, 1997).

4. Preliminary experimental results

In order to show the convergence of the LSTM machines we performed offline experiments. In a preliminary experiment using 10-fold cross-validation sets and 10 hidden nodes in the network, we obtained the results shown in the experiment 1 in Table 4. SSE denotes the averaged sum of squares error for test sets, which is the objective function minimized by the LSTM and is a good performance measure for regression problems. In the experiment 2, we only used feature vectors $\mathbf{s}_i = \mathbf{v}_i$, i.e., only containing polyflap poses. Because of the non-deterministic nature of a certain control command, slightly different behaviours are produced. We plan to use active learning techniques driven by e.g. curiosity for the selection of samples.

Exp.	Avg. epochs	Avg SSE	Samples
1	4700	0.03	500
2	5622	0.007	500

Table 1: Preliminary results

References

- Gibson, J. J. (1977). The theory of affordances. In Shaw, R. and Bransford, J., (Eds.), *Perceiving, Acting, and Knowing: Toward an Ecological Psychology*, pages 67–82. Lawrence Erlbaum.
- Graves, A. (2008). *Supervised Sequence Labelling with Recurrent Neural Networks*. PhD thesis, Technische Universität München.
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, pages 1735–1780.
- Jacobsson, H. (2006). The crystallizing substochastic sequential machine extractor - CrySSMEx. *Neural Computation*, 18(9):2211–2255.
- Oudeyer, P.-Y., Kaplan, F., and Hafner, V. V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(1).
- Roa, S., Kruijff, G. J., and Jacobsson, H. (2008). Curiosity-driven acquisition of sensorimotor concepts using memory-based active learning. In *Proceedings of the 2008 IEEE International Conference on Robotics and Biomimetics*, pages 665–670.
- Sloman, A. (2006). Polyflaps as a domain for perceiving, acting and learning in a 3-D world. In *Position Papers for 2006 AAAI Fellows Symposium*, Menlo Park, CA. AAAI.
- Sloman, A. and Chappell, J. (2005). The altricial-precocial spectrum for robots. In *Proceedings IJCAI'05*, pages 1187–1192, Edinburgh. IJCAI.

Modeling Emotional Development via Finite Topological Spaces and Stratified Manifolds

Lee Rudolph*

Li Han*

Eric Charles**

*Department of Mathematics and Computer Science
Clark University
Worcester, MA 01610 USA

**Department of Psychology
Pennsylvania State University
Altoona, PA 16601 USA

Abstract

Much human experience (particularly experience mediated by language) appears to be intrinsically finitistic. In contrast, many phenomena in the experienced world are conventionally measured and modeled by intrinsically infinite continua, based on the system \mathbb{R} of real numbers but including much more complex spaces. We briefly describe how to combine *finite topological spaces* with *stratified manifolds* into a conceptual and analytic tool with diverse applications. To illustrate the use of this tool, we state a theorem showing how adopting any “circumplex model of affect” forces strong topological restrictions on continuous families of schematic stimuli that emotional response. This purely mathematical theorem has strong practical implications for studying human or robotic emotions—for design and interpretation of experiments as well as for theory construction.

1. Introduction

The most telling point in London’s critique (London, 1944) of Lewin’s *Principles of Topological Psychology* (Lewin, 1938) is the observation that “Lewin in reality does not utilize one single theorem of topology. Always there is an interminable use of a few definitions ripped out of their proper context” (p. 287). Having made that point, however, London himself falls deep into error—and then shows the way out.

In topology notions of connectivity and continuity, between which a very close relation exists, imply an infinitely structured space (that is to say, an infinite set of points)—a fact which Lewin acknowledges is presupposed by topology. [...] Lewin goes on further to say that, as far as he knows, mathematics has not yet followed up Riemann’s suggestion that it is not necessary logically that spaces should be infinitely structured. But finite spaces and geometries have been developed and investigated for some years. [...] [I]t would be with

such finitely structured spaces that a geometrical or spatialized coordination of psychology might be attempted [...]. (288–289)

London’s error is to assert that “connectivity and continuity [...] imply an infinitely structured space” (like \mathbb{R} and constructions based upon \mathbb{R} , *e.g.*, Euclidian spaces, manifolds, metric spaces, ...). The way out is to become aware that not only do they not mathematically “imply” any such thing but in fact there is a rich universe of “finitely structured”—indeed, *finite*—topological spaces, in which certainly all the “connectivity” (and arguably all the “continuity”) of, *e.g.*, the universe of compact differential manifolds is realized. For background on finite topological spaces, see Stong (1966), McCord (1966), and Barmak & Minian (2008).

2. Circumplex Models of Affect

Russell (1980) describes “a circumplex model” as both as a way psychologists can represent the structure of affective experience [...] and as a representation of the cognitive structure that laymen utilize in conceptualizing affect. (Russell, 1980, p. 1161)

That particular model features eight “affective concepts [...] in a circle in the following order: pleasure (0), excitement (45), [...], sleepiness (270), and relaxation (315)” (*ibid.*). Other circumplex models of affect have different numbers of “affective concepts” and/or different names (see, for example, Fig. 1, which illustrates a 12-affect circumplex investigated by Yik, Russell, and Steiger); but—although we have not reviewed the entire, very large, literature—it appears that, like Russell’s, all (or the vast majority of) these models presuppose, whether as an ideal form or as a concrete construction literally embedded in some reified Cartesian plane described by real ‘dimensions’ (as psychologists call what most mathematicians would call ‘linear coordinates’) bearing names like “pleasure” or “activation”, a circle with infinitely many points along which a small, finite number of “affective concepts” are situated more or less precisely (for instance, by associating them with angular measures between 0 and 360 degrees).

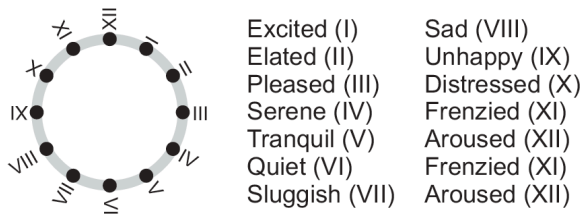


Figure 1: A 12-affect circumplex of Yik, Russell, and Steiger (adapted from a figure in Yik & Russell, 2004).

As Fig. 1 makes clear, any such construction leads directly to a *stratification* of the circle, *i.e.*, a partition into a finite set of points interpolated by a finite set of (open) arcs, each arc having two of the points as its endpoints. A general topological construction (Rudolph, in preparation, will give details) converts that stratification to a finite topological space. For instance, the Yik–Russell–Steiger 12-affect circumplex becomes a 24-point topological *space of strata*; each of the 12 “sharply defined” points on the circle becomes a so-called “closed point” in the finite topological space, and each of the 12 open arcs becomes a so-called “open point” there. This maneuver can be viewed as a simple example of how the notion of “emergence of meanings through ambivalence” (Abbey and Valsiner, 2005) can be mathematically formalized; the ambivalence in these cases is between ‘sharp’, named “affective concepts” (modeled by 0-dimensional strata) and ‘fuzzy’, unnamed but not meaningless “affective concepts” (modeled by 1-dimensional strata). The distinctive feature of the strata-spaces of circumplexes is that each point, ‘sharp’ or ‘fuzzy’, has exactly two ‘immediate neighbors’ (always of the other sort); for instance, in Fig. 1, “Pleased” is neighbored by “ambivalent between Pleased and Elated” and “ambivalent between Pleased and Serene”, while “ambivalent between Sluggish and Sad” is neighbored by “Sluggish” and “Sad”.

A fundamental theorem about circumplex models can be roughly stated as follows (Rudolph, in preparation, includes a precise statement and proof). Suppose that a particular circumplex model C of affect is valid, and consider any continuous family M of stimuli that reliably evoke affect. (For instance, M might be the space of affectively-meaningful facial configurations, considered as a subset of a Euclidean space of dimension at least 21, by identifying a configuration with its vector of continuous degrees of actuation of up to 21 muscle groups; compare Wehrle *et al.*, 2000.) For a stimulus S in M , let $F(S)$ denote the affect (sharp or two-way ambiguous) evoked by S . Then at least one of three things happens. (1) F is *discontinuous*: for some two arbitrarily close stimuli S_1, S_2 in M , $f(S_1) \neq f(S_2)$. (2) F has “*dead ends*” (local extrema): there is a region U in M and an affect A in C having neighboring affects B and

Z , such $f(S) = A$ for all S in M , while for all S' sufficiently close to M , $f(S') = B$ (not Z). (3) M has an unpatchable “hole” in it—no matter how you extend the ‘meaningful’ stimuli in M to a larger set of stimuli that can be described by a set of entirely independent variables, there will be (lots of!) these new stimuli that are ‘meaningless’. (For instance, assuming a circumplex model is valid, there *must* be many vectors of degrees of actuation of facial muscles that produce configurations that are not affectively meaningful; which is observed.)

Acknowledgements

The authors thank the referees for their comments. This work was partially supported by NSF awards IIS-0713335 (Han and Rudolph) and DMS-0308894 and BCS-0420939 (Rudolph) and NICHD Shriver Postdoctoral Individual National Research Service Award 5F32HD050037-02 (Charles).

References

- Abbey, E. and Valsiner, J. (2005). Emergence of meanings through ambivalence. *Forum Qualitative Sozialforschung*, 6(1).
- Barmak, J. A. and Minian, E. G. (2008). Simple homotopy types and finite spaces. *Adv. Math.*, 218(1):87–104.
- Lewin, K. (1938). *Principles of Topological Psychology*. Duke University Press, Durham, NC.
- London, I. D. (1944). Psychologists’ misuse of the auxiliary concepts of physics and mathematics. *Psychol. Rev.*, 51:266–291.
- McCord, M. C. (1966). Singular homology groups and homotopy groups of finite topological spaces. *Duke Math. J.*, 33:465–474.
- Rudolph, L. (in preparation). The hole in emotion space: topological consequences for circumplex models of affect.
- Russell, J. A. (1980). A circumplex model of affect. *J. Person. Soc. Psychol.*, 39:1161–1178.
- Stong, R. E. (1966). Finite topological spaces. *Trans. Amer. Math. Soc.*, 123:325–340.
- Wehrle, T., Kaiser, S., Schmidt, S., and Scherer, K. R. (2000). Studying the dynamics of emotional expression using synthesized facial muscle movements. *J. Person. Soc. Psychol.*, 78(1):105–119.
- Yik, M. S. M. and Russell, J. A. (2004). On the Relationship Between Circumplexes: Affect and Wiggins’ IAS. *Multivariate Behav. Res.*, 39(2):203–230.

Selective integration based on subjective consistency facilitates simultaneous development of vocal imitation and lexicon acquisition

Yuki Sasamoto

Yuichiro Yoshikawa

Minoru Asada

Asada Synergistic Intelligence Project, ERATO, JST
Graduate School of Eng., Osaka University
2-1 Yamadaoka, Suita, Osaka, 565-0871 Japan
yuki.sasamoto@ams.eng.osaka-u.ac.jp, yoshikawa@jeap.org, asada@jeap.org

1. Introduction

In human-language communication, vocalization is one of the most efficient channels because humans can share a large lexicon within a short duration of time. Human infants start to understand caregiver's words from eight months of age and produce their first word by the end of their first year. Meanwhile, they exhibit mimicry of adults' single vowels by eight months of age as well as that of adults' strings of consecutive vowels by 14 months of age (Jones, 2007). Therefore, the development processes of lexicon acquisition and vocal imitation seem to overlap each other. Furthermore, we conjecture that these processes might facilitate each other. For example, the ability of vocal imitation could help infants to vocalize unheard words, and the knowledge of a lexicon and its correspondence to objects could help them to imitate partially inaudible words. What kinds of mechanisms underlie the developmental processes of such complementary abilities?

Synthetic studies have attracted wide attention as one of the most promising approaches to resolving such questions of developmental mechanisms (Asada et al., 2009). In previous work, the development of lexical acquisition (Roy and Pentland, 2002, Yoshikawa et al., 2008) and that of vocal imitation (Kanda et al., 2007, Miura et al., 2007) have been modeled as learning processes. However, such studies on these abilities have generally been conducted separately, and thus their interaction has remained unexplored.

In this paper, we propose a method for simultaneous development of vocal imitation and lexicon acquisition through mutually associative cross-modal mapping using subjective consistency. Subjective consistency of a signal from each pathway in the mapping is calculated by its proximity to those from others and used as a contribution rate in integrating signals. The integrated vector is used as a learn-

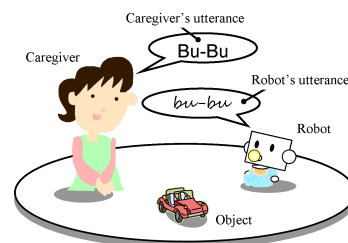


Figure 1: Assumed environment of caregiver-robot interaction

ing signal that is expected to ignore errors of the caregiver along with the learning progress of other pathways.

2. Assumptions

A robot and a caregiver take turns in an environment that includes objects (Fig.1). In each step, it looks at either the caregiver or any of the objects and decides whether to utter. Then, the caregiver selects either of three types of behaviors: replying, showing, and describing. The behavior of the caregiver is successful based on the pre-determined probability of each type (p_R , p_S , p_D).

Through such interactions, it learns connection-weight matrixes between nodes in two different layers, namely those between one's own phonemes and the caregiver's phonemes \mathbf{W}^{uv} (imitation mapping), those between the caregiver's phonemes and objects \mathbf{W}^{vo} (word-listening mapping), and those between objects and one's own phonemes, \mathbf{W}^{ou} (word-producing mapping) (Fig.2).

3. Selective combination based on subjective consistency

We propose a method of selective combination to create a reliable learning signal based on subjective consistency. Let \mathbf{x}^i and \mathbf{x}^j be external input vectors to the i -th and j -th layers and \mathbf{x}^{ij} be a direct prediction vector of \mathbf{x}^j from \mathbf{x}^i by the mapping with \mathbf{W}^{ij} . Furthermore, suppose that there

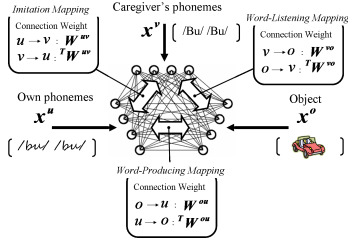


Figure 2: Mutually associative cross-modal mapping

is another layer labeled by k that receives a vector from the i -th layer and outputs an indirect prediction vector \mathbf{x}^{kj} to the j -th layer. Three vectors, \mathbf{x}^j , \mathbf{x}^{ij} , and \mathbf{x}^{kj} , are regarded as potentially having information for learning \mathbf{W}^{ij} . A integrated vector \mathbf{x}^j is calculated as $\mathbf{x}^j = f(\mathbf{x}^j, \mathbf{x}^{ij}, \mathbf{x}^{kj}) = \lambda_j \mathbf{x}^j + \lambda_{ij} \mathbf{x}^{ij} + \lambda_{kj} \mathbf{x}^{kj}$, where λ_n ($n = j, ij, kj$) represents a subjective consistency of each vector. Here, each vector's subjective consistency indicates how close it is to other vectors, and it is calculated by $\lambda_n = \exp\left(-\prod_{l, l \neq n} \|\mathbf{x}^n - \mathbf{x}^l\|/\sigma^2\right) / \sum_{m, m \in \{i, ij, kj\}} \exp\left(-\prod_{o, o \neq m} \|\mathbf{x}^m - \mathbf{x}^o\|/\sigma^2\right)$, where σ is the parameter of sensitivity for consistencies. The integrated vector is used as a learning signal. It is expected not only to basically bias the learning of \mathbf{W}^{ij} to predict the current signal \mathbf{x}^j from \mathbf{x}^i but also to ignore \mathbf{x}^j when it seems to involve errors of the caregiver along with the learning progress of the other pathways (\mathbf{W}^{ij} and \mathbf{W}^{kj}).

4. Simulation

We conducted a series of computer simulations to show the validity of the proposed method for mutually associative cross-modal mappings. We assumed that the number of objects was 39 and the number of phonemes was 37. The parameter *sigma* was empirically set to 1.0 for good performance. We compared the learning performances of the proposed method (hereinafter *proposed*) to those of another method without integration based on subjective consistency for updating the connection matrix (hereinafter *direct*).

We ran 10 sets of simulation with 200,000-step interaction for different sets of parameters p_R , p_S and p_D . These parameters were set to be equal with each other and varied from 0.2 to 1.0. Figure 3 shows the transitions of the average performance of each mapping over different sets of simulation, where $p_R = p_S = p_D = 0.2$. Figure 4 shows the final performances of the entire learning process with respect to the success rate of the caregiver's behaviors. This is calculated from the average performances among all three mappings. We can see that the performance of both methods is high in the case of a high success rate of the caregiver's behaviors. However, the

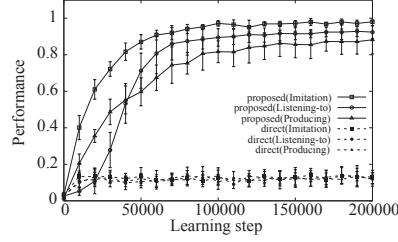


Figure 3: Transition of performance of each mapping

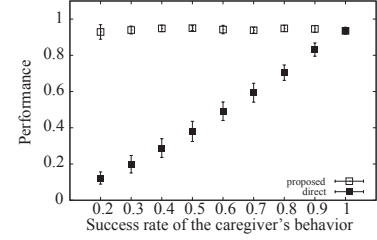


Figure 4: Final performance with respect to the success rate of a caregiver's behaviors

performances of *direct* (filled symbol) becomes worse along with the decrease in the success rate, while that of *proposed* (blank symbol) remains high against the decrease of the success rate.

5. Conclusion

In this paper, we proposed a method to combine several sources of a learning signal for mutually associative cross-modal mappings, which is formed by an external input and internal outputs from possible streams of mapping within it. The subjective consistency of each signal, which evaluates how close it is to other signals, is used to weight it to calculate the combined signal. The proposed method makes it possible to successfully ignore the external input in the case where the caregiver fails to give examples of correct mapping, which is presumed to be typical in real caregiver-infant interaction.

References

Asada, M., Hosoda, K., Kuniyoshi, Y., Ishiguro, H., Inui, T., Yoshikawa, Y., Ogino, M., and Yoshida, C. (2009). Cognitive developmental robotics: a survey. *IEEE Trans. on Autonomous Mental Dev.*, 1(1):12-34.

Bates, E., Dale, P. S., and Thal, D. (1995). *The Handbook of Child Language*, chapter 4: Individual Differences and their Implications for Theories of Language Development, pages 96-151. Blackwell Publishing.

Jones, S. S. (2007). Imitation in infancy the development of mimicry. *Psycho. Sci.*, 18(7):593-599.

Kanda, H., Ogata, T., Komatani, K., and Okuno, H. G. (2007). Vocal imitation using physical vocal tract model. In *Proc. of the 2007 IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, pages 1846-1851.

Miura, K., Yoshikawa, Y., and Asada, M. (2007). Unconscious anchoring in maternal imitation that helps finding the correspondence of caregiver's vowel categories. *Advanced Robotics*, 21:1583-1600.

Roy, D. K. and Pentland, A. P. (2002). Learning words from sights and sounds: a computational model. *Cognitive Science*, 26(1):113-146.

Yoshikawa, Y., Nakano, T., Ishiguro, H., and Asada, M. (2008). Multimodal joint attention through cross-facilitative learning based on μx principle. In *Proceedings of the 7th International Conference on Development and Learning*, pages 226-231.

Facing the Homunculus: On Innate Structures for Vision of Assistive Robots*

Matthias J. Schlemmer Markus Vincze

Vienna University of Technology, Automation and Control Institute
Gusshausstrasse 27-29/E376, 1040 Vienna, Austria

Abstract

This work presents a follow-up to our 2007 EpiRob paper in which we showed the philosophical foundations to our approach to vision for “cognitive robotics”. Whereas our emphasis by that time lay on the notion of the thing-in-itself, this time we concentrate on the discussion on developmental issues involved in our undertaking to arrive at a specific form of a cognitive robot: an assistive robot in a home setting. This work outlines major research issues tackled.

1. Research questions

A far goal, such as a cognitive assistive robot serving at the user’s side in a home environment, involves the following questions: What is “cognition”? Can we locate a common ground of argumentation between different disciplines involved in its discussion? What are useful interdisciplinary starting points?

Perception – particularly a remote mode such as visual perception – is of utmost importance for a cognitive agent working in interaction with its environment. Hence, for the special case of vision for robotics, there are more specific questions entailed, e.g.: Which “kind” of computer vision output can be used and what do we do with it?

We will later see that we chose a *functional approach* which itself, however, entails a lot more questions: What is the minimal set of functions that seem necessary for situated perception in robots? What is the glue that holds them together? What is the “knowledge” of the system. Especially the latter question already indicates that developmental considerations become crucial in this discussion, as at least the knowledge of any *living* cognitive system is built up and not pre-given. For assistive robots, we relax this constraint and argue for a concise definition of innate *structure* paired with pre-given concepts.

*This work has been supported by EU-Projects XPERO (contract no. 6029427) and CogX (contract no. 215181) as well as by the Austrian Science Foundation (grant no. S9101).

2. Our approach

Working on perceptual capabilities of cognitive robots means understanding the “homunculus” inside the cognitive system that looks at the images and *interprets* them, guiding action and knowledge enrichment. Our stance is that it needs to use a lot of knowledge which is not in the data itself. This is what we refer to as the important and difficult bridge from quantitative data (delivered by the vision techniques) and qualitative information that fits the knowledge structures of the agent. This guiding knowledge is built up by the system itself, but it needs to *start* with something. For assistive robots, we propose, this amount of predetermined information might be quite high.

From a philosophical point of view, we are necessarily facing an *inseparability* of ontology and epistemology. This is due to the fact that humans design robots and are thus defining their *ways* of accumulating knowledge and hence also *what* they see and learn. This seems trivial, yet is crucial to be thought of when talking about “development” or nature vs. nurture.

Considering a Kantian approach, we may talk about the needed *a priori concepts* of which Kant himself named at least four: space, time, causality, and number. For home robots, we can expand this list much further, for developmental robotics less further. This *trade-off* is an important design choice. In any case, a priori concepts must be defined carefully, which has sometimes been made explicit, e.g., by (Hamlyn, 1990): “Experience can make us see that certain things are so. We may not be able to see them in that way unless we have the concepts which are presupposed in so seeing them.”¹

To put it in a nutshell, our stance within this discussion is that for research on assistive robots it is totally fine to pre-give quite a lot of “innate knowledge” instead of letting the robot evolve and develop totally by its own. The major open question then remains how it is possible that the agent learns *on top of* this pre-given knowledge all the stuff it needs to know in order to accomplish his very task: to

¹Cited after (Russell, 1999).

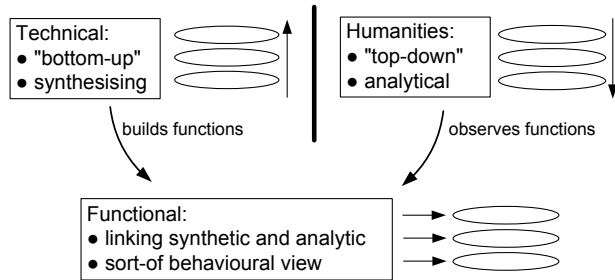


Figure 1: Our functional interdisciplinary approach

serve the human on his/her side. Consequently, the concepts that are pre-defined need to be stored in a manner that further information pieces can be seamlessly incorporated.

3. Cognitive Functions for Robots

We are following a *functional* approach to robotics in order to account for the interdisciplinary nature of our research goal. As shown in Figure 1, this is where the analytical approach of the humanities and the synthetic one of the technical sciences can meet: analysing which “cognitive functions” are observable in living cognitive agents may lead to the insight which are equally needed in an artefact. This helps us synthesising them and leads to the possibility to apply behaviouristic methods for assessing functional fit. The goal is to circumvent the ill-posed “implementation of intelligence” that has led artificial intelligence research astray.

To be more definite, we currently have defined the following functions to be crucial for vision in an autonomous, intelligent, cognitive assistive robot: intentionality (i.e., task-guidedness), prediction (anticipating what is likely to be seen next), abstraction (seeing the concept behind an instance), generalisation (learning abstract descriptions), and symbol binding (connecting seen things to one’s ontology)².

These functions work – along with a non-perceptual ones – on a shared *ontology* using a common representation format. Figure 2 shows our vision of this ontology which contains “vision-near” as well as more abstract concepts. The relations are giving the semantics, e.g., a cup that *can_be_moved* and *is_a_container* directly gives “affordance”-information and enriches thus the knowledge about the cup for concrete situations.

The amount which of these concepts and relations are developed and which are pre-given is not only a matter of design but also of the intended niche of the robot. With specific regard to vision problems, a feasible approach is what we started with already in (Schlemmer et al., 2007): Pre-giving cer-

²These are the same cognitive functions – although not explicitly termed this way – that we have already touched in (Schlemmer et al., 2007).

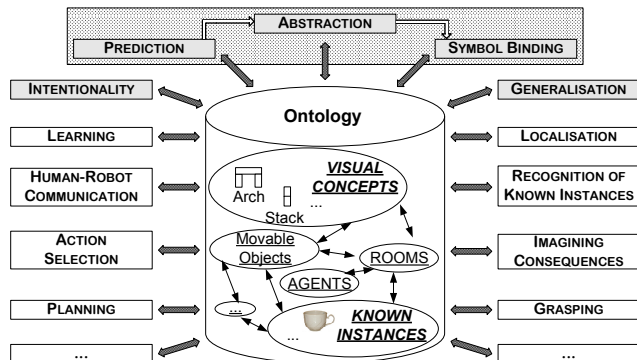


Figure 2: Various cognitive functions connect to a common ontology.



(a) Closures (b) Blobs after processing. (c) Best arch hypothesis. (Zillich, 2007).

Figure 3: Detection of a door by detecting its surrounding “arch” (door frame).

tain structural relations (on_top_of, left_of,...) and using simple blobs in the image that can be described by these relations. This leads to higher-order *object concepts* that describe structures like arch or stack. They, in turn, can be used to describe concrete instances of actually (semantically) much richer concepts, such as door. Figure 3 shows the detection of a door by its surrounding arch.

Such a lax hierarchical composition of object concepts bails rich potential for an assistive robot in interpreting the world of humans – which is exactly its ecological niche.

References

Hamlyn, D. (1990). *In and Out of the Box: On The Philosophy of Cognition*. Blackwell, Oxford, United Kingdom.

Russell, J. (1999). Playing a passing game: Rationalism, empiricism, and cognitive development. In Bennett, M., (Ed.), *Developmental Psychology – Achievements and Prospects*, chapter 14, pages 253–271. Psychology Press, Philadelphia, PA, USA.

Schlemmer, M. (2009). *Getting Past Passive Vision – On the Use of an Ontology for Situated Perception in Robots*. PhD thesis, Vienna University of Technology, Austria.

Schlemmer, M., Vincze, M., and Favre-Bulle, B. (2007). Modelling the thing-in-itself – a philosophically motivated approach to cognitive robotics. In *Proc. of the 7th Intl Conf on Epigenetic Robotics*, LUCS 134, pages 149–156.

Zillich, M. (2007). *Making Sense of Images: Parameter-Free Perceptual Grouping*. PhD thesis, Vienna University of Technology, Austria.

History of Usage of Piaget's Theory of Cognitive Development in AI and Robotics: a Look Backwards for a Step Forwards

Georgi Stojanov
The American University of Paris
gstojanov@aup.fr

It appears that the topic of *cognitive development* went mainstream in AI and robotics in the last 10 years or so¹. As a result, Jean Piaget's ideas on child cognitive development and his genetic epistemology are receiving an unprecedented interest.

On the other hand a brief *historic* research clearly shows that his ideas have been introduced and re-introduced to the AI community on several occasions and almost always independently during the last 40 years.

In this paper I report some preliminary results of an ongoing research project the aim of which is to identify and critically compare the approaches to *computational modeling of cognitive development* directly inspired by Jean Piaget's genetic epistemology. Some of these results presented here were presented during the two symposia that I have organized within the annual Jean Piaget Society meetings 2004 and 2009.

The paper is organized in the following way: for approximately every decade ('70s through present) I present a sample or two of relevant research works with a short comment for each one. In the second part I point to some of the commonalities among them and draw attention to areas where there has been little progress.

Papert, (Papert, 1963), Boden (Boden, 1978) and Rosenberg (Rosenberg, 1980) all pointed to the potential mutual benefit that AI and Piagetian theory can get from each other: AI (with its methodologies) can complement Piaget's theory which often relies on notions that lack specific details. In return AI can get advantage from Piaget's *big picture* framework of cognitive development.

One of the first attempts to build programs that simulate infant behavior in variety of Piagetian tasks (like *class inclusion* and *conservation of quantity*) is the work of Klahr and Wallace (Klahr and Wallace, 1972, 1973, 1976). Along similar lines Baylor et al (1973) and Young (1976) developed simulations for the *seriation* task where children are asked to order a set of objects along certain attribute (e.g. length or weight). All these models used *production rule systems* (i.e. a list of Condition-Action pairs) to model infant behavior at different *stages* of development and at different *granularity*. Low level perception processes were not modeled, and the programs were given high level description of the problem space (e.g. the position and

attributes like color and length of the blocks in the seriation task simulator by Young). The creative work was to find and order the set of rules (e.g. If you see the biggest block THEN put it first in the series) which, starting from some initial configuration, will come (or not) to the goal configuration, exhibiting behavior similar to the children of particular stage.

What is curious for these early works is the virtual absence of reference to the notions scheme, adaptation, and assimilation, all central to Piaget's theory. Much in the spirit of the then dominant *information processing* paradigm (within the *cognitive turn* in psychology), some of the preferred terms were *knowledge structures*, *information processing*, *discrimination*, *generalization*, and the like.

This changed during the '80s when Gary Drescher (Drescher, 1985, 1987, 1991) undertook probably the most ambitious attempt until then to give a computational model for the *schema* learning mechanism, and used this mechanism in a simulated agent which would learn a useful representation of its environment with no innate knowledge. Drescher's simulation included a discrete 2D microworld, baby's body, her visual field (foveal and peripheral), one hand, objects, and a set of innate primitive actions (*grasp*, *move-hand-backwards*) and primitive perceptual items (*hand-at-1-1*, *hand-closed*, *hand-grasping-something*). Schemas were represented as triplets (context/action/consequence) where context and consequence can be conjunctions or disjunctions of (possibly negated) items. Drescher's main contributions were: a) a statistical technique called *marginal attribution* which learns *reliable* schemas that for given context can predict the consequences of the actions in the microworld; and b) introduction of *synthetic items* which which subsume several primitive (or synthetic) items.

Quite independently a research group in Geneva called CEPIAG (for Cybernétique Epistémologie Psychologie Intelligence Artificielle Génétiques) has produced a considerable body of research during the early 1990. To my knowledge, they reported the first physical implementation of a constructivist agent. They also provided a minimal social context for the developing agent by including a second *mother* robot. Unfortunately, their work has largely remained unpublished save some internal publications and non English language local conferences (Schachner, 1996; Schachner et al., 1999; Ducret et al., 1999). The robot could move around on two wheels and had needs to be satisfied (like being hungry or sleepy). The schema in their implementation comprised three parts: *sensorium*,

¹ As witnessed by the emergence of several conferences like Epigenetic Robotics, the International Conference on Development and Learning, as well as several workshops on Developmental Robotics within AAAI symposia. Most of them were initiated at the turn of the new millennium.

motivarium, and *motorium*. Depending what robot's current needs were (*motivarium*) it would try to *sense* (*assimilate*) the current situation (*sensorium*) and (in case of successful assimilation) would apply the appropriate actions (*motorium*). Learning consisted of (among other things) in *getting the right ordering of execution of schemas* in order for a need to be satisfied. New schemas could be produced by the existing ones by: a) *differentiation* (e.g. original schema with only *motorium* specified (e.g. *go_forward*) gets closer to a light source *or* makes the robot touch an object in front of it) when *motivarium* and the *sensorium* part will be specified depending on the consequence of the execution of the *motorium* part, b) *assimilation of two existing schemas* (e.g. one schema has the action *turning-left* in the *motorium* part and another *turning-right*) when merged they will produce a new schema which will move the robot forward; and c) by introduction of *meta-schemas* where the *motivarium* and *sensorium* part will be specified (e.g. the robot is hungry and want to go to a state where it is not hungry) and the learning mechanism will have to find one or several existing schemas that would move the system from the actual state (hungry) to the desired one (not hungry).

The focus of my group was to come up of a mechanism where a simulated agent in a 2D maze-like environment would autonomously learn a useful representation of environment out of its interactions with the environment (Stojanov et al., 1996; Stojanov, 2001; Stojanov et al., 2006). The agent is able to perform 4 elementary actions (moving forward, backward, left, and right) and had only a touch sensor. Initially the agent had only one *schema* comprised of random sequence of the elementary actions (e.g. FFRFFFLFFFRBFFFF). The agent tries to perform the whole sequence but environmental constraints would make it impossible at certain point (e.g. going F when in front of an obstacle). In that case the agent would skip the impossible action(s) and continue with the next possible one. The actually executed subsequence (the *enabled subschema*) would then be memorized, together with the link to the previous enabled subschema. We called this process *accommodation* and the result of it was a repertoire of enabled subschemas, with their contingency links. The environment would be completely assimilated if at every moment (after having executed a particular enabled subschema) the agent could find a sequence of enabled subschemas that would bring it to a desired place in the environment.

After 2000 there were dozens of researchers that proposed their own version of Piagetian inspired constructivist agents who have suggested computational mechanism for schema based learning, including assimilation and accommodation mechanisms. For example, Perotto (Perotto et al., 2007) and Guerin (Guerin & McKenzie, 2008) worked with simulated agents and used a variant of Drescher's schema construct.

Some tentative conclusions: first, a trend can be observed where computational models move from modeling infant's behavior during some rather high level tasks towards trying to simulate earliest periods (beginnings of the sensory motor stage) of the cognitive development.

Second, it seems that several researchers starting from quite abstract and somewhat loosely defined Piagetian notions like the schema mechanism, equilibration, accommodation/assimilation and the like, independently came up with rather similar computational mechanisms. For example, most of the above computational models of a schema use a variant of a data structure of the form (S1-A-S2) where S1 is the sensory input before action A is applied and S2 is the resulting consequence. They can be all regarded as action based, future oriented representations of the agent's world (cf. Bickhard, 2005). Other researchers, starting from fairly different assumptions arrived to similar conclusions regarding these properties of mental representations (e.g. Pezullo, 2008; Grush, 2004).

Third, most of them stressed the importance of *open-ended* learning and hence the importance of *modeling the inner value system* and phenomena like *curiosity* or *epistemic hunger*.

Fourth, in all of the above systems, the process of development seems to be driven predominantly by the environmental input, leaning thus towards the empiricist end of the nativism-empiricism epistemological specter. The *knowledge structures* that arise in this way are unavoidably a deterministic outcome of the agent-environment interaction. This precludes any creative process where, say, by analogy, an agent would extend its knowledge from one domain to another. We have discussed some of these issues in (Stojanov et al. 2006; Kulakov&Stojanov, *forthcoming*). In Piagetian parlance this would mean that most of the above presented models account primarily for the *empirical abstraction* and neglect the *reflective abstraction* which is crucial for development and creativity. Briefly, by empirical abstraction some quality (e.g. weight or color) is abstracted from an object. On the other hand, reflective abstraction is about reorganization of existing schemas and their projection on a higher plane. (see Kitchener, 1983, pp. 61-65, for informative discussion of empirical and reflective abstraction as well as Campbell & Bickhard, 1993 discussion on the *knowing levels*). So far, only limited schema manipulation mechanisms seem to be proposed (Drescher's synthetic items or CEPIAG's meta-schemas).

Finally none of the (so far) reviewed models tried to tackle the effects of maturation and biological growth on the cognitive development.

References

Please find all references in the paper here: <http://tinyurl.com/StojanovEpiRob09>

A minimum relative entropy principle for the brain

Antoine van de Ven

Fontys University of Applied Sciences
Postbus 347, 5600 AH Eindhoven, The Netherlands
Antoine.vandeVen@fontys.nl

Abstract

The principle of minimum relative entropy is proposed as a general fundamental principle that could be used by the brain to do inference and update beliefs about the world. It originates from information and probability theory, but we relate it to the brain, to the concept of surprise and to a minimum free-energy principle that has already been proposed for the brain. The measure of surprise that is based on relative entropy (Bayesian surprise) is compared with another definition of surprise (Shannon surprise) that is used by Friston for a minimum free-energy principle. Theoretical and experimental justifications are given to propose to use Bayesian surprise as a better and more natural definition of surprise. It can be used as a novel way to quantify surprise or related concepts in developmental robotics. This can then be used in implementations of intrinsic motivations like curiosity to drive exploration, interactive learning and autonomous mental development.

1. Introduction

The concept of intrinsic motivation is important in developmental psychology because it seems necessary for open-ended cognitive development. In the field of developmental robotics one goal is to understand and model these intrinsic motivations. It has been said (Oudeyer et al., 2007) that the challenge is to operationalize and quantify the concepts behind words like "surprise" (Ranasinghe and Shen, 2008) and "novelty" (Huang and Weng, 2002) which are important to model and implement intrinsic motivations.

In this paper a definition and implementation of surprise is suggested that is based on relative entropy. First the principle of minimum relative entropy is introduced by discussing the theoretical foundations of a general universal method for inference. Then it is shown to be related to and that it confirms a definition of surprise by Itti and Baldi (Itti and Baldi, 2009) that they call Bayesian surprise. After that the theory is compared with Friston's (Friston, 2009) minimum free-energy principle

and another definition of surprise. In the conclusion the advantages of relative entropy and opportunities for the field of developmental robotics are discussed.

2. Theoretical foundations

It is possible to derive a general universal method for inference on the basis of three axioms (Giffin, 2008). An important assumption is the principle of minimal updating: beliefs should be updated only to the extent required by the new information. This is incorporated by a locality axiom, and the other two axioms are only used to require coordinate invariance and consistency for independent subsystems. By eliminative induction this singles out the logarithmic relative entropy as the formula to minimize. This way the Kullback-Leibler Divergence (KLD) (Kullback and Leibler, 1951) has been derived as the only correct and unique divergence to minimize. Other forms of divergences and relative entropies in the literature are excluded.

It can be seen as a confirmation of the Principle of Minimum Discrimination Information (MDI) as proposed by Kullback. It states that given new facts, a new distribution should be chosen which is as close to the original distribution as possible so that the new data produces the smallest possible information gain. This means the KLD can also be used to measure information gain. In another form it is called the method of Maximum relative entropy, or Maximum Entropy (ME) (Giffin, 2008). The only difference is a minus-sign. Note that this is not equal to the MaxEnt method, which also has been called Maximum Entropy. To avoid this possible confusion and because we use the form of the KLD without the minus-sign, we will call it the principle of minimum relative entropy (PMRE).

It has been shown (Giffin, 2008) that this principle is capable of producing every aspect of orthodox Bayesian inference (which allows arbitrary priors) and MaxEnt (which allows arbitrary constraints), and can also process both forms simultaneously, which Bayes and MaxEnt cannot do alone.

This principle was derived by only using mathematics, but we propose that it could be a principle that is used by the brain to adjust its beliefs about the world and to do inference. Because energy effi-

ciency is important for the brain, the axiom of minimal updating makes this principle biologically more plausible than many other algorithms.

3. Bayesian surprise

To further support the principle and to relate it to the brain we now refer to experiments that have been done. A definition of Bayesian surprise has been proposed (Itti and Baldi, 2009) that is equal to the KLD between the prior and posterior beliefs of the observer. In experiments they showed that by calculating this they could predict with high precision where humans would look. This formula and definition was found to be more accurate than all other models they compared it with, such as Shannon entropy, saliency and several other measures.

In their derivation they picked the KLD as the divergence to define Bayesian surprise by referring to the work of Kullback. Although we agree it would also have been possible to pick another divergence as a measure, because the KLD is just one out of one class of divergences called f-divergences. They didn't explicitly exclude all other possibilities. The benefit of the derivation of the PMRE is that it uniquely selects the KLD. So in this way the PMRE helps to better select and confirm this definition of Bayesian surprise.

4. The free-energy principle

In the field of neuroscience, the minimum free-energy principle has been proposed (Friston, 2009) as a fundamental principle to explain many things about the brain. Friston uses the minimization of Shannon surprise as fundamental principle and as a starting point. The principle of minimum free-energy is related to that, because mathematically free-energy is always an upper bound to Shannon surprise. Bayesian surprise is a measure between two distributions, like the prior and posterior beliefs. Shannon surprise is different because it is only based on one probability distribution.

Free-energy can be expressed as the sum of a KLD and Shannon surprise. For perception the minimization of free-energy is similar to the PMRE, because the extra term with Shannon surprise has no influence on the solution. But when applying the minimum free-energy principle to actions, as Friston proposes, this extra term will influence the results, so the two principles are not equivalent in all ways.

5. Conclusion

It has been shown that the PMRE has a very solid theoretical foundation and that the principle of minimal updating makes it more biologically plausible than many other techniques or algorithms. Much

of the impressive work by Friston can also be seen as support for this principle because the principle of minimum free-energy is similar in many ways.

The biggest difference is the use of different definitions of surprise. Being aware of the experiments by Itti and Baldi (Itti and Baldi, 2009) Friston stated: "it remains an interesting challenge to formally relate Bayesian surprise to the free-energy bound on (Shannon) surprise." (Friston, 2009) In our approach we don't have this problem or challenge because we only use Bayesian surprise as the natural measure of surprise. An overview of computational approaches for intrinsic motivations (Oudeyer and Kaplan, 2007) shows many different implementations and definitions, including ways to quantify surprise, but relative entropy hasn't been used for that yet. We propose to use Bayesian surprise to define and quantify surprise because it has a very good theoretical foundation and because experiments indicate that it is currently the best way to model human surprise. This could be useful to model and implement surprise and intrinsic motivations in the field of developmental robotics.

References

- Friston, K. (2009). The free-energy principle: a rough guide to the brain? *Trends in Cognitive Sciences*, 13(7):293–301.
- Giffin, A. (2008). *Maximum Entropy: The Universal Method for Inference*. PhD thesis, Massey U., Albany.
- Huang, X. and Weng, J. (2002). Novelty and reinforcement learning in the value system of developmental robots. Lund University Cognitive Studies.
- Itti, L. and Baldi, P. (2009). Bayesian surprise attracts human attention. *Vision Research*, 49(10):1295–1306.
- Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.
- Oudeyer, P. and Kaplan, F. (2007). What is intrinsic motivation? a typology of computational approaches. *Frontiers in Neurorobotics*, 1.
- Oudeyer, P., Kaplan, F., and Hafner, V. V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(2):265–286.
- Ranasinghe, N. and Shen, W. (2008). Surprise-Based learning for developmental robotics. In *Learning and Adaptive Behaviors for Robotic Systems, 2008. LAB-RS '08. ECSIS Symposium on*, pages 65–70.

CASA MILA: Cross-cultural and social aspects of multimodal interactions in language acquisition

Paul Vogt

Tilburg center for Creative Computing, Tilburg University
P.O. Box 90153, NL-5000 LE Tilburg
<http://www.paul-vogt.nl>, science@paul-vogt.nl

Abstract

This poster introduces the recently started CASA MILA project, which aims to study cross-cultural and social aspects of multimodal interactions aimed to establish joint attention and their impact on language development with infants and artificial agents. One of the objectives of the study is to collect data on the usage frequencies of three different types of joint attention and feed these into a simulation of the Talking Heads experiment in order to test mechanisms that would underlie the learning of word-meaning mappings. The poster will present some empirical findings from the pilot project that is currently been carried out in Mozambique.

1. Introduction

One of the biggest problems humans face when learning a language is identifying the meaning of words. The extent of this problem has been famously illustrated by (Quine, 1960), who sketched the situation of an anthropologist studying a – to him – unknown language. When a native speaker exclaims ‘Gavagai!’ at the moment a rabbit scurries by, the anthropologist notes that ‘gavagai’ means *rabbit*, but how can he be sure? Gavagai, Quine argued, could mean an infinite number of things, such as *undetached rabbit parts*, *dinner*, *animal with large ears* or even a completely unrelated event such as *it’s going to rain*.

Humans, especially children, are notoriously good at solving this problem. Various biases, constraints and (social) mechanisms have been proposed trying to explain how humans acquire word-meaning mappings. Examples include the whole object bias, shape bias, taxonomic bias, mutual exclusivity, principle of contrast, Theory of Mind and joint attention; for an overview see, e.g., (Bloom, 2000). All these biases, constraints and mechanisms serve to reduce the uncertainty of a word’s meaning.

The recently started CASA MILA project aims to study the effect of joint attention on language

acquisition in different cultural societies and simulated robots. In particular, the objective is to investigate how the usage-frequencies of various multimodal interactions (e.g., pointing gestures, gaze following, etc.) between infants and caregivers affect the speed of word learning. The multimodal interactions, which the project focuses on, are used to establish one of the three forms of joint attention proposed in (Carpenter et al., 1998):

Sharing / checking attention is the first form that emerges around the age of 9-10 months in a child’s development and occurs when a caregiver follows a child’s attention to an object, while both are aware of sharing attention (the child looks back and forth from object to the caregiver).

Following attention emerges second and happens around 10.5 months when a child’s attention is drawn to an object by the caregiver (e.g., through eye-gaze following).

Directing attention emerges thirdly at the average age of 12.6 months when a child directs the attention of the caregiver to an object.

Various studies have suggested that the onset of these types of attention, as well as, the frequency of joint attentional usage have an effect on the early vocabulary development of infants (Carpenter et al., 1998, Mundy et al., 2007).

One approach to study the effects of joint attentional usage on language development is by using developmental robotics, realised either in physical systems or in simulations. A recent study that used a simulation of Steels’ Talking Heads experiment (Steels et al., 2002) has shown that differences in the use of the three joint attentional mechanisms can lead to strong differences in vocabulary development, assuming a statistical language learning mechanism (Kwisthout et al., 2008). In this model the word-meaning mappings are acquired through *cross-situational learning* (Siskind, 1996), which is a statistical learning mechanism based on the co-variance in the occurrences of words and meanings across situations (or learning contexts). In the model it is

assumed that the three joint attentional mechanisms have different ways to reduce the learning context size. It was shown that, relatively speaking, checking attention yielded the largest reduction, following attention the second largest reduction and directing attention the smallest. Since cross-situational learning works faster when the learning mechanisms are smallest (Smith et al., 2006), the same ordering was found for the speed of vocabulary development (Kwisthout et al., 2008).

In this study, however, the frequencies by which the different joint attentional mechanisms were used was all or nothing. This is very unrealistic, because humans tend to use these mechanisms in various frequencies that differ individually (Mundy et al., 2007), and possibly cross-culturally as well (Keller et al., 2005). In order to computationally verify the validity of the underlying language learning mechanisms and the influence that joint attention can have on language development, it is desirable to predict the speed of vocabulary development using empirically obtained data on joint attentional usage and compare the outcome with relating development with human children (Vogt and de Boer, 2009).

The CASA MILA project aims to collect such empirical data in three cultures: one urban and one rural Changana speaking culture from Mozambique, and a Dutch speaking culture. The study will involve an observational study in which infants are videotaped in a natural setting in their native environment. The purpose is to collect the frequency distributions with which multimodal interactions occur with caregivers, siblings and others that lead to the three joint attentional forms at various stages during their development between the ages of 9 to 24 months. It is anticipated that there are differences between the three cultures regarding the frequencies with which the different forms of joint attention are used. The question remains whether such differences are also found in the speed of vocabulary development. By closely monitoring their language development, it will be possible to correlate the differences in joint attentional use with the development of joint attention.

The empirically obtained frequency distributions will then be used as input to an adaptation of the computational model used in (Kwisthout et al., 2008) that simulates the acquisition and evolution of language to investigate the effects that different distributions have on language development. Such simulations are helpful to investigate whether the learning mechanism used in the computer model predicts a similar development as the empirical findings. If this is the case, then the investigated learning mechanism is a likely candidate for the mechanism used by humans. If not, the imple-

mented learning mechanism probably needs revision.

Acknowledgements

The CASA MILA project is funded by the Netherlands Organisation for Scientific Research (NWO) through a VIDI grant awarded to the author.

References

- Bloom, P. (2000). *How Children Learn the Meanings of Words*. The MIT Press, Cambridge, MA. and London, UK.
- Carpenter, M., Nagell, K., Tomasello, M., Butterworth, G., and Moore, C. (1998). Social cognition, joint attention, and communicative competence from 9 to 15 months of age. *Monographs of the Society for Research in Child Development*, 63(4).
- Keller, H., Voelker, S., and Yovsi, R. (2005). Conceptions of parenting in different cultural communities: The case of West African Nso and Northern German women. *Social Development*, 14:158–180.
- Kwisthout, J., Vogt, P., Haselager, P., and Dijkstra, T. (2008). Joint attention and language evolution. *Connection Science*, 20:155–171.
- Mundy, P., Block, J., Delgado, C., Pomares, Y., Van Hecke, A. V., and Parlade, M. V. (2007). Individual differences and the development of joint attention in infancy. *Child development*, 78:938–954.
- Quine, W. V. O. (1960). *Word and object*. Cambridge University Press.
- Siskind, J. M. (1996). A computational study of cross-situational techniques for learning word-to-meaning mappings. *Cognition*, 61:39–91.
- Smith, K., Smith, A., Blythe, R., and Vogt, P. (2006). Cross-situational learning: a mathematical approach. In Vogt, P., Sugita, Y., Tuci, E., and Nehaniv, C., (Eds.), *Symbol grounding and beyond*. Springer.
- Steels, L., Kaplan, F., McIntyre, A., and Van Looven, J. (2002). Crucial factors in the origins of word-meaning. In Wray, A., (Ed.), *The Transition to Language*, Oxford, UK. Oxford University Press.
- Vogt, P. and de Boer, B. (2009). Language evolution: Computer models for empirical data. *Adaptive Behavior*. to appear.