

**Understanding the Embodied Teacher:
Nonverbal Cues for Sociable Robot Learning**

by

Matthew Roberts Berlin

S.M., Media Arts and Sciences, Massachusetts Institute of Technology, 2003
A.B., Computer Science, Harvard College, 2001

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

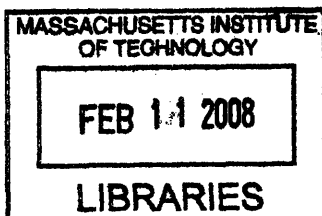
February 2008

© Massachusetts Institute of Technology 2008. All rights reserved.

Author Matthew Roberts Berlin
Program in Media Arts and Sciences
January 11, 2008

Certified by Cynthia Breazeal
Associate Professor of Media Arts and Sciences
Program in Media Arts and Sciences
Thesis Supervisor

Accepted by Deb Roy
Chair, Departmental Committee on Graduate Students
Program in Media Arts and Sciences



ARCHIVES

**Understanding the Embodied Teacher:
Nonverbal Cues for Sociable Robot Learning**

by

Matthew Roberts Berlin

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
on January 11, 2008, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Media Arts and Sciences

Abstract

As robots enter the social environments of our workplaces and homes, it will be important for them to be able to learn from natural human teaching behavior. My research seeks to identify simple, non-verbal cues that human teachers naturally provide that are useful for directing the attention of robot learners. I conducted two novel studies that examined the use of embodied cues in human task learning and teaching behavior. These studies motivated the creation of a novel data-gathering system for capturing teaching and learning interactions at very high spatial and temporal resolutions. Through the studies, I observed a number of salient attention-direction cues, the most promising of which were visual perspective, action timing, and spatial scaffolding. In particular, this thesis argues that spatial scaffolding, in which teachers use their bodies to spatially structure the learning environment to direct the attention of the learner, is a highly valuable cue for robotic learning systems. I constructed a number of learning algorithms to evaluate the utility of the identified cues. I situated these learning algorithms within a large architecture for robot cognition, augmented with novel mechanisms for social attention and visual perspective taking. Finally, I evaluated the performance of these learning algorithms in comparison to human learning data, providing quantitative evidence for the utility of the identified cues. As a secondary contribution, this evaluation process supported the construction of a number of demonstrations of the humanoid robot Leonardo learning in novel ways from natural human teaching behavior.

Thesis Supervisor: Cynthia Breazeal

Title: Associate Professor of Media Arts and Sciences, Program in Media Arts and Sciences

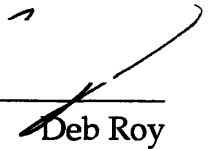
**Understanding the Embodied Teacher:
Nonverbal Cues for Sociable Robot Learning**

by

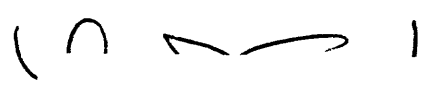
Matthew Roberts Berlin

The following people served as readers for this thesis:

Thesis Reader _____


Deb Roy
Associate Professor of Media Arts and Sciences
Program in Media Arts and Sciences

Thesis Reader _____


Linda Smith
Professor of Psychology
Indiana University

Acknowledgments

I owe thanks to many, many people, without whom this thesis would not have been possible. First and foremost, I want to thank my committee - Cynthia Breazeal, Deb Roy, and Linda Smith - for their patience with me and for the steady, illuminating guidance that they gave to this work. I am particularly grateful to Cynthia, who has given me a fantastic home here at the Media Lab for the past three-and-a-half years.

I have had the great fortune to be a member of two extraordinary research groups during my time at MIT. It was a true pleasure working with the Synthetic Characters: Bruce Blumberg, Daphna Buchsbaum, Rob Burke, Jennie Cochran, Marc Downie, Scott Eaton, Matt Grimes, Damian Isla, Yuri Ivanov, Mike Johnson, Aileen Kawabe, Derek Lyons, Ben Resner, and Bill Tomlinson (plus Ari Benbasat and Josh Lifton). I am forever indebted to Bruce, Bill, and Marc, who were my first mentors at the lab and who got me started down this crazy path.

Likewise, it has been a tremendous pleasure working with the Personal Robots: Sigurður Örn Adalgeirsson, Polly Guggenheim, Matt Hancher, Cory Kidd, Heather Knight, Jun Ki Lee, Jeff Lieberman, John McBean, Philipp Robbel, Mikey Siegel, Dan Stiehl, and Rob Tuscano. In particular, I want to send an emphatic shout-out to my co-conspirators on the Leo software team: Andrew 'Zoz' Brooks, Jesse Gray, Guy Hoffman, and Andrea Thomaz.

To the Characters and the Robots: thank you so much for riding into battle with me time and time again, and for sharing your ferocious creativity with me.

Some special thanks are due to a few people who provided crucial assistance in the final stages of this work. Jeff Lieberman helped with the design and assembly of the motorized puzzle boxes. Mikey Siegel and Aileen Kawabe assisted with the construction of the illuminated table. And my hotshot UROP Crystal Chao contributed some essential pieces of software down the stretch.

I am, as always, particularly indebted to Jesse Gray, who has been my great friend and collaborator for many years now. This thesis would have been much less interesting without his help, and it certainly wouldn't have been nearly as much fun to put together.

This thesis is dedicated to my family. Mom, Dad, Alex - I love you all so much.

Contents

Abstract	3
1 Introduction	15
1.1 Human Studies and Benchmark Tasks	17
1.2 Robotic Learning Algorithms and Cognitive Architecture	18
1.3 Interactive Learning Demonstrations	19
2 Embodied Emphasis Cues	21
2.1 Background and Related Literature	22
2.2 Perspective Taking Study	24
2.2.1 Task Design and Protocol	24
2.2.2 Results	26
2.3 Emphasis Cues Study	27
2.3.1 Task Design and Protocol	29
2.3.2 Data-Gathering Overview	33
2.3.3 Motion Capture: Tracking Heads and Hands	35
2.3.4 The Light Table: Tracking Object Movement	38
2.3.5 The Puzzle Boxes: Tracking Object Manipulation	45
2.3.6 Video Recording	46
2.3.7 Data Stream Management and Synchronization	47
2.3.8 Study Execution and Discussion	48
2.3.9 Data Analysis Tools and Pipeline	53
2.4 Summary: Cues That Robotic Learners Need to Understand	60
3 Learning Algorithms and Architecture	61
3.1 Self-as-Simulator Cognitive Architecture	62
3.2 Perspective Taking Mechanisms	65
3.2.1 Belief Modeling	65
3.2.2 Perspective Taking and Belief Inference	68
3.3 Social Attention Mechanisms	71
3.4 Unified Social Activity Framework	73
3.4.1 Action Representation: Process Models	74
3.4.2 Exploration Representation: Interpretation Modes	77
3.5 Learning Algorithms	79
3.5.1 Task and Goal Learning	80

3.5.2	Perspective Taking and Task Learning	82
3.5.3	Constraint Learning from Embodied Cues	82
4	Evaluations and Demonstrations	85
4.1	Visual Perspective Taking Demonstration and Benchmarks	85
4.2	Emphasis Cues Benchmarks	88
4.3	Emphasis Cues Demonstration	93
5	Conclusion	99
5.1	Contributions	99
5.2	Future Work	100
5.2.1	Additional Embodied Emphasis Cues	100
5.2.2	Cues for Regulating the Learning Interaction	101
5.2.3	Robots as Teachers	101
5.2.4	Dexterity and Mobility	102
	Bibliography	103

List of Figures

2-1	The four tasks demonstrated to participants in the study.	24
2-2	Input domains consistent with the perspective taking (PT) vs. non-perspective taking (NPT) hypotheses.	25
2-3	Task instruction cards given to learners in Task 2.	31
2-4	One of the motorized puzzle boxes.	32
2-5	Graphical visualization within the object tracking toolkit.	36
2-6	Hats, gloves, and rings outfitted with trackable markers.	37
2-7	Initial assembly of the light table, with detail of the frosted acrylic top. . . .	38
2-8	Light table component details.	39
2-9	Final setup of the light table.	41
2-10	Color segmentation process.	42
2-11	The block shape recognition system.	43
2-12	The block tracking system.	44
2-13	Puzzle boxes in the study environment, on top of the light table.	45
2-14	Two camcorders were mounted around the study area.	46
2-15	Data visualization environment.	54
2-16	Blocks were mapped into the coordinate frame of the motion capture system. .	55
2-17	Change in distance to the body of the learner for block movements initiated by the teacher.	57
2-18	Predictive power of movements towards and away from the body of the learner.	58
2-19	Predictive power of teacher's movements following learner's movements. . .	59
3-1	The Leonardo robot and graphical simulator	62
3-2	System architecture overview.	63
3-3	Architecture for modeling the human's beliefs.	69
3-4	Timeline following the progress of the robot's beliefs for one button.	70
3-5	Saliency of objects is computed from several environmental and social factors. .	71
3-6	Leo in his workspace with a human partner.	72
3-7	Action selection algorithm for process models.	76
4-1	Leo was presented with similar learning tasks in a simulated environment. .	86
4-2	The robot was presented with the recorded study data.	89
4-3	The robot learned live by interacting with a human teacher.	93
4-4	The gestural interface and figure planning algorithm.	95

4-5	Interaction sequence between the robot and a human teacher.	96
-----	---	----

List of Tables

2.1	Differential rule acquisition for study participants in social vs. nonsocial conditions.	27
2.2	Embodied cues of positive emphasis.	50
2.3	Embodied cues of negative emphasis.	51
4.1	High-likelihood hypotheses entertained by the robot at the conclusion of benchmark task demonstrations.	87
4.2	Hypotheses selected by study participants following task demonstrations. .	87
4.3	Performance of the human learners on study tasks 2a and 2b.	91
4.4	Learning performance of the robot observing benchmark task interactions. .	92

Chapter 1

Introduction

How can we design robots that are competent, sensible learners? Learning will be an important part of bringing robots into the social, cooperative environments of our workplaces and homes. But social environments present a host of novel challenges for learning machines. Learning from real people means no carefully labeled data sets, no clear-cut reward signals, and no obvious indications of when to start learning and when to stop. Social environments are typically cluttered and dynamic, with other agents altering the world in potentially confusing or unpredictable ways. People working alongside of robots may be largely unfamiliar with robotic technology and machine learning and, to make matters worse, will expect a robot to adapt and learn as quickly and “effortlessly” as a real human teammate.

To address these issues, and inspired by the way people and animals learn from others, researchers have begun to investigate various forms of social learning and interactive training techniques, such as imitation-based learning [Schaal, 1999], clicker training [Blumberg et al., 2002], learning by demonstration [Nicolescu and Matarić, 2003], and tutelage [Breazeal et al., 2004].

Most existing approaches to socially guided learning take an all-or-nothing approach to interpreting the human’s behavior. The human’s behavior or directives are either du-

plicated exactly, as in imitation or verbal instruction, or else used simply as feedback indicating the success or failure of the robot’s actions, as in reinforcement learning and clicker training. I seek a more flexible approach spanning this range of interpretation. This thesis advances a model wherein the human’s behavior is viewed as a communicative, dynamic constraint upon the robot’s exploration of its environment.

This thesis is focused on the question of how the embodied presence of the teacher directs and constrains the learner’s attention and bodily exploration. How does the body pose and activity of the teacher help to identify what matters in the interaction? By designing a robotic system with the right internal representations and processes, coupled in the right ways to the complexity of the human system, I aim to enable a human teacher to effectively guide the robot.

My research seeks to identify simple, non-verbal cues that human teachers naturally provide that are useful for directing the attention of robot learners. The structure of social behavior and interaction engenders what I term “social filters:” dynamic, embodied cues through which the teacher can guide the behavior of the robot by emphasizing and de-emphasizing objects in the environment.

This thesis describes two novel studies that I conducted to examine the use of social filters in human task learning and teaching behavior. Through these studies, I observed a number of salient attention-direction cues, the most promising of which were visual perspective, action timing, and spatial scaffolding. In particular, I argue that spatial scaffolding, in which teachers use their bodies to spatially structure the learning environment to direct the attention of the learner, is a highly valuable cue for robotic learning systems.

In order to directly evaluate the utility of the identified cues, I constructed a number of learning algorithms. I situated these learning algorithms within a large architecture for robot cognition, augmented with novel mechanisms for social attention and visual perspective taking. I evaluated the performance of these learning algorithms in comparison to human learning data on benchmark tasks drawn from the studies, providing quantitative evidence for the utility of the identified cues. As a secondary contribution, this evaluation

process supported the construction of a number of demonstrations of the humanoid robot Leonardo learning in novel ways from natural human teaching behavior.

1.1 Human Studies and Benchmark Tasks

I conducted two studies that examined the use of embodied cues in human task learning and teaching behavior. The studies focused on embodied, non-verbal cues through which human teachers emphasize and de-emphasize objects in the learning environment. The first study examined the role of visual perspective taking in human learning. The second study was more open-ended, and was designed to capture observations of a number of dynamic, embodied cues including visual attention, hand gestures, direct object manipulations, and spatial/environmental scaffolding. This study motivated the creation of a novel data-gathering system for capturing teaching and learning interactions at very high spatial and temporal resolutions.

While a broad set of attention-direction cues were observed in the second study, a surprising result was the prevalent and consistent use of spatial scaffolding by the human teachers. The term spatial scaffolding refers to the ways in which teachers use their bodies to spatially structure the learning environment to direct the attention of the learner. One of the main contributions of my work is the empirical demonstration of the utility of spatial scaffolding for robotic learning systems. In particular, this thesis focuses on a simple, reliable, component of spatial scaffolding: attention direction through object movements towards and away from the body of the learner.

Both of the human studies involved learning tasks that were designed to be closely matched to the Leonardo robot’s existing perceptual and inferential capabilities. This served two purposes. First, it meant that the recorded observations would be directly applicable to the robot’s cognitive architecture. Second, it allowed for the creation of a benchmark suite, whereby the robot’s performance on the benchmark learning tasks could be directly compared to human learning performance on similar or identical tasks. This

comparison provided direct, quantitative evidence for the utility of the attention-direction cues identified through the studies.

1.2 Robotic Learning Algorithms and Cognitive Architecture

In order to directly evaluate the utility of the cues identified in the studies, including visual perspective, action timing, and spatial scaffolding, I constructed a number of learning algorithms. I situated these learning algorithms within a large architecture for robot cognition, augmented with novel mechanisms for social attention and visual perspective taking. Designing an integrated learning system in this way supported not only the direct evaluation of the robot's performance on the study benchmark tasks, but also a number of demonstrations of interactive social learning.

I believe that socially situated robots will need to be designed as socially cognitive learners that can infer the intention behind human instruction, even if the teacher's demonstrations are insufficient or ambiguous from a strict machine learning perspective. My approach to endowing machines with socially-cognitive learning abilities is inspired by leading psychological theories and recent neuroscientific evidence for how human brains might infer the mental states of others. Specifically, *Simulation Theory* holds that certain parts of the brain have dual use; they are used to not only generate behavior and mental states, but also to predict and infer the same in others [Davies and Stone, 1995, Barsalou et al., 2003, Sebanz et al., 2006].

My research introduces a set of novel software technologies that build upon a large architecture for robotic behavior generation and control developed by the Personal Robots Group and based on [Blumberg et al., 2002]. My research extends the mechanisms of attention, belief, goals, and action selection within this architecture. Taken together, these changes result in an integrated architecture we call the "self-as-simulator" behavior system, wherein the robot's cognitive functionality is organized around an assumption of shared social embodiment - the assumption that the teacher has a body and mind "like

mine.” This design allows the robot to simulate the cognitive processes of a human interaction partner using its own generative mechanisms. The architecture was designed for and evaluated on the 65 degree of freedom humanoid robot Leonardo and its graphical simulator.

This thesis introduces and describes the technologies behind the self-as-simulator behavior system. These include: mechanisms for understanding the environment from the visual perspective of the teacher, social mechanisms of attention direction and emphasis, and a unified framework for social action recognition and behavior generation. The details of these mechanisms are presented in chapter 3.

1.3 Interactive Learning Demonstrations

In addition to comparing the learning performance of our cognitive architecture against that of human learners on benchmark tasks drawn from the studies, this thesis presents a pair of demonstrations of the Leonardo robot making use of embodied cues to learn in novel ways from natural human teaching behavior. In the first demonstration, the Leonardo robot takes advantage of perspective taking to learn from ambiguous task demonstrations involving colorful foam blocks. The second demonstration features Leo making use of action timing and spatial scaffolding to learn secret constraints associated with a number of construction tasks, again involving foam blocks. Leonardo is the first robot to make use of visual perspective, action timing, and spatial scaffolding to learn from human teachers.

This document proceeds as follows. Chapter 2 describes the two studies of human teaching and learning behavior, and the novel data-gathering and analysis system that was designed to support this work. I discuss the embodied attention-direction cues observed through the studies, and argue that visual perspective, action timing, and spatial scaffolding are particularly promising cues for robot learners. Chapter 3 introduces the learning algorithms that were developed to empirically evaluate these embodied cues. I

also provide a detailed description of the robot's cognitive architecture and its novel mechanisms of social attention and visual perspective taking. Chapter 4 presents the robot's learning performance on the study benchmark tasks, providing quantitative evidence for the utility of the identified cues. In addition, I describe the interactive demonstrations of the Leonardo robot making use of dynamic, embodied cues to learn from natural teaching behavior. Finally, chapter 5 provides some concluding thoughts and discusses some plans and possibilities for future research.

Chapter 2

Embodied Emphasis Cues

In this chapter, I describe two studies that I conducted to examine the use of embodied cues in human task learning and teaching behavior. The studies focused on embodied, non-verbal cues through which human teachers emphasize and de-emphasize objects in the learning environment. The first study examined the role of visual perspective taking in human learning. The second study was more open-ended, and was designed to capture observations of a number of dynamic, embodied cues including visual attention, hand gestures, direct object manipulations, and spatial/environmental scaffolding. This study motivated the creation of a novel data-gathering system for capturing teaching and learning interactions at very high spatial and temporal resolutions.

While a broad set of attention-direction cues were observed in the second study, a surprising result was the prevalent and consistent use of spatial scaffolding by the human teachers. In this chapter, I present quantitative results highlighting a simple, reliable, spatial cue: attention direction through object movements towards and away from the body of the learner.

Both studies involved learning tasks that were designed to be closely matched to the Leonardo robot's existing perceptual and inferential capabilities. This served two purposes. First, it meant that the recorded observations could be used to inform and revise the

robot’s cognitive architecture. Second, it allowed me to create a benchmark suite so that the robot’s performance on the benchmark learning tasks could be directly compared to human learning performance on similar or identical tasks. This comparison, described in the following chapters, provided direct, quantitative evidence for the utility of the promising attention-direction cues identified through the studies: visual perspective, action timing, and spatial scaffolding.

2.1 Background and Related Literature

There has been a large, interesting body of work focusing on human gesture, especially communicative gestures closely related to speech [Cassell, 2000, Kendon, 1997, McNeill, 1992]. A number of gestural classification systems have been proposed [Kipp, 2004, McNeill, 2005, Nehaniv et al., 2005]. Others have focused on the role of gesture in language acquisition [Iverson and Goldin-Meadow, 2005], and on the use of gesture in teaching interactions between caregivers and children [Zukow-Goldring, 2004, Singer and Goldin-Meadow, 2005].

In the computer vision community, there has been significant prior work on technical methods for tracking head pose [Morency et al., 2002] and for recognizing hand gestures such as pointing [Wilson and Bobick, 1999, Kahn et al., 1996]. Others have contributed work on using these cues as inputs to multi-modal interfaces [Bolt, 1980, Oviatt et al., 1997]. Such interfaces often specify fixed sets of gestures for controlling systems such as graphical expert systems [Kobsa et al., 1986], natural language systems [Neal et al., 1998], and even directable robotic assistants [Ghidary et al., 2002, Severinson-Eklundh et al., 2003, Fransen et al., 2007].

However, despite a large body of work on understanding eye gaze [Perrett and Emery, 1994, Langton, 2000], much less work has been done on using other embodied cues to infer a human’s emphasis and de-emphasis in behaviorally realistic scenarios. It is important to stress that tracking the human’s head pose, which is a directly observable feature, is quite a different thing from extrapolating from this feature and other related features to

infer the human's object of emphasis, which we might think of as an unobservable, mental state. There has been a small amount of related work on mapping from head pose to a specific attentional focus, such as a driver's point of interest within a car [Pappu and Beardsley, 1998] or the listener that a particular speaker is addressing in a meeting scenario [Stiefelhagen, 2002].

One of the important contributions of my work is the analysis of spatial scaffolding cues in a human teaching and learning interaction, and the empirical demonstration of the utility of spatial scaffolding for robotic learning systems. Spatial scaffolding refers to the ways in which teachers use their bodies to spatially structure the learning environment to direct the attention of the learner. In particular, my work identifies a simple, reliable, component of spatial scaffolding: attention direction through object movements towards and away from the body of the learner. It is important to note that spatial scaffolding may be effective through social mechanisms or through nonsocial mechanisms implicit in the layout of the learning space. The learner may be directly interpreting the bodily gestures of the teacher, or it may be that the teacher is simply creating organization for the spatial direction and extent of the learner's attention. Space itself, as mediated by the bodily orientation of attention, has been shown to be a powerful factor in conceptual linkage and word learning in children [Smith et al., 2007]. It may well be the case that social and nonsocial mechanisms play a combined role in learning through spatial scaffolding.

The first study presented in this chapter examined the role of visual perspective taking in human task learning. A number of related studies have examined aspects of human perspective taking, most notably in false-belief reasoning [Wimmer and Perner, 1983]. Researchers have examined the role of visual perspective taking in language-mediated collaboration between people working face-to-face [Keysar et al., 2000, Hanna et al., 2003] as well as in distributed teams [Jones and Hinds, 2002]. Others have studied the role of spatial perspective taking in demonstrations of physical assembly tasks [Martin et al., 2005]. The study described in the following section is unique in its focus on strictly non-verbal interaction and on the potentially critical role of the teacher's visual perspective in disambiguating task demonstrations.

2.2 Perspective Taking Study

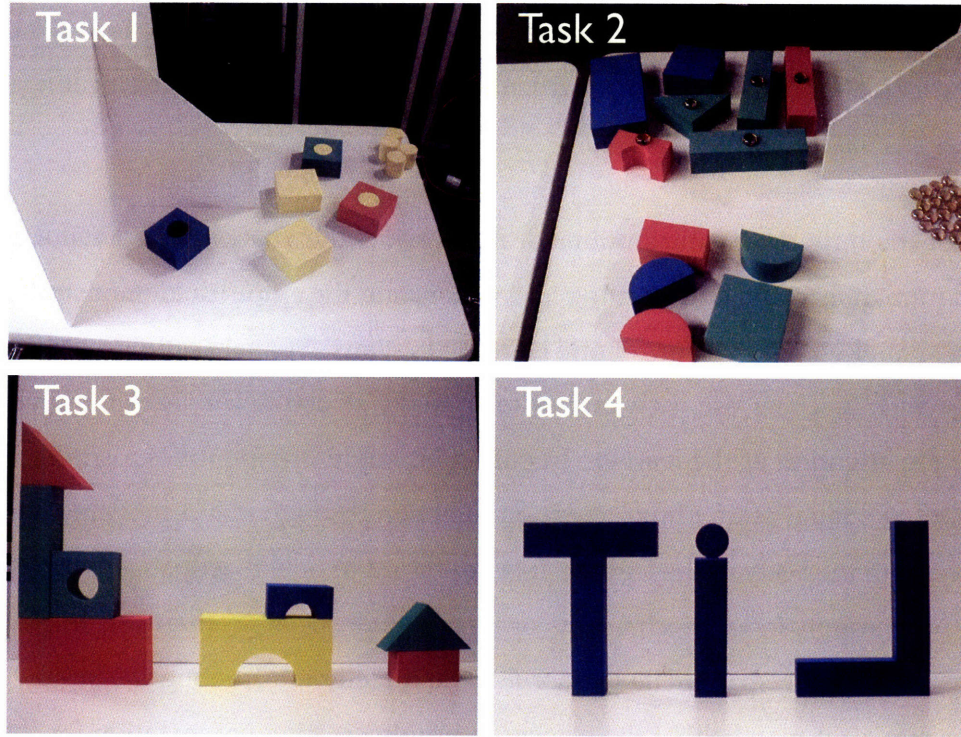


Figure 2-1: The four tasks demonstrated to participants in the study (photos taken from the participant's perspective). Tasks 1 and 2 were demonstrated twice with blocks in different configurations. Tasks 3 and 4 were demonstrated only once.

The first study examined the role of visual perspective taking in human learning, and enabled the creation of benchmark tasks with which to evaluate the robot's perspective taking abilities and analyze the utility of visual perspective as an information channel for automated learning systems.

2.2.1 Task Design and Protocol

Study participants were asked to engage in four different learning tasks involving foam building blocks. I gathered data from 41 participants, divided into two groups. 20 participants observed demonstrations provided by a human teacher sitting opposite them (the social condition), while 21 participants were shown static images of the same demonstrations, with the teacher absent from the scene (the nonsocial condition). Participants were

asked to show their understanding of the presented skill either by re-performing the skill on a novel set of blocks (in the social context) or by selecting the best matching image from a set of possible images (in the nonsocial context).

Figure 2-1 illustrates sample demonstrations of each of the four tasks. The tasks were designed to be highly ambiguous, providing the opportunity to investigate how different types of perspective taking might be used to resolve these ambiguities. The subjects' demonstrated rules can be divided into three categories: perspective taking (PT) rules, non-perspective taking (NPT) rules, and rules that did not clearly support either hypothesis (Other).

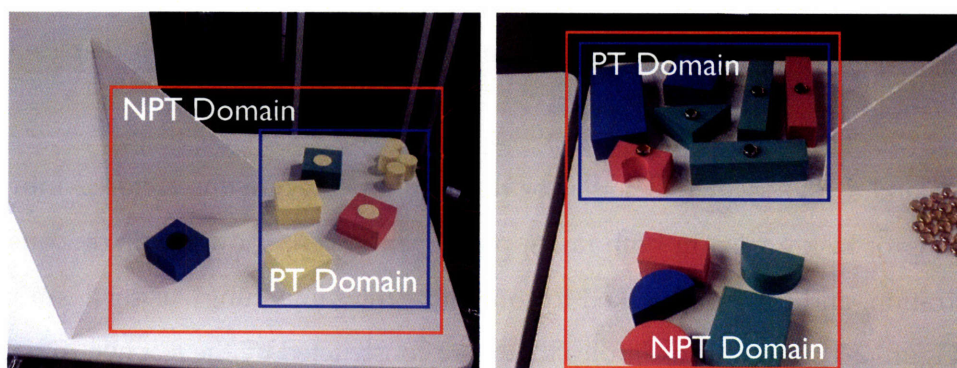


Figure 2-2: Input domains consistent with the perspective taking (PT) vs. non-perspective taking (NPT) hypotheses. In visual perspective taking (left image), the student's attention is focused on just the blocks that the teacher can see, excluding the occluded block. In resource perspective taking (right image), attention is focused on just the blocks that are considered to be "the teacher's," excluding the other blocks.

Task 1 focused on visual perspective taking during the demonstration. Participants were shown two demonstrations with blocks in different configurations. In both demonstrations, the teacher attempted to fill all of the holes in the square blocks with the available pegs. Critically, in both demonstrations, a blue block lay within clear view of the participant but was occluded from the view of the teacher by a barrier. The hole of this blue block was never filled by the teacher. Thus, an appropriate (NPT) rule might be "fill all but blue," or "fill all but this one," but if the teacher's perspective is taken into account, a more parsimonious (PT) rule might be "fill all of the holes" (see Fig. 2-2).

Task 2 focused on resource perspective taking during the demonstration. Again, par-

ticipants were shown two demonstrations with blocks in different configurations. Various manipulations were performed to encourage the idea that some of the blocks “belonged” to the teacher, whereas the others “belonged” to the participant, including spatial separation in the arrangement of the two sets of blocks. In both demonstrations, the teacher placed markers on only “his” red and green blocks, ignoring his blue blocks and all of the participant’s blocks. Because of the way that the blocks were arranged, however, the teacher’s markers were only ever placed on triangular blocks, long, skinny, rectangular blocks, and bridge-shaped blocks, and marked all such blocks in the workspace. Thus, if the blocks’ “ownership” is taken into account, a simple (PT) rule might be “mark only red and green blocks,” but a more complicated (NPT) rule involving shape preference could account for the marking and non-marking of all of the blocks in the workspace (see Fig. 2-2).

Task 3 and 4 investigated whether or not visual perspective is factored into the understanding of task goals. In both tasks, participants were shown a single construction demonstration, and then were asked to construct “the same thing” using a similar set of blocks. Figure 2-1 shows the examples that were constructed by the teacher. In both tasks, the teacher assembled the examples from left to right. In task 4, the teacher assembled the word “LiT” so that it read correctly from their own perspective. The question was, would the participants rotate the demonstration (the PT rule) so that it read correctly for themselves, or would they mirror the figure (the NPT rule) so that it looked exactly the same as the demonstration (and thus read backwards from their perspective). Task 3, in which the teacher assembled a sequence of building-like forms, was essentially included as a control, to see if people would perform any such perspective flipping in a non-linguistic scenario.

2.2.2 Results

The results of the study are summarized in Table 2.1 where participant behavior was recorded and classified according to the exhibited rule. For every task, differences in rule choice between the social and nonsocial conditions were highly significant (chi-square, $p < 0.001$). The most popular rule for each condition is highlighted in bold (note that,

Table 2.1: Differential rule acquisition for study participants in social vs. nonsocial conditions. ***: $p < 0.001$

Task	Condition	PT Rule	NPT Rule	Other	p
Task 1	social	6	1	13	***
	nonsocial	1	12	8	
Task 2	social	16	0	4	***
	nonsocial	7	12	2	
Task 3	social	12	8	-	***
	nonsocial	0	21	-	
Task 4	social	14	6	-	***
	nonsocial	0	21	-	

while many participants fell into the “Other” category for Task 1, there was very little rule agreement between these participants). These results strongly support the intuition that perspective taking plays an important role in human learning in socially situated contexts.

However, a critical question remains: can a robot, using a simple learning algorithm and paying attention to the visual perspective of the teacher, exhibit the differences in rule choice observed in human learners? Is visual perspective a sufficiently informative cue to support automated learning from natural teaching behavior? In the next chapter, I describe a learning algorithm that I constructed to answer this question, along with the visual perspective taking mechanisms that I incorporated into the Leonardo robot’s cognitive architecture. In chapter 4, I present the learning performance of the robot on benchmark tasks drawn from this study, providing evidence of the utility of visual perspective as an information channel for automated learning systems.

2.3 Emphasis Cues Study

The first study examined how teachers implicitly emphasize and de-emphasize objects through their limited visual perspectives. In this section, I describe a second study that was more open-ended. This study was designed to capture a range of dynamic, embodied cues through which emphasis and de-emphasis are communicated by human teachers.

These cues included visual attention, hand gestures, direct object manipulations, and spatial/environmental scaffolding.

The study employed a novel sensory apparatus to capture embodied teaching and learning behavior with very high spatial and temporal resolution. This apparatus combined machine vision technologies including motion capture and object tracking with other technical interventions such as instrumented mechanical objects with which study participants interacted.

The study had a number of goals. First, to produce a high-resolution observational data set that would be interesting to myself and to other researchers in its own right. Second, to provide data sufficient for identifying and implementing heuristics for tracking a number of important dynamic emphasis and de-emphasis cues that teachers provide in a realistic teaching domain. Finally, to produce a set of benchmark tasks with which to automatically evaluate the identified cues - by demonstrating an automated learning system, attending only to a handful of simple cues, learning “alongside of” the real human learners.

This section proceeds as follows. I first introduce the teaching/learning tasks that study participants were asked to engage in. I then describe the design and construction of the sensory apparatus and data collection tools that were created to collect high-quality observations of the study tasks. Next, I describe the execution of the study itself, and discuss some qualitative observations of the participants’ teaching and learning behavior. Finally, I describe the automated data analysis tools that were created to detect and measure the teachers’ emphasis cues, and present the quantitative results generated by these tools.

While a broad set of attention-direction cues were observed in this study, a surprising result was the prevalent and consistent use of spatial scaffolding by the human teachers. In particular, the quantitative results highlight a simple, reliable, spatial cue: attention direction through object movements towards and away from the body of the learner.

2.3.1 Task Design and Protocol

A set of tasks was designed to examine how teachers emphasize and de-emphasize objects in a learning environment with their bodies, and how this emphasis and de-emphasis guides the exploration of a learner and ultimately the learning that occurs.

The study centered around three types of tasks. The first two types of tasks involved colorful foam building blocks similar to the blocks used in the perspective taking study. The third type of task involved motorized “puzzle” boxes, shown in figure 2-4. Each study session included two of the first type of task, three of the second type, and three of the third type, for a total of eight tasks per study session.

I gathered data from 72 individual participants, combined into 36 pairs. Each study session lasted approximately 45 minutes.

Participants were combined into pairs, and were asked to engage in all eight of the study tasks. For each pair, one participant was randomly assigned to play the role of teacher and the other participant assigned the role of learner for the duration of the study. For all of the tasks, participants were asked not to talk, but were told that they could communicate in any way that they wanted other than speech. Tasks were presented in a randomized order. For all of the tasks, the teacher and learner stood on opposite sides of a tall table, with either the foam blocks or the puzzle boxes laid out between them on the tabletop. Participants were given general verbal instructions for each type of task by the experimenter, and then were handed printed instructions on note cards for each specific task.

The first type of task was a non-interactive, demonstration task, designed to examine how teachers draw attention to a subset of objects in the environment. The teacher was asked to construct three successive demonstrations of a particular rule, that involved placing markers on some of the foam blocks and not placing markers on others of the blocks. The first rule (Task 1a) was to “put markers on all of the red and green blocks that aren’t triangles.” The second rule (Task 1b) was to “put markers on all the rectangular (and square)

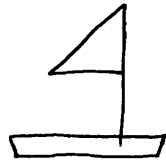
blocks that aren't red." After each of the teacher's three successive demonstrations, the learner was asked to write down their current best guess as to what the rule might be.

To make the teacher's job more difficult, the teacher was only given six markers, which was not enough to successfully demonstrate the rule over all of the blocks on the table: the rule in Task 1a matched ten of the blocks on the table, while the rule in Task 1b matched nine. Thus, one of the interesting questions for this task was how would the teacher overcome this resource limitation to successfully communicate the rule. How would teachers differentiate demonstration blocks from distractor blocks (blocks that matched the rule but that necessarily would go unmarked)? Other questions of interest at the outset were how the teachers would arrange their demonstrations spatially and how they would transition from one demonstration in the sequence to the next demonstration.

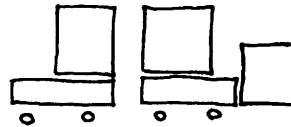
The same 24 foam blocks were used for all of the tasks involving blocks. These 24 blocks were made up of four different colors - red, green, blue, and yellow, with six different shapes in each color - triangle, square, small circle, short rectangle, long rectangle, and a large, arch-shaped block. At the beginning of each task, the experimenter arranged the blocks in a default, initial configuration, shown in figure 2-9. This initial configuration was designed to seem random, and featured a fairly even distribution of the different colors and shapes into the four quadrants of the tabletop. The blocks were reset into this initial configuration between every task, with the exception of Tasks 1a and 1b, where the teachers were allowed to continue on from their demonstrations of the first rule into their demonstrations of the second rule.

The second type of task also involved teaching with foam blocks, but in a more interactive setting. The tasks were secret constraint tasks, where one person (the learner) knows what the task is but does not know the secret constraint. The other person (the teacher) doesn't know what the task is but does know the constraint. So, both people must work together to successfully complete the task. For each of the tasks, the learner received instructions, shown in figure 2-3, for a figure to construct using the blocks. In Task 2a, the learner was instructed to construct a sailboat figure using at least 7 blocks; in Task 2b, a truck/train figure using at least 8 blocks; and in Task 2c, a smiley face figure using at least

Construct using at least 7 blocks:



Construct using at least 8 blocks:



Construct using at least 8 blocks:



Figure 2-3: Task instruction cards given to learners in Task 2.

6 blocks. The block number requirements were intended to prevent minimalist interpretations of the figures (and thus very quick solutions to the tasks). When put together with the secret constraints, the number requirements turned tasks 2a and 2b into modestly difficult Tangram-style spatial puzzles.

The secret constraint handed to the teacher for Task 2a was that “the figure must be constructed using only blue and red blocks, and no other blocks.” The secret constraint for Task 2b was that “the figure must include all of the triangular blocks, and none of the square blocks,” and for Task 2c, that “the figure must be constructed only on the left half of the table (from your perspective).” At the end of each task, the learner was asked to write down what they thought the secret constraint might have been.

These tasks were designed to examine how the teacher would guide the actions of the learner. Would they interrupt only when the learner made a mistake, or would they provide guidance preemptively? Would they remove mistakenly placed blocks or possibly provide correct replacements? Would they direct the learner towards specific blocks gesturally (by pointing or tapping), or by handing them specific blocks to use, or might they provide assistance by organizing the blocks spatially for the learner? The interactive setting of these tasks provokes a rich range of questions for analysis, which I will return to

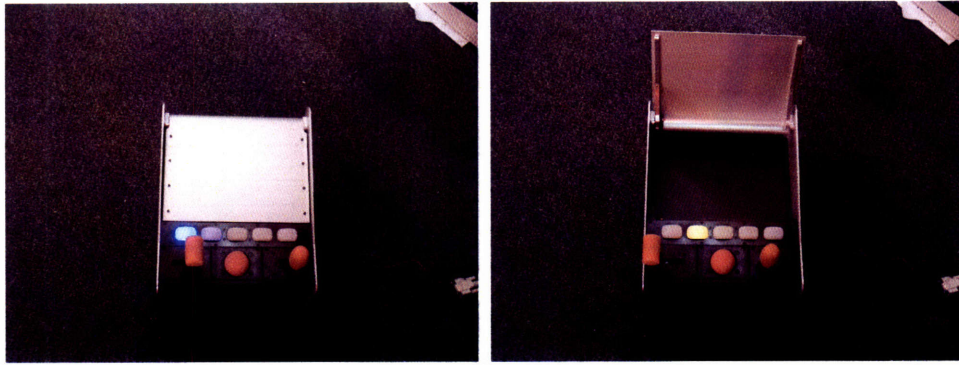


Figure 2-4: One of the motorized puzzle boxes. Left: box closed with blue status light illuminated. Right: box open with orange status light illuminated.

later in the section.

The third type of task followed a similar secret constraint task setup, but applied to the domain of sequence learning. Participants interacted with a pair of mechanical puzzle boxes, one with blue controls and one with red controls (as in figure 2-4). The puzzle boxes were designed as follows. Each box has three squishy, silicone controls: a button, a left-to-right slider, and a left-to-right switch. The controls were designed to be easy to manipulate for the Leonardo robot as well as for a human. In addition to the controls, each box has five colored status lights, and a motorized lid that opens and closes based on changes in box state. Additionally, changes in box state can cause songs and other noises to play through a nearby speaker. Since the boxes are controlled by a computer, the experimenter can design arbitrary mappings between control manipulations, status light changes, box openings and closings, and auditory events (a similar experimental setup, used for studying causal learning in children, is described in [Gopnik et al., 2001] and [Schulz and Gopnik, 2004]).

In the study setup, both boxes were arranged on the tabletop so that the controls faced the learner (see figure 2-13). For these tasks, the learner was instructed that their job was to discover the specific sequence of manipulations of the box controls that would cause a song to play (in this case, a highly enthusiastic, game show-style song). The teacher was instructed to help the learner accomplish this task using the secret hints that they would be provided with - additional information that they could use to guide the actions of the learner.

For Task 3a, the hint provided partial sequence information: “When the orange light goes on: first the red slider and then the red switch must be flipped. - When both lights turn green: the slider on the blue box must be flipped.” For Task 3b, the hint provided some information about the box controls: “The blue slider and the blue button and the red switch are all bad. - The other controls are good.” Finally, for Task 3c, the hint provided information about the status lights: “The lights on the blue box are distractors. The lights on the red box are helpful.”

For these box puzzle tasks, I was looking for: how the teacher might direct the learner towards particular controls gesturally or through direct manipulation, how they might direct the learner’s attention towards important state changes, how they might guide the learner away from unhelpful controls and states, and so on.

Now that I have described the tasks that study participants were asked to perform, I move on to a discussion of the sensory apparatus and data-gathering tools that were created to record their behavior.

2.3.2 Data-Gathering Overview

In order to record high-resolution data about the study interactions, I developed a data-gathering system which incorporated multiple, synchronized streams of information about the study participants and their environment. For all of the tasks, I tracked the positions and orientations of the heads and hands of both participants, recorded video of both participants, and tracked all of the objects with which the participants interacted. For Tasks 1 and 2, I tracked the positions and orientations of all of the foam blocks. For Task 3, I recorded the states of the lights and controls of both puzzle boxes.

The data gathering system was designed to satisfy a number of goals. First, it was important to minimize the burden that the sensory system might impose on study participants, and thus the effect that it might have on the naturalness of their behavior. Second, it was important for the recording system to be able to be controlled by a single experimenter. Finally, it was important for the recording system to automatically digitize and

synchronize all of the sensory streams on the fly, to minimize any time-consuming manual processing of the data that might be required after the conclusion of the study. Synchronization of the various sensory streams enabled the simultaneous playback of all of the recorded data for the purposes of visualization and analysis at the end of the study.

The data gathering system consisted of a collection of specialized modules which managed the different sensory streams, and a central module which kept track of the study state and which was responsible for the synchronization of the other modules. In order to track the heads and hands of the participants, a motion capture system was used in combination with customized tracking and recording software. To track the foam blocks in Tasks 1 and 2, a machine vision system was created and embedded within a special illuminated table that was constructed for the study. Video digitization and compression software was used to record from two camcorders mounted in the study environment. Finally, an additional module recorded the states of the two puzzle boxes used in Task 3. In total, the system managed data from 13 cameras and 6 different streams of information.

The study took place in the space in front of the Leonardo robot. The study made use of a number of the robot's cameras as well as the robot's networking infrastructure. Communication between the different recording modules was accomplished using a part of the robot's software architecture called IRCP (the Intra-Robot Communication Protocol). The modules communicated with each other over a dedicated gigabit network, allowing for high-bandwidth data transfer. This reuse of the robot's environment and infrastructure was for more than just the sake of convenience. It made possible a rapid transition from having the robot learn from sensory data recorded during the study interactions to having the robot learn live from highly similar sensory data generated by people interacting directly with the robot.

In the following sections, I describe the sensory modules and physical materials which were the essential parts of the data gathering system.

2.3.3 Motion Capture: Tracking Heads and Hands

In order to accurately track and record the head and hand movements of the study participants, I employed a Vicon motion capture system along with customized tracking and recording software.

Our Vicon system uses ten cameras with rapidly-strobing red light emitting diodes to track small retroreflective spheres that can be attached to clothing and other materials. The positions of these trackable markers can be determined with very high accuracy within the volume defined by the cameras - often to within a few millimeters of error at a tracking rate of 100-120Hz. The most common use for such systems is to record human motion data for 3D animation used in films and video games.

However, the Vicon system is designed to track a single human wearing a full-body motion capture suit, and thus its software is not well suited for tracking multiple independently-moving objects. The software is very sensitive to the presence of extraneous markers and to object occlusion. All tracked objects are required to remain in the environment at all times; if objects are removed, the software performance becomes slow and unreliable. These software design flaws can lead to frequent crashes, which can jeopardize data gathering from human subjects.

To address these issues, I developed a software toolkit for tracking rigid objects using the raw data provided by the Vicon system about individual marker positions. The toolkit identifies objects in the scene by searching for matches to tracking templates which specify the pairwise distances between markers mounted to the objects. The markers can be attached to the objects in any arrangement, with more distinguishable configurations leading to greater tracking reliability and lesser sensitivity to sensor noise and occlusion.

The toolkit supports the 3D, graphical visualization of live or recorded tracking data (see Figure 2-5) and allows for rapid, on-the-fly calibration of tracking templates as well as object bounding boxes and forward vectors. The toolkit is robust to missing objects, partially- and fully-occluded objects, intermittent tracking of individual markers, and the presence of extraneous markers and objects.

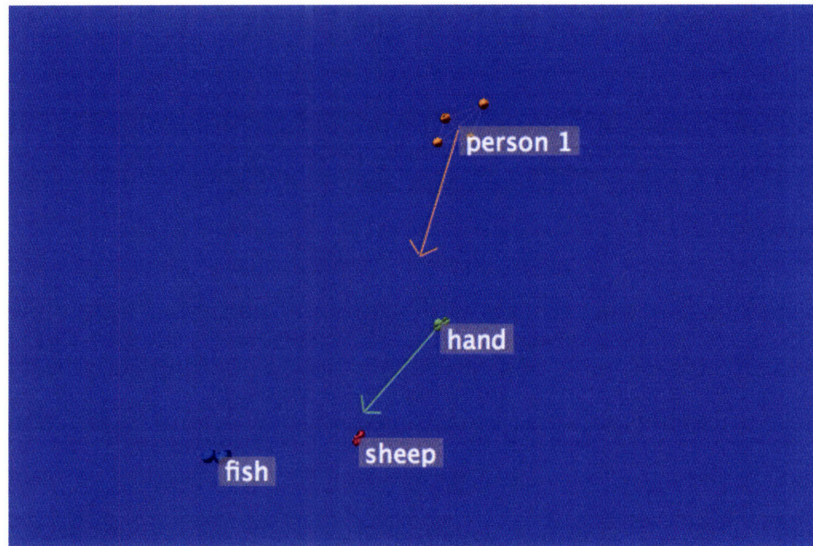


Figure 2-5: Graphical visualization within the object tracking toolkit, with two toy objects and a tracked head and hand with calibrated forward vectors.

For the data-gathering setup, I needed to attach reflective markers in rigid configurations to the heads and hands of the study participants. To this end, I constructed a set of large, wooden “buttons” that could be sewn onto various articles of clothing. The buttons were constructed by cutting 4-inch square pieces from a thin sheet of plywood, with four small holes per square.

For the heads, I purchased two baseball caps with elastic headbands. The caps featured two different MIT logos, which were selected as a neutral affiliation that most of the study participants had in common (as opposed to, say, the logo for a professional sports team, which could be quite divisive). A single wooden button was sewn onto each hat. Then, the buttons were covered with adhesive-backed Velcro, and the reflective Vicor markers were attached to the Velcro in unique configurations (see figure 2-6).

For the hands, I purchased two sets of spandex bicycle-racing gloves with adjustable wrist straps. The gloves were chosen because they were very lightweight, and left the fingers of the study participants free. Buttons were sewn onto the backs of each glove, and again the buttons were covered in Velcro and outfitted with unique arrangements of reflective markers.



Figure 2-6: Hats, gloves, and rings outfitted with trackable markers.

Additionally, to track the index fingers of the participants, I constructed four small rings. Each ring consisted of a very small reflective marker sewn onto a thin strip of double-sided velcro. By rolling the strips of velcro more or less tightly, the rings could be sized to fit any finger.

The baseball caps were worn backwards by the study participants so that their faces would not be occluded from each other or from the cameras. The index finger rings were worn between the first and second knuckle from the tip of the finger. The sizing of the trackable clothing seemed to work out very well: most study participants reported a comfortable fit for the apparel.

While the motion capture system was very well suited for tracking the positions and orientations of the participants' heads and hands, it was ruled out as a method for tracking the foam blocks in Tasks 1 and 2. The blocks were too small, and too numerous, to attach unique marker configurations to each block. Further, the density and proximity of the blocks on the tabletop would have caused problems for the template finding algorithm, as well as for the underlying dot finding algorithm used by the Vicon system. Finally, the markers on the blocks would have been frequently occluded by the hands of the participants, and often at the most critical of moments: when the participant was interacting with

the block!

In the following section, I discuss a better solution for tracking the blocks: a specially constructed, illuminated table with an embedded machine vision system.

2.3.4 The Light Table: Tracking Object Movement

In order to accurately measure the positions and orientations of the colorful foam blocks, a special illuminated table was constructed which tracked blocks on the table's surface. A camera mounted underneath the table looked up at the blocks through the transparent tabletop. The blocks were tracked using a machine vision system which combined color segmentation, shape recognition, and object tracking, as described later in this section.

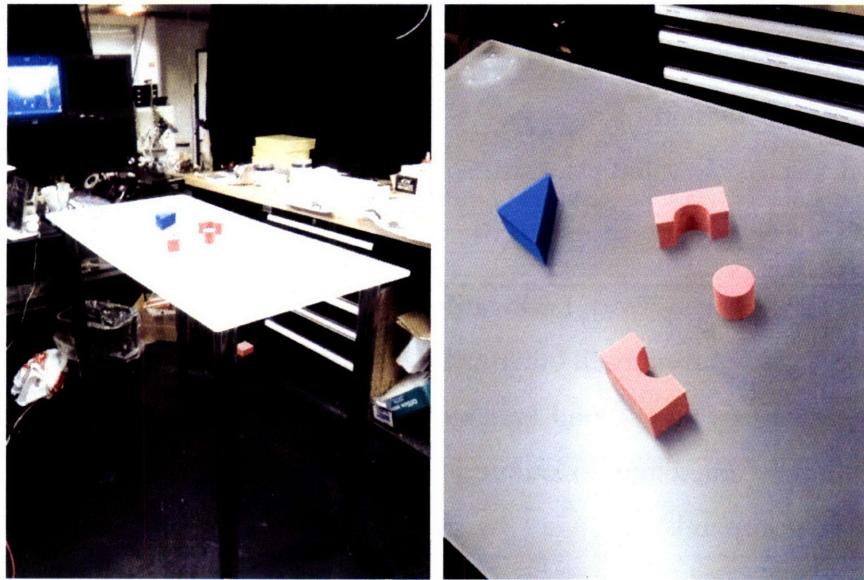


Figure 2-7: Initial assembly of the light table, with detail of the frosted acrylic top.

The basic table consisted of four cylindrical metal legs and a clear acrylic tabletop (see figure 2-7). The tabletop measured two feet by four feet, and was one-half of an inch thick. Acrylic was chosen instead of glass because it is much lighter and harder to break, and thus required a much less extensive support structure under the table, resulting in a clearer view for the camera. The legs were adjustable length, and were set to make the tabletop 38 inches high, a good height for people standing on either side of the table to

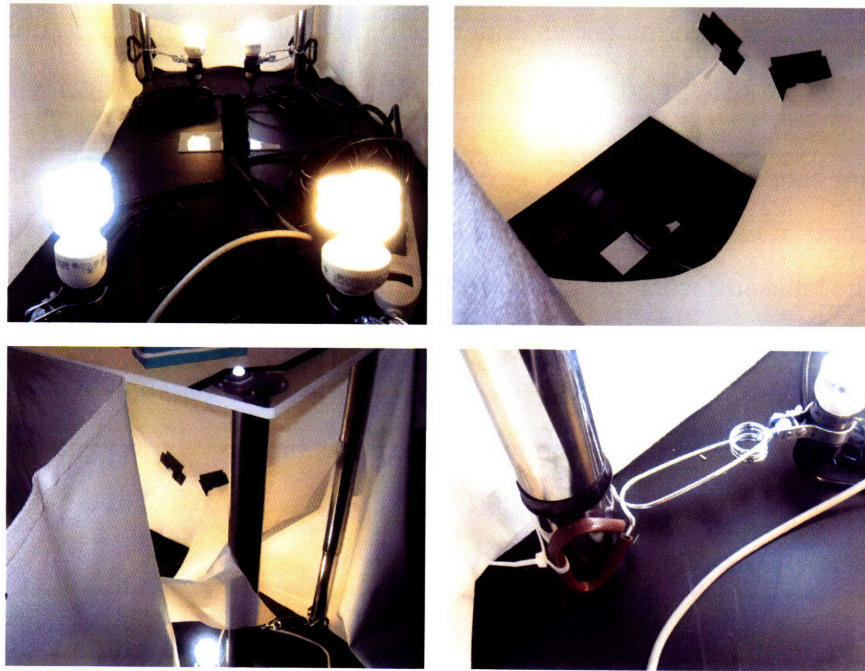


Figure 2-8: Light table component details. Top left: lights and camera mounted underneath the table. Top right: screens for diffusing the lights and preventing reflective glare. Bottom left: attachment of skirt and screen to table. Bottom right: clamp for holding the lights in place.

interact with objects on the tabletop. The legs were attached to the four corners of the table using machine screws set into the bottom surface of the tabletop, leaving the top surface completely smooth.

After the table was constructed, the acrylic top was frosted using a rotary sander and fine-grained sandpaper. Frosting was both for aesthetic purposes as well as for the benefit of the machine vision system. Frosting turned the tabletop into a thin, diffusive surface, meaning that objects pressed against the tabletop could be seen very clearly from below, but would become more and more diffuse as their distance up from the table surface increased. Thus, the blocks could be seen in sharp detail by the camera underneath the table, but the distracting effects of other objects such as hands, clothing, and external light sources were somewhat mitigated.

A skirt was constructed for the table using a long piece of white canvas cloth. The skirt was attached to the table perimeter using adhesive-backed Velcro, allowing for easy access

to the electronics and other materials underneath the table, as shown in figure 2-8. White canvas was selected to support the even diffusion of light underneath the table.

Significant care was taken to ensure a consistent, optimized lighting environment for the camera, and to ensure a consistent geometrical relationship between the camera, the lights, and the table surface. Four lights were installed underneath the table, each attached to a different table leg using a strong clamp and zip tie. The bulbs used were 75-watt-equivalent, compact fluorescent bulbs. To optimize the color separation of the foam blocks, it was determined that the best setup was to use a mix of two “warm white” (2700K color temperature) and two “cool white” (4100K) bulbs, with similarly colored bulbs positioned diagonally across from each other.

Since the tabletop was somewhat reflective, the bright lights introduced four “hot” spots where blocks placed directly above the lights were invisible to the camera. To solve this problem, two large screens were constructed out of vellum paper to diffuse the lights. The screens were mounted at an angle, sloping down towards the camera from high up on the table legs, and attached to the inner side of the skirt. With the screens in place, the table surface was illuminated brightly and evenly. Other light sources that impinged upon the camera, such as the overhead lights directly above the table, were turned off or removed. Care was also taken in the positioning of the Vicon cameras so that the LEDs mounted on those cameras did not shine directly into the camera under the table.

Images of the table surface were provided by a Videre Design DCAM firewire camera. While this is a stereo camera system, images from only one of the cameras were used to track the blocks. The camera was selected because it provided good image quality, and had a field-of-view which was well matched to the dimensions of the table.

Camera images were captured by a Linux machine running frame grabbing software, and streamed uncompressed across the gigabit network. This setup supported both live processing as well as recording of the camera images. Images were provided at a resolution of 320 by 240 pixels, at approximately 17Hz. During the study, camera images were time stamped and saved directly to disk as sequences of minimally-compressed JPEG images.

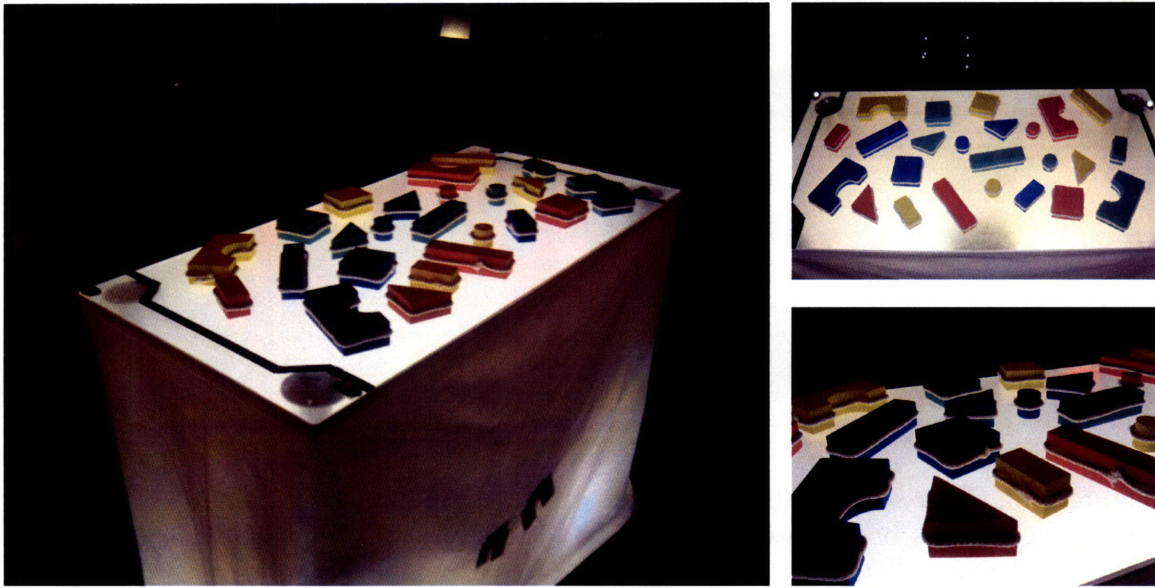


Figure 2-9: Final setup of the light table, with overhead view of the initial blocks configuration and detail of the rope bumpers on the foam blocks.

The final setup of the table and blocks is shown in figure 2-9. A few additional details should be noted. While the camera could see most of the table surface, it could not see blocks placed directly above the table legs or on the far edges of the table. So, thin borders were drawn on either side of the table using black electrical tape to delineate the preferred workspace for the blocks. When participants were first introduced to the blocks, they were told that they could do whatever they wanted with the blocks - pick them up, pass them around, put them on their head - but that when they put the blocks down, they were asked to place them flat (as opposed to stacked up) and between the two black lines. The participants were given no other instructions or constraints about the blocks.

As can be seen in figure 2-9, rope bumpers were constructed and placed around the perimeter of each block, midway down the sides. The bumpers were constructed using clothesline, and attached to the blocks using dressmaker's pins. The bumpers assisted the machine vision system in two ways. First, they ensured a minimum spacing between adjacent blocks, reducing the likelihood that two blocks of the same color, placed next to each other, would be perceived as one large block of that color. Second, the bumpers introduced a subtle affordance cue about which block face should lie on the tabletop, since

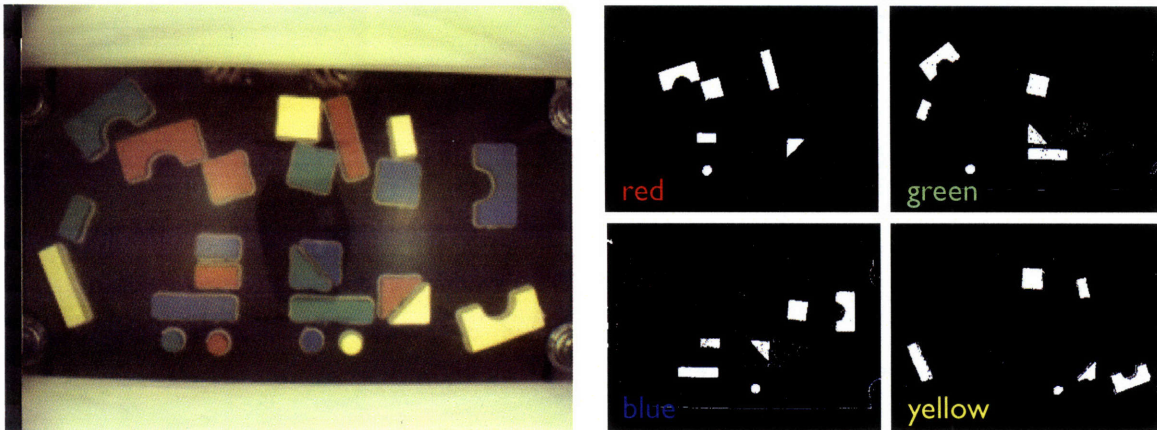


Figure 2-10: Color segmentation converted each incoming image into four separate color probability images, one for each color of block. These probability images were thresholded to create binary color masks.

the blocks would stick up at an odd angle if placed down on one of the faces to which the bumper was attached. This simplified the job of the block shape recognition algorithm.

Additionally, reflective markers were mounted in each corner of the table so that the position and orientation of the table itself could be tracked by the motion capture system.

I now turn to the machine vision system that was created for tracking the blocks. Camera images of the table surface were processed in a number of stages. First, color segmentation was used to identify pixels that were associated with the red, green, blue, and yellow blocks. Next, a blob finding algorithm identified the locations of possible blocks within the segmented images. Then, a shape recognition system classified each blob as one of the six possible block shapes. Finally, an object tracking algorithm updated the positions and orientations of each block using these new observations in conjunction with historical information about each block.

Color models were constructed for each of the four block colors in the study. An interface was developed so that color models could be built interactively by the user, by clicking directly on live or recorded images from the camera. Pixels selected by the user for a given color were collected into a sample set and projected into the hue-saturation-value (HSV) color space. The value component for each sample was discarded, resulting in a sample

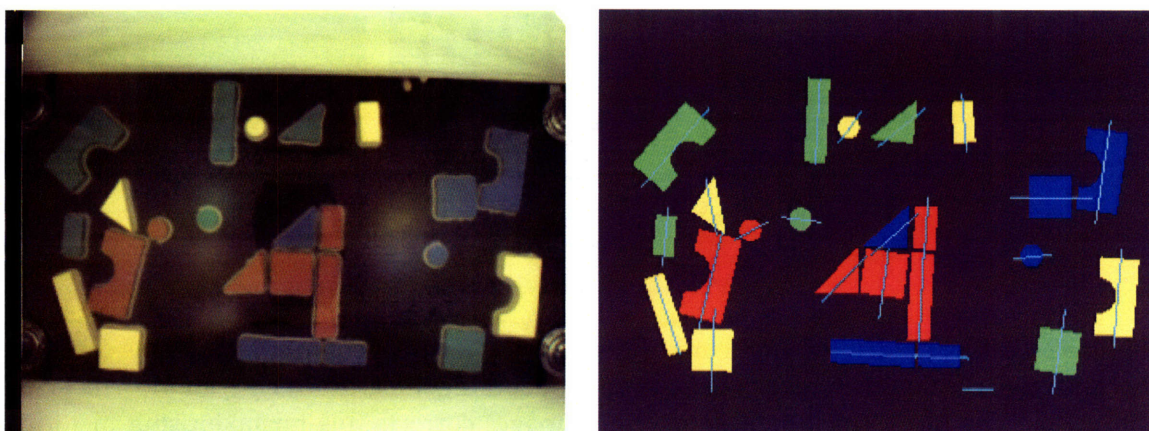


Figure 2-11: The block shape recognition system produced an estimate of the shape, orientation, and position of each blob of color in the image.

set of hue-saturation pairs. Next, a gaussian mixture model was fitted to these samples using expectation maximization (EM). To increase computational efficiency, these mixture models were then discretized into two-dimensional histograms. Thus, the color calibration process resulted in the creation of four hue-saturation histograms, one for each color.

During the color segmentation process, each incoming image was converted to the HSV color space. Then, histogram back-propagation was performed to create a color probability image for each of the four color models. These color probability images were then thresholded to create binary images (color masks), where each pixel in the image indicated whether or not the corresponding pixel in the original image matched the given color model.

Each color mask (thresholded color probability image) was then handed to a blob finding algorithm. The blob finder performed a depth-first search over the color mask to identify every contiguous region of the given color in the image. All blobs smaller than a given minimum size threshold were discarded, and the remaining blobs were handed to the shape recognition system.

The shape recognition system was responsible for classifying each blob as one of the six possible block shapes, and also for estimating the position and orientation of each blob given its shape classification. The system first calculated a number of important, differen-

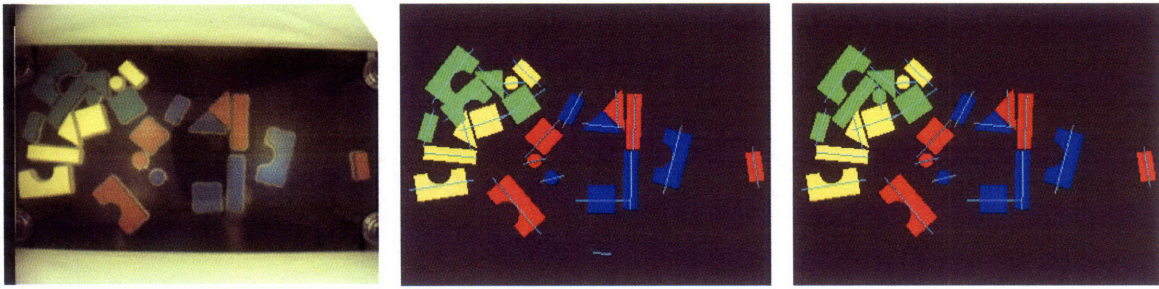


Figure 2-12: Block tracking. In the middle image, the shape recognition system is confused because two of the green blocks have been blobbed together, producing an incorrect classification. The block tracking system (right image) corrects this mistake using historical information.

tiating features for each blob, such as its mass, length-to-width ratio, symmetry, and so on. The system then used a classification tree operating on these geometrical features to assign a shape classification to each blob, or to reject the blob as not being a good match for any of the possible shapes.

Finally, an object tracking algorithm used the classified blobs to update the tracked positions and orientations of each block using the new observations in conjunction with historical information about the blocks. Since the shape recognition system typically produced very high-fidelity results, the object tracking algorithm could be implemented relatively simply. The algorithm made use of the fact that, for the most part, all 24 of the blocks would be on the table at all times, except for brief periods of movement when blocks might disappear from one part of the table and reappear on another part of the table.

The tracking algorithm used a three-tier system whereby each tracked block could lay “claim” to one of the newly classified blobs. Conflicts were resolved based on which block was closest to the claimed blob. While blocks could temporarily lay claim to blobs whose shape did not match their own, their positions and orientations would only be updated when they were matched to blobs with the same color and shape. In the first tier, blocks could lay claim to “strong matches,” nearby blobs whose color and shape matched their own. Next, unmatched blocks could lay claim to “loose matches,” nearby blobs whose color matched their own but whose shape did not. These loose matches were only allowed to be claimed for a short period of time. Finally, unmatched blocks became “free agents”

after a short period of time, and were allowed to claim any unclaimed blob on the table whose color and shape matched their own.

The object tracking algorithm worked very well, and was successful at cleaning up many of the mistakes that would occasionally arise from sensor noise, adjacent blocks of the same color being temporarily blobbed together, and other factors. Overall, I was very pleasantly surprised at the accuracy of the vision system in measuring the positions and orientations of the blocks over the course of the study.

2.3.5 The Puzzle Boxes: Tracking Object Manipulation

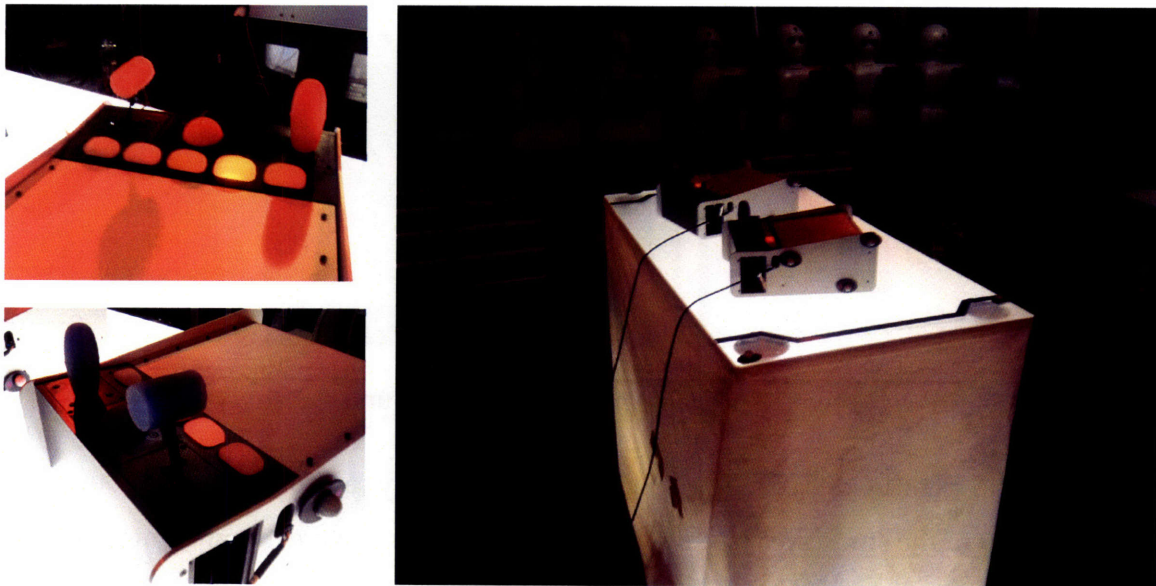


Figure 2-13: Puzzle boxes in the study environment, on top of the light table.

During study Task 3, the states of the lights and controls of both puzzle boxes were recorded. Since the boxes were computer-controlled, this recording module was quite easy to implement. No sensors were required beyond those already present in the box switches and buttons. On each update tick, the control software simply recorded a time stamp, along with the current states of all of the box controls, lights, and auditory events. Recording took place at approximately 60Hz.

In order to track the positions and orientations of the two boxes relative to the partici-

pants' hands and heads, reflective markers were mounted on each box, so that they could be tracked by the motion capture system. The full setup of the puzzle boxes in the study environment is shown in figure 2-13.

2.3.6 Video Recording

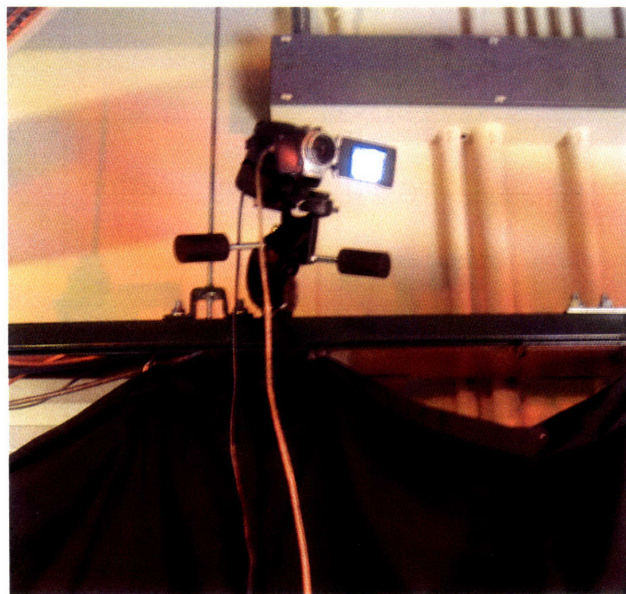


Figure 2-14: Two camcorders were mounted around the study area.

In order to record video of the two participants, two Sony camcorders were mounted at the periphery of the study area (see figure 2-14). Both cameras were mounted high so as to have a relatively unobstructed view of the tabletop, with one camera angled towards the hands and face of the teacher and the other camera angled towards the learner. Additionally, the cameras were aimed so as to avoid looking directly at any of the Vicon cameras, whose strobing red LEDs caused undesirable streaks and artifacts in the camcorder image.

Instead of being recorded to tape, the camcorder video was digitized and compressed live during the study interactions, preventing a considerable amount of work and tedium at the end of the study. This had the important additional benefit that the video could be automatically time stamped and synchronized with the other sensory streams.

To process the camcorder video, I used the BTV Pro shareware application created by

Ben Bird. The software was set up to record the camera data as MPEG-4 compressed video streams with a resolution of 640 by 480 pixels. In order to automatically start and stop this recording process, a wrapper module was created using our software codebase. The wrapper module received start and stop commands over the network from the central study module, which were then relayed to the BTV Pro application via the AppleScript scripting language. A small but noticeable delay (about 1 second) was associated with starting and stopping the recording process. In order to accommodate this, the wrapper module sent receipts back to the central module with the actual recording start and stop times, enabling accurate synchronization of the video data.

2.3.7 Data Stream Management and Synchronization

A central recording module kept track of the study state and managed the synchronization of the other recording modules. The central module was designed to be controlled via a simple graphical user interface as well as via a wireless presentation remote, giving a single experimenter the ability to control the complete recording infrastructure. A simple display presented the status of the various recording modules: "idle," "recording," or "off-line." Using the remote, the experimenter could advance to the next task in the random task sequence, and also start and stop recording for each task.

The central recording module generated a log of task start and stop times for each study session. The module was responsible for sending start and stop commands over the network to the other recording modules, along with information about which file names to use for the various data streams.

The video recording processes which controlled the two camcorders ran on two separate Macintosh tower computers - a G5 Power Mac and an Intel Mac Pro. The other recording modules, which recorded the motion capture data, the puzzle box data, and the stream of images from the light table camera, were all run on the same computer as the central module - a 4-processor, 3GHz Intel Mac Pro. Thus, with the exception of the video recording modules, which were synchronized as described in the previous section, the

other modules could all be synchronized using the main computer's system clock, which was used to time stamp all of the remaining data streams.

2.3.8 Study Execution and Discussion

I now turn to a discussion of the execution of the study itself, and present some qualitative observations about the teaching and learning behavior exhibited by participants on the various study tasks.

As mentioned above, data was gathered from 72 study participants, grouped into 36 pairs. In total, over 130 gigabytes of raw data was recorded, representing approximately 10 hours of observed task interaction. Over the course of the study, the data gathering framework performed very well and with minimal loss of data, meaning that none of the study interactions were dropped or discarded from the final data set. On two occasions, the data from one of the two camcorders was interrupted, but in both cases the connection was restored in under a minute. These were the only problems encountered with the recording systems.

My analysis focused on two of the study tasks: Tasks 2a and 2b, two of the secret-constraint tasks involving the foam blocks. These tasks were selected because they were the most vigorously interactive tasks in the study (and also, interestingly, the tasks that study participants seemed to enjoy the most). Since neither participant had enough information to complete the task on their own, these tasks required the direct engagement and cooperation of both participants. Correspondingly, I observed a rich range of dynamic, interactive behaviors during these tasks.

To identify the emphasis and de-emphasis cues provided by the teachers in these tasks, an important piece of "ground-truth" information was exploited: for these tasks, some of the blocks were "good," and others of the blocks were "bad." In order to successfully complete the task, the teacher needed to encourage the learner to use some of the blocks in the construction of the figure, and to steer clear of some of the other blocks. In Task 2a, the blue and red blocks were "good," while the green and yellow blocks were "bad." In Task

2b, the triangular blocks were good, the square blocks were bad, and the remaining blocks fell into a neutral “other” category.

To set the stage, I will first describe two pairs of study interactions before diving into a more detailed analysis of the observed cues. The first pair of interactions were for Task 2a, where the goal was to construct a sailboat figure using only red and blue blocks. In one recorded interaction (session 27), the teacher is very proactive, organizing the blocks almost completely before the learner begins to assemble the figure. The teacher clusters the yellow and green blocks on one side of the table and somewhat away from the learner. The learner initially reaches for a yellow triangle. The teacher shakes her head and reaches to take the yellow block back away from the learner, before continuing to organize the blocks. The learner proceeds to complete the task successfully.

In another recorded interaction (session 7), the teacher’s style is very different. Instead of arranging the blocks ahead of time, he waits for the learner to make a mistake, and then “fixes” the mistake by replacing the learner’s block with one that fits the constraint. When the learner positions a green rectangle as part of the mast of the sailboat figure, the teacher quickly reaches in, pulls the block away, and replaces it with a red rectangle. Later, the teacher fixes a triangular part of the sail in a similar way, after which the learner completes the task successfully.

The second pair of interactions were for Task 2b, where the goal was to construct a truck/train figure using all of the triangular blocks, and none of the square blocks. In one interaction (session 2), the teacher provides some very direct structuring of the space, pulling the square blocks away from the learner and placing the triangular blocks in front of her. In contrast, in another interaction (session 21), the teacher almost entirely refrains from moving the blocks. She instead provides gestural feedback, tapping blocks and shaking her hand “no” when the learner moves an inadmissible block, and nodding her head when the learner moves an acceptable block.

As these descriptions suggest, I observed a wide range of embodied cues provided by the teachers in the interactions for these two tasks, as well as a range of different teaching

Table 2.2: Cues of positive emphasis. Embodied cues provided by the teacher and directed toward good blocks.

Simple Hand and Head Cues
tapping with the index finger
touching with the index finger
pointing
framing with both hands of clustered good blocks
targeting by gaze
Block Movement Cues
block movement towards learner's body or hands
block movement towards center of table
addition of block to figure (often, via replacement of a bad block)
placement of blocks along edge of table closest to learner
clustering with other good blocks
Compound Cues
head nodding accompanying pointing or hand contact with block
head nodding following learner's pointing or hand contact with block
shrugging gesture following learner's block movement - "I don't know/seems OK"
"thumbs up" gesture following pointing or sequence of pointing gestures
pointing back and forth between clustered good blocks and the learner
Emphasis Through Inaction
observation of learner's actions, accompanied by lack of intervention
passing over block in process of providing negative emphasis

styles. Table 2.2 enumerates some of the cues of positive emphasis that were observed. These were cues provided by the teachers and directed towards good blocks, in the process of guiding the learners to interact with these blocks. Table 2.3 enumerates some of the negative cues that were observed, which tended to steer the learners away from the targeted blocks.

Positive cues included simple hand gestures such as tapping blocks, touching blocks, and pointing at blocks with the index finger. Teachers sometimes used both hands to frame the space occupied by single good blocks or collections of blocks to use. These cues were often accompanied by gaze targeting, or looking back and forth between the learner and the target blocks.

Table 2.3: Cues of negative emphasis. Embodied cues provided by the teacher and directed toward bad blocks.

Simple Hand and Head Cues
covering blocks with the hands
holding blocks fast with the fingers
prolonged contact with blocks despite proximity of learner's hands
interrupting learner's reaching action by blocking learner's hand
interrupting learner's reaching action by touching or lightly slapping learner's hand
Block Movement Cues
block movement away from learner's body or hands
block movement away from center of table
removal of block from figure (sometimes followed by replacement with a good block)
placement of blocks along edge of table closest to teacher
clustering with other bad blocks
interrupting the learner's reaching action by grabbing away the target block
Compound Cues
head shaking accompanying pointing or hand contact with block
head shaking following learner's pointing or hand contact with block
"thumbs down" gesture following pointing or sequence of pointing gestures
vigorous horizontal "chop" gesture
hand "wagging" gesture (with index finger or all fingers extended)
large "X" symbol formed using both forearms
Emphasis Through Inaction
passing over block in process of providing positive emphasis

Simple negative cues included covering up blocks with the hands, preventing visibility of and physical access to the blocks. Blocks were occasionally held fast by the teachers, so that they could not be used by the learners, or were kept in prolonged contact by the teachers despite the proximity of the learner's hands. Teachers would occasionally interrupt reaching motions directly by blocking the trajectory of the motion or even by touching or (rarely) lightly slapping the learner's hand.

A number of compound cues were observed, often involving the specification of a particular block via pointing or tapping, accompanied by an additional gestural cue suggesting the valence of the block. Such gestures included head nodding, the "thumbs up" gesture, and even shrugging, which often seemed to be interpreted as meaning "I'm not

sure, but it seems OK.” Teachers nodded in accompaniment to their own pointing gestures, and also in response to actions taken by the learners, including actions that seemed to be direct queries for information from the teacher.

Correspondingly, a number of compound, negative cues were observed. These included head shaking, the “thumbs down” gesture, and side-to-side finger or hand wagging gestures. A number of teachers used a vigorous horizontal “chop” gesture to identify bad blocks (with the palm down, the hand starts out near the center of the body and then moves rapidly outward and down). Another negative gesture was a large “X” symbol formed using both forearms.

Inaction, in some contexts, was another important emphasis cue. When the teacher watched the learner’s actions and did not intervene, this often seemed to be interpreted as positive feedback. In a similar vein, when the teacher’s gestures passed over particular blocks on the way to provide negative or positive feedback about other blocks, the passed-over blocks could be seen as implicitly receiving some of the opposite feedback. For example, when the teacher left some blocks in place, while selecting other blocks and moving them away from the center of the table, the blocks left behind seemed to attain some positive status.

Another important set of cues were cues related to block movement and the use of space. To positively emphasize blocks, teachers would move them towards the learner’s body or hands, towards the center of the table, or align them along the edge of the table closest to the learner. Conversely, to negatively emphasize blocks, teachers would move them away from the learner, away from the center of the table, or line them up along the edge of the table closest to themselves. Teachers often devoted significant attention to clustering the blocks on the table, spatially grouping the bad blocks with other bad blocks and the good blocks with other good blocks. The learner’s attention could then be directed towards or away from the resulting clusters using many of the gestural cues previously discussed. These spatial scaffolding cues were some of the most prevalent cues in the observed interactions. In particular, the teachers in the study, with very few exceptions, consistently used movements towards and away from the body of the learner to encourage

the use of some of the blocks on the table, and discourage the use of other blocks.

Having observed this collection of embodied emphasis cues, my next step was to establish how reliable and consistent these cues were in the recorded data set, and most importantly, how useful these cues were for robotic learners. In the next section, I describe the tools I developed for automatically analyzing some of these cues and generating quantitative data about their reliability. In later chapters, I describe the development of an even more powerful tool: a robotic learning architecture for directly assessing the utility of these cues for automated learning systems.

2.3.9 Data Analysis Tools and Pipeline

In this section, I describe the automated data analysis tools that were created to detect and measure the emphasis and de-emphasis cues that teachers provided in the study. I then present the quantitative results generated by these tools.

For all of the quantitative and evaluative data presented in this thesis, a cross-validation methodology was followed. All of the data analysis tools and learning algorithms were implemented and tested using a small set of study interactions, pulled from just 6 of the 36 study sessions. Reported data were generated by running these same tools and algorithms over the remaining 30 study sessions. While these sessions did not represent completely “blind” data, since I was present when their data was initially collected, I believe that this methodology was nevertheless valuable for minimizing the risk of overfitting by the systems presented in this thesis.

I developed a set of data visualization tools to assist with the playback and analysis of the study data. My data visualization environment is shown in figure 2-15. Using a simple graphical interface, the user could select a study session by number and choose a particular task to load in for analysis. A time scrubber allowed the user to control the position and speed of playback of the synchronized data streams. The two windows at the bottom of the screen displayed camcorder footage from the two different camera recording angles. In both shots, the teacher is on the left and the learner is on the right. The big blue

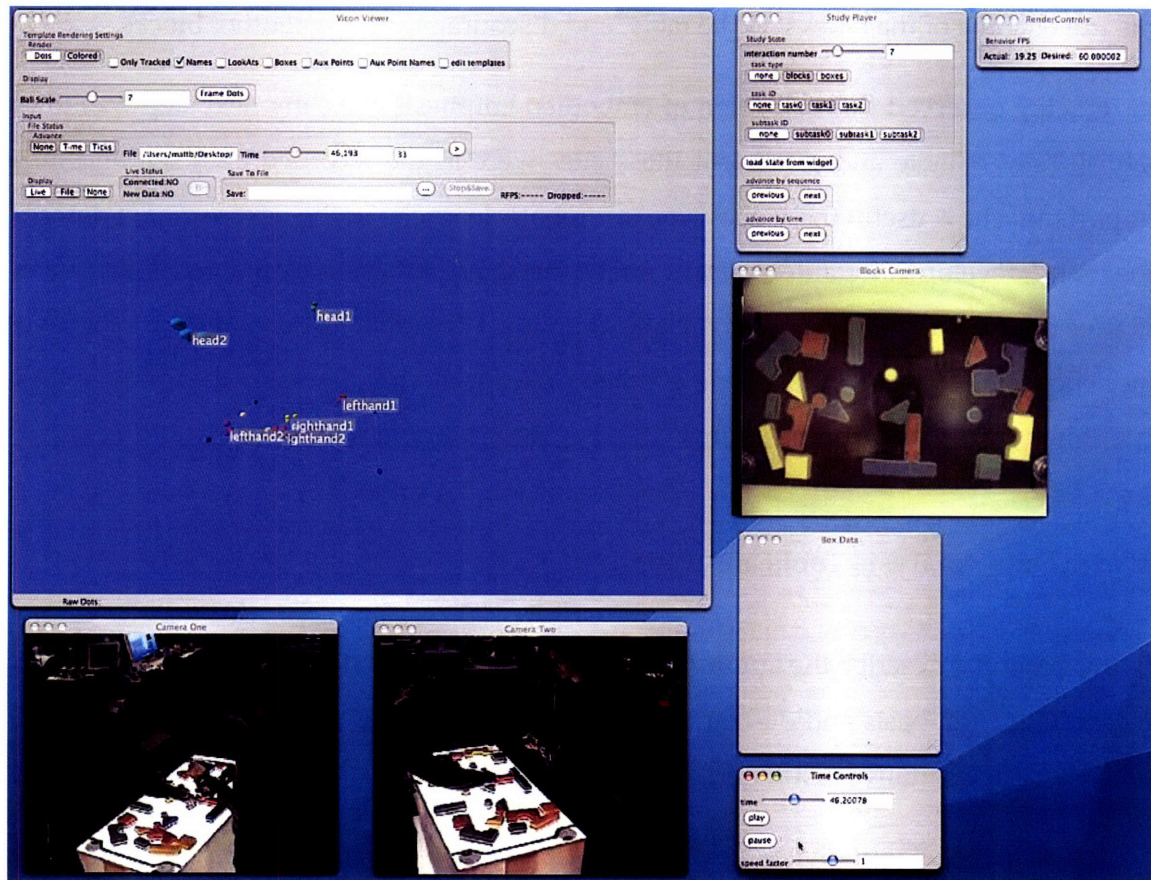


Figure 2-15: Data visualization environment.

window presented the motion capture data, with tracking information about the position and orientation of the table, heads, and hands. The window on the right of the screen showed the view from the under-table camera. In this view, the teacher is at the top of the screen and the learner is at the bottom. This environment could also be run in “batch-mode” for automated traversal and analysis of the data set.

As mentioned previously, my quantitative analysis focused on two of the study tasks: Tasks 2a and 2b, two of the secret-constraint tasks involving the foam blocks. To analyze the behavioral data for these tasks, I developed an automated pipeline for extracting high-level events from the raw, recorded sensor data. This pipeline proceeded in a number of stages. In the first stage, the recorded data about the positions of the reflective markers was run through the rigid object tracking system, producing a trace of the positions and

orientations of the heads, hands, table, and boxes (for more information on this system, see section 2.3.3, above). Also in this stage, the images from the under-table camera were run through the block tracking system, producing a trace of the positions and orientations of all of the foam blocks (see section, 2.3.4, above). By recording the output of these tracking systems, I produced compact streams of information about the objects in the study environment, information that could then be randomly traversed and queried in later stages of processing.

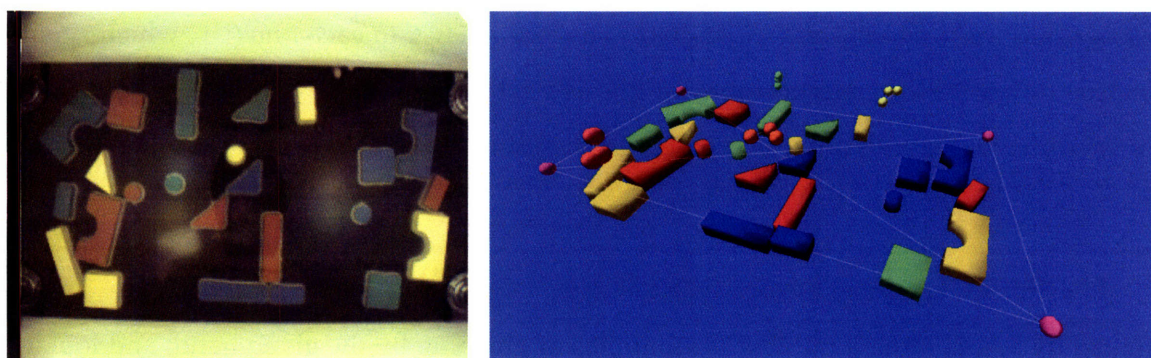


Figure 2-16: The blocks were mapped into the coordinate frame of the motion capture system, allowing for 3D visualization and analysis of all of the tracked objects. On the right, the scene is shown from a point of view over the learner's shoulder. The learner's right hand, shown as orange dots, is hovering over the red triangular block.

In the next stage, the tracking information about the foam blocks was mapped into the coordinate system of the motion capture system, so that all of the tracked study objects could be analyzed in the same, three-dimensional frame of reference. To accomplish this, a correspondence was established between four pixel locations in the under-table camera image and four positions that were specified relative to the rigid arrangement of reflective markers attached to the table. The four locations that were selected were the endpoints of the black tape lines that delineated the block tracking region, since these spots could be easily identified from both below the table as well as from above. These four correspondence points were used to calculate a linear transformation for mapping between image coordinates and the spatial coordinates of the motion capture system. A simple interface was created which allowed the user to click on the correspondence points in the camera image stream. Luckily, the mounting of the under-table camera proved to be steady enough

that the camera did not move perceptibly relative to the table over the course of the study, so this correspondence only needed to be specified once for the entire set of collected data.

With all of the study objects now in the same frame of reference, the next stage of processing used spatial and temporal relationships between the blocks and the bodies of the participants to extract a stream of potentially salient events that occurred during the interactions. These events included, among other things, block movements and hand-to-block contact events, which were important focal points for my analysis. My processing system recognized these events, and attempted to ascribe agency to each one (i.e., which agent - learner or teacher - was responsible for this event?). Finally, statistics were compiled looking at different features of these events, and assessing their relative utility at differentiating the “good” blocks from the “bad” blocks.

In order to recognize block contact events, a “grab” location for each hand was estimated by aggregating a number of examples of grasping by the given hand, and looking at the position of the grasped block relative to the tracked location of the hand (grasping typically occurred in the palm, whereas the reflective markers were mounted to the glove on the back of the hand). Subsequent block contact events were recognized by simply thresholding the distance between this “grab” location and the centroid of potential target blocks.

A detector for raw block movements was constructed on top of the block tracking system, and primarily looked at frame-to-frame changes in position for each block to classify motion. A small distance threshold was used to classify block motion rather robustly. To recognize temporally-extended block movements, this raw motion detector was combined with a simple temporal filter which filtered out very brief movements as well as brief moments of stillness within extended movements, thus smoothing out block motion trajectories. Agency was ascribed to each recognized block movement event by determining whose hand was closest to the given block throughout the duration of the movement.

By running the interaction data through these event recognizers, I produced a stream of sequential event information corresponding to some of the high-level actions taken by

the study participants during the tasks. I then analyzed some interesting features of these events to assess their relative ability to differentiate good blocks from bad.

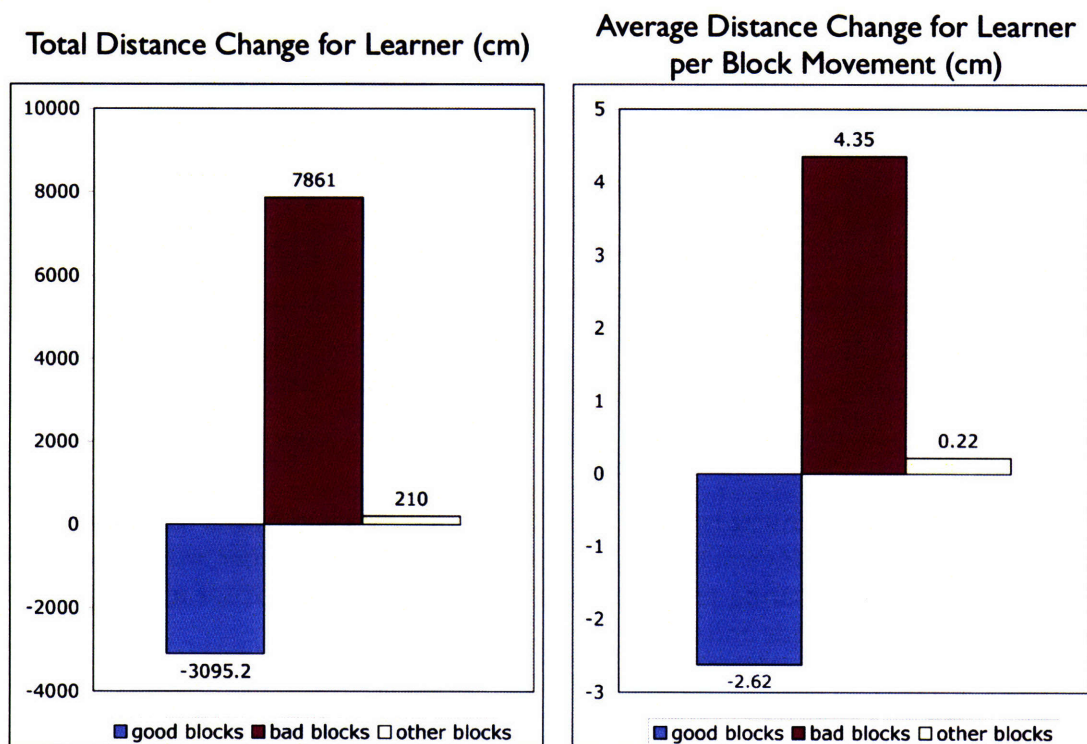


Figure 2-17: Change in distance to the body of the learner for block movements initiated by the teacher. Negative values represent movement towards the learner, while positive values represent movement away from the learner.

One of the most interesting features that I analyzed was movement towards and away from the bodies of the participants. The results of my analysis are summarized in figures 2-17 and 2-18. As can be seen in figure 2-17, the aggregate movement of good blocks by teachers is biased very substantially in the direction of the learners, while the aggregate movement of bad blocks by teachers is biased away from the learners. In fact, over the course of all of the 72 analyzed interactions, teachers differentiated the good and bad blocks by more than the length of a football field in terms of their movements relative to the bodies of the learners.

Looking at individual movements, travel towards and away from the body of the learner was strongly correlated with whether or not the given block was good or bad.

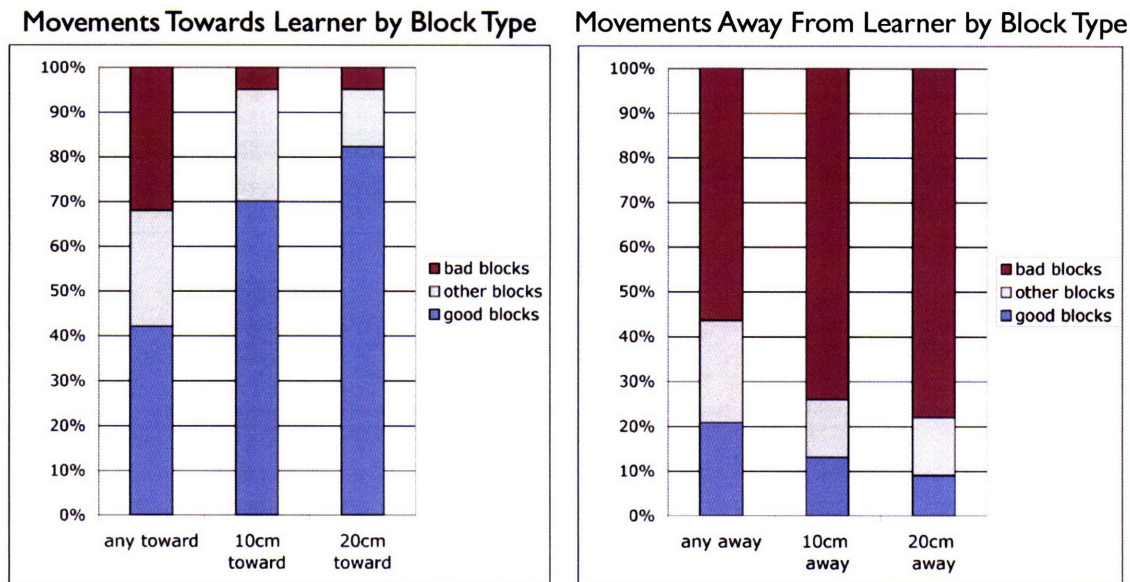


Figure 2-18: Movements towards the body of the learner initiated by the teacher were predictive of good blocks. Movements away from the body of the learner were predictive of bad blocks. The differentiating power of these movements increased for more substantial changes in distance towards and away.

Movements towards the body of the student were applied to good blocks 42% of the time, bad blocks 32% of the time, and other blocks 26% of the time. For movements that changed the distance towards the learner by more than 10cm, these differences were more pronounced: 70% of such movements were applied to good blocks, 25% to other blocks, and just 5% to bad blocks. For changes in distance of 20cm or greater, fully 83% of such movements were applied to good blocks versus 13% for other blocks and 5% for bad blocks. A similar pattern was seen for block movements away from the body of the learner, with larger changes in distance being strongly correlated with a block being bad, as shown in figure 2-18.

These results are very exciting for a number of reasons. First, I have used an entirely automated data processing system to produce quantitative results that match up well with the qualitative observations of what happened in the interactions between the study participants. This is an encouraging validation of my data-gathering technology and methodology. Second, I have identified an embodied cue which might be of significant value to

Movements Within 5 Seconds by Block Type

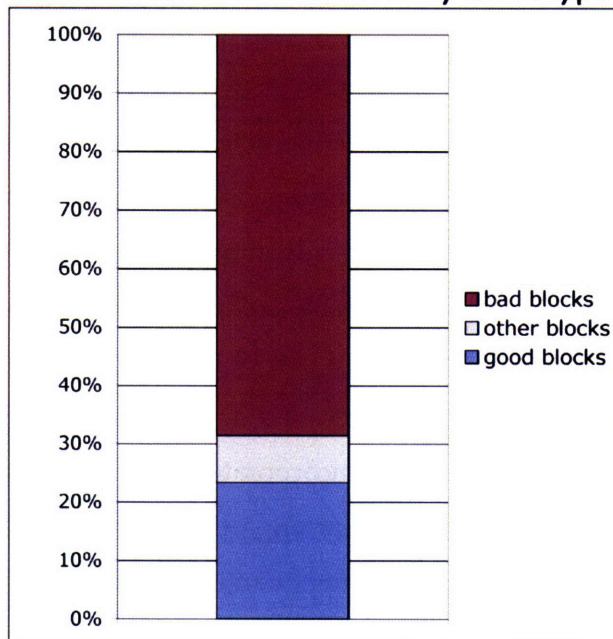


Figure 2-19: Learners moving a block, followed by teachers moving the same block within 5 seconds, was highly indicative of the given block being bad.

a robotic system learning in this task domain. The results suggest that such a robot, observing a block movement performed by a teacher, might be able to make a highly reliable guess as to whether the target block should or should not be used by measuring the direction and distance of the movement. Such a cue, which can be interpreted simply and reliably even within the context of a chaotic and fast-paced interaction, is exactly what I was looking for.

Another feature that I analyzed was the timing of the actions taken by the teacher relative to the actions of the learner. The results are summarized in figure 2-19. As can be seen, when the student moves a block, and then the teacher moves the same block within 5 seconds, the given block is good only 23% of the time, bad 68% of the time, and other 8% of the time. Thus, I have identified another highly reliable cue that a robot might be able to use to discover which blocks to steer clear of in this task domain.

2.4 Summary: Cues That Robotic Learners Need to Understand

In this chapter, I have presented the results of two studies highlighting the usefulness of embodied cues including visual perspective, action timing, and spatial emphasis to humans engaging in realistic teaching and learning tasks. These are clear, useful cues that real teachers provide and that real learners take advantage of. Robots should also pay attention to these cues in order to learn more naturally and effectively from humans in social environments. In the next chapter, I describe a pair of learning algorithms that I constructed to directly evaluate the utility of these cues, and describe the mechanisms of social attention and visual perspective taking that I incorporated into the Leonardo robot's cognitive architecture. In chapter 4, I present the learning performance of the robot on benchmark tasks drawn from the two studies, providing quantitative evidence for the utility of visual perspective, action timing, and spatial scaffolding as attention-direction cues for robotic learning systems.

Chapter 3

Learning Algorithms and Architecture

In this chapter, I describe a pair of learning algorithms that I constructed to evaluate the utility of the cues identified in the studies: visual perspective, action timing, and spatial scaffolding. I situated these learning algorithms within a large architecture for robot cognition, augmented with novel mechanisms for social attention and visual perspective taking. Designing an integrated learning system in this way supported not only the direct evaluation of the robot's performance on benchmark tasks drawn from the studies, but also a number of demonstrations of interactive social learning.

In this chapter, I first provide an overview of the self-as-simulator cognitive architecture. I then describe the key components of this architecture, focusing on mechanisms for understanding the environment from the visual perspective of the teacher, social mechanisms of attention direction and enhancement, and a unified framework for social action recognition and behavior generation. Finally, I describe the algorithms that I developed within this architecture for learning tasks from human teaching behavior, and describe how these algorithms take advantage of visual perspective, action timing, and spatial scaffolding cues.

The architecture, along with all of the demonstrations and evaluations described in this thesis, was designed to run on the 65 degree of freedom humanoid robot Leonardo and its graphical simulator, shown in figure 3-1.



Figure 3-1: The Leonardo robot and graphical simulator

3.1 Self-as-Simulator Cognitive Architecture

My approach to endowing machines with socially-cognitive learning abilities is inspired by leading psychological theories and recent neuroscientific evidence for how human brains might infer the mental states of others and the role of imitation as a critical precursor. Specifically, *Simulation Theory* holds that certain parts of the brain have dual use; they are used to not only generate our own behavior and mental states, but also to predict and infer the same in others. To understand another person's mental process, we use our own similar brain structure to simulate the introceptive states of the other person [Davies and Stone, 1995, Gallese and Goldman, 1998, Barsalou et al., 2003].

For instance, Gallese and Goldman [Gallese and Goldman, 1998] propose that a class of neurons discovered in monkeys, labeled mirror neurons, are a possible neurological mechanism underlying both imitative abilities and Simulation Theory-type prediction of the behavior of others and their mental states. Further, Meltzoff and Decety [Meltzoff and Decety, 2003] posit that imitation is the critical link in the story that connects the function of mirror neurons to the development of mindreading. In addition, Barsalou [Barsalou et al., 2003] presents additional evidence from various social embodiment phenomena that when observing an action, people activate some part of their own representation of that action as well as other cognitive states that relate to that action.

Inspired by this theory, my simulation-theoretic approach and implementation enables a humanoid robot to monitor an adjacent human teacher by simulating his or her behavior within the robot's own generative mechanisms on the motor, goal-directed action, and perceptual-belief levels. This grounds the robot's information about the teacher in the robot's own systems, allowing it to make inferences about the human's likely beliefs in order to better understand the intention behind the teacher's demonstrations.

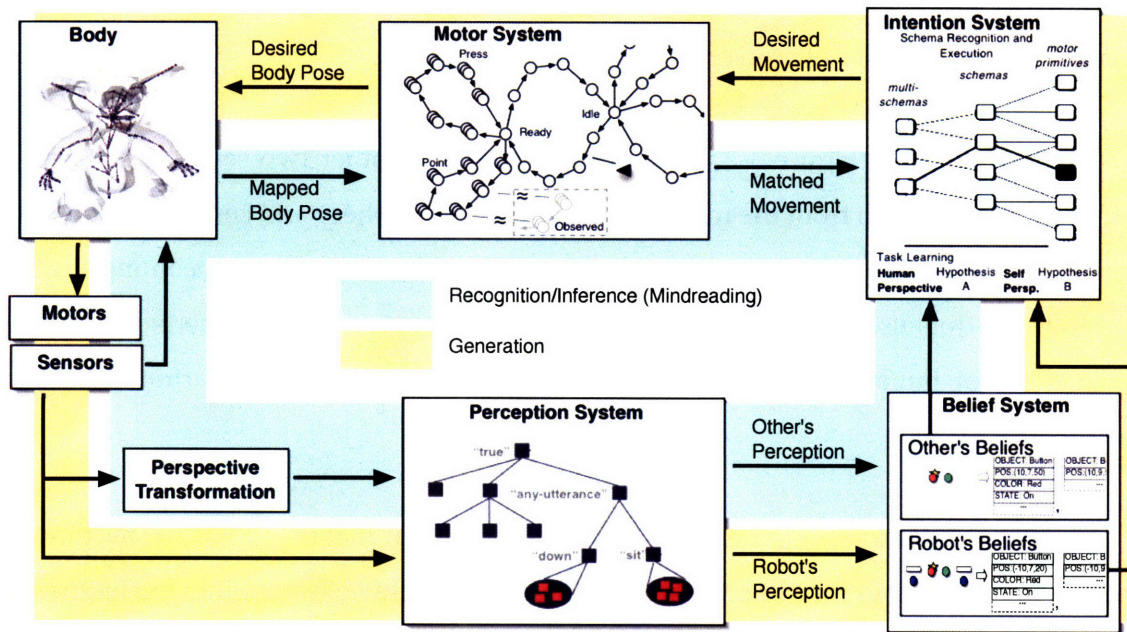


Figure 3-2: System architecture overview.

An overview of the robot's cognitive architecture, based on [Blumberg et al., 2002] and [Gray, J. et al., 2005], is shown in Figure 3-2. The system diagram features two concentric loops highlighting the primary flow of information within the architecture. The outer loop is the robot's main behavior generation pipeline. Information from the environment is provided by the robot's sensors. Incoming sensory events are processed by the Perception System, which maintains a tree of classification functions known as the percept tree. Perceptual information extracted by the percept tree is then clustered into discrete object representations known as object beliefs by the robot's Belief System. Object beliefs encode the tracked perceptual histories of the discrete objects that the robot perceives to exist in its environment.

The robot's object beliefs, in conjunction with internal state information, affect the decisions made by the robot's planning and action selection mechanisms. These action decisions result in commands that are processed by the robot's Motor System. The Motor System is a posegraph-based motor control system that combines procedural animation techniques with human-generated animation content to generate expressive movements of the robot's body. The Motor System ultimately controls the motors which actuate the physical body of the robot, and also controls the virtual joints of the animated robot in the robot's 3D graphical simulation environment.

The inner loop in Figure 3-2 is not a loop at all, but rather two separate streams of information coming in from the robot's sensory-motor periphery and converging towards the central cognitive mechanisms of action selection and learning. These incoming streams highlight the dual use of the robot's cognitive mechanisms to not only generate the behavior of the robot, but also to recognize the behavior of human interaction partners and infer their mental states.

My implementation computationally models simulation-theoretic mechanisms throughout several systems within the robot's overall cognitive architecture. Within the motor system, mirror-neuron inspired mechanisms are used to map and represent perceived body positions of another into the robot's own joint space to conduct action recognition. Leo reuses his belief-construction systems, and adopts the visual perspective of the human, to predict the beliefs the human is likely to hold to be true given what he or she can visually observe. Finally, within the goal-directed behavior system, where schemas relate preconditions and actions with desired outcomes and are organized to represent hierarchical tasks, motor information is used along with perceptual and other contextual clues (i.e., task knowledge) to infer the human's goals and how he or she might be trying to achieve them (i.e., plan recognition).

In the following sections, I describe the most salient components of this architecture: the mechanisms of visual perspective taking and belief inference, the mechanisms of social attention, and the unified framework for social action recognition and behavior generation.

3.2 Perspective Taking Mechanisms

I turn now to the robot’s visual perspective taking mechanisms. While others have identified that visual perspective taking coupled with spatial reasoning are critical for effective action recognition [Johnson and Demiris, 2005], understanding spatial semantics in speech [Roy et al., 2004], and human-robot collaboration on a shared task within a physical space [Trafton et al., 2005], and collaborative dialog systems have investigated the role of plan recognition in identifying and resolving misconceptions (see [Carberry, 2001] for a review), this is the first work to examine the role of perspective taking for introceptive states (e.g., beliefs and goals) in human-robot learning tasks.

3.2.1 Belief Modeling

In order to convey how the robot interprets the environment from the teacher’s perspective, I must first describe how the robot understands the world from its own perspective. This section presents a technical description of two important components of the cognitive architecture: the Perception System and the Belief System. The Perception System is responsible for extracting perceptual features from raw sensory information, while the Belief System is responsible for integrating this information into discrete object representations. The Belief System represents an integrated approach to sensor fusion, object tracking and persistence, and short-term memory.

On every time step, the robot receives a set of sensory observations $O = \{o_1, o_2, \dots, o_N\}$ from its various sensory processes. As an example, imagine that the robot receives information about buttons and their locations from an eye-mounted camera, and information about the button indicator lights from an overhead camera. On a particular time step, the robot might receive the observations $O = \{(\text{red button at position } (10,0,0)), (\text{green button at } (0,0,0)), (\text{blue button at } (-10,0,0)), (\text{light at } (10,0,0)), (\text{light at } (-10,0,0))\}$. Information is extracted from these observations by the Perception System. The Perception System consists of a set of *percepts* $P = \{p_1, p_2, \dots, p_K\}$, where each $p \in P$ is a classification function defined

such that

$$p(o) = (m, c, d), \quad (3.1)$$

where $m, c \in [0, 1]$ are match and confidence values and d is an optional derived feature value. For each observation $o_i \in O$, the Perception System produces a *percept snapshot*

$$s_i = \{(p, m, c, d) | p \in P, p(o_i) = (m, c, d), m * c > k\}, \quad (3.2)$$

where $k \in [0, 1]$ is a threshold value, typically 0.5. Returning to the example, the robot might have four percepts relevant to the buttons and their states: a location percept which extracts the position information contained in the observations, a color percept, a button shape recognition percept, and a button light recognition percept. The Perception System would produce five percept snapshots corresponding to the five sensory observations, containing entries for relevant matching percepts.

These snapshots are then clustered into discrete object representations called *beliefs* by the Belief System. This clustering is typically based on the spatial relationships between the various observations, in conjunction with other metrics of similarity. The Belief System maintains a set of beliefs B , where each belief $b \in B$ is a set mapping percepts to history functions: $b = \{(p_x, h_x), (p_y, h_y), \dots\}$. For each $(p, h) \in b$, h is a history function defined such that

$$h(t) = (m'_t, c'_t, d'_t) \quad (3.3)$$

represents the “remembered” evaluation for percept p at time t . History functions may be lossless, but they are often implemented using compression schemes such as low-pass filtering or logarithmic timescale memory structures.

A Belief System is fully described by the tuple (B, G, M, d, q, w, c) , where

- B is the current set of beliefs,
- G is a generator function map, $G : P \rightarrow \mathcal{G}$, where each $g \in \mathcal{G}$ is a history generator function where $g(m, c, d) = h$ is a history function as above,

- M is the belief merge function, where $M(b_1, b_2) = b'$ represents the “merge” of the history information contained within b_1 and b_2 ,
- $d = d_1, d_2, \dots, d_L$ is a vector of belief distance functions, $d_i : B \times B \rightarrow \mathcal{R}$,
- $q = q_1, q_2, \dots, q_L$ is a vector of indicator functions where each element q_i denotes the applicability of d_i , $q_i : B \times B \rightarrow \{0, 1\}$,
- $w = w_1, w_2, \dots, w_L$ is a vector of weights, $w_i \in \mathcal{R}$, and
- $c = c_1, c_2, \dots, c_J$ is a vector of culling functions, $c_j : B \rightarrow \{0, 1\}$.

Using the above, I define the Belief Distance Function, D , and the Belief Culling Function, C :

$$D(b_1, b_2) = \sum_{i=1}^L w_i q_i(b_1, b_2) d_i(b_1, b_2) \quad (3.4)$$

$$C(b) = \prod_{j=1}^J c_j(b) \quad (3.5)$$

The Belief System manages three key processes: creating new beliefs from incoming percept snapshots, merging these new beliefs into existing beliefs, and culling stale beliefs. For the first of these processes, I define the function N , which creates a new belief b_i from a percept snapshot s_i :

$$b_i = N(s_i) = \{(p, h) | (p, m, c, d) \in s_i, \\ g = G(p), h = g(m, c, d)\} \quad (3.6)$$

For the second process, the Belief System merges new beliefs into existing ones by clustering proximal beliefs, assumed to represent different observations of the same object. This is accomplished via bottom-up, agglomerative clustering as follows.

For a set of beliefs B :

- 1: **while** $\exists b_x, b_y \in B$ such that $D(b_x, b_y) < thresh$ **do**
- 2: find $b_1, b_2 \in B$ such that $D(b_1, b_2)$ is minimal

- 3: $B \leftarrow B \cup \{M(b_1, b_2)\} \setminus \{b_1, b_2\}$
- 4: **end while**

I label this process $merge(B)$. Finally, the Belief System culls stale beliefs by removing all beliefs from the current set for which $C(b) = 1$. In summation, then, a complete Belief System update cycle proceeds as follows:

- 1: begin with current belief set B
- 2: receive percept snapshot set S from the Perception System
- 3: create incoming belief set $B_I = \{N(s_i) | s_i \in S\}$
- 4: merge: $B \leftarrow merge(B \cup B_I)$
- 5: cull: $B \leftarrow B \setminus \{b | b \in B, C(b) = 1\}$

Returning again to the example, the Belief System might specify a number of relevant distance metrics, including a measure of Euclidean spatial distance along with a number of metrics based on symbolic feature similarity. For example, a symbolic metric might judge observations that are hand-shaped as distant from observations that are button-shaped, thus separating these observations into distinct beliefs even if they are collocated. For the example, the merge process would produce three beliefs from the original five observations: a red button in the ON state, a green button in the OFF state, and a blue button in the ON state.

The Belief System framework supports the implementation of a wide range of object tracking methods, including advanced tracking techniques such as Kalman filters [Kalman, 1960] and particle filters [Carpenter et al., 1999, Arulampalam et al., 2002]. The ability to specify multiple distance metrics allows sophisticated, general-purpose tracking methods such as these to operate side-by-side with hand-crafted rules which encode prior domain knowledge about object categories, dynamics and persistence.

3.2.2 Perspective Taking and Belief Inference

I now describe the integration of perspective taking with the mechanisms for belief modeling discussed above. Inferring the beliefs of the teacher allows the robot to build task

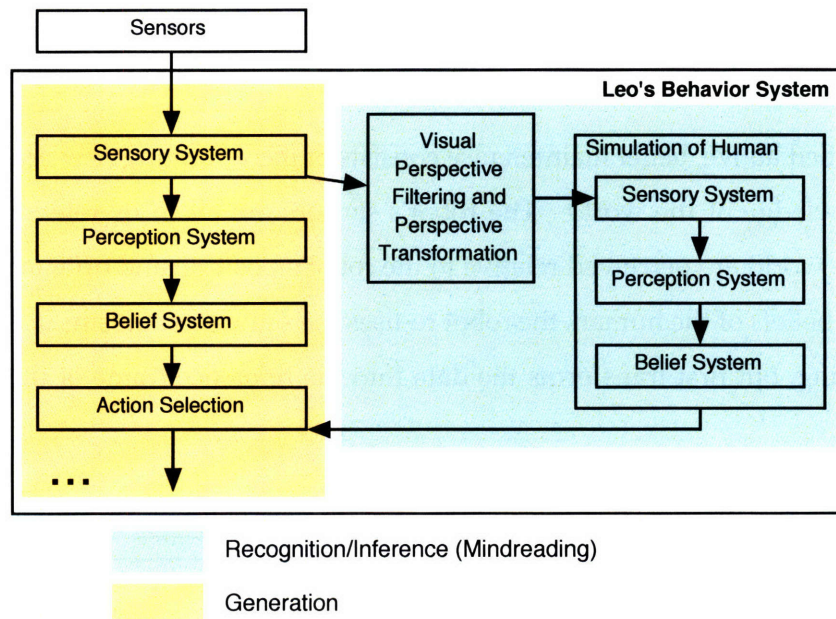


Figure 3-3: Architecture for modeling the human's beliefs re-uses the robot's own architecture for belief maintenance.

models which capture the intent behind human demonstrations.

When demonstrating a task to be learned, it is important that the context within which that demonstration is performed be the same for the teacher as it is for the learner. However, in complex and dynamic environments, it is possible for the instructor's beliefs about the context surrounding the demonstration to diverge from those of the learner. For example, a visual occlusion could block the teacher's viewpoint of a region of a shared workspace (but not that of the learner) and consequently lead to ambiguous demonstrations where the teacher does not realize that the visual information of the scene differs between them.

To address this issue, the robot must establish and maintain mutual beliefs with the human instructor about the shared context surrounding demonstrations. The robot keeps track of its own beliefs about object state using its Belief System, described above. In order to model the beliefs of the human instructor as separate and potentially different from its own, the robot re-uses the mechanism of its own Belief System. These beliefs that represent the robot's model of the human's beliefs are in the same format as its own, but

are maintained separately so the robot can compare differences between its beliefs and the human's beliefs.

As described above, belief maintenance consists of incorporating new sensor data into existing knowledge of the world. The robot's sensors are all in its reference frame, so objects in the world are perceived relative to the robot's position and orientation. In order to model the beliefs of the human, the robot re-uses the same mechanisms used for its own belief modeling, but first transforms the data into the reference frame of the human (see Fig. 3-3).

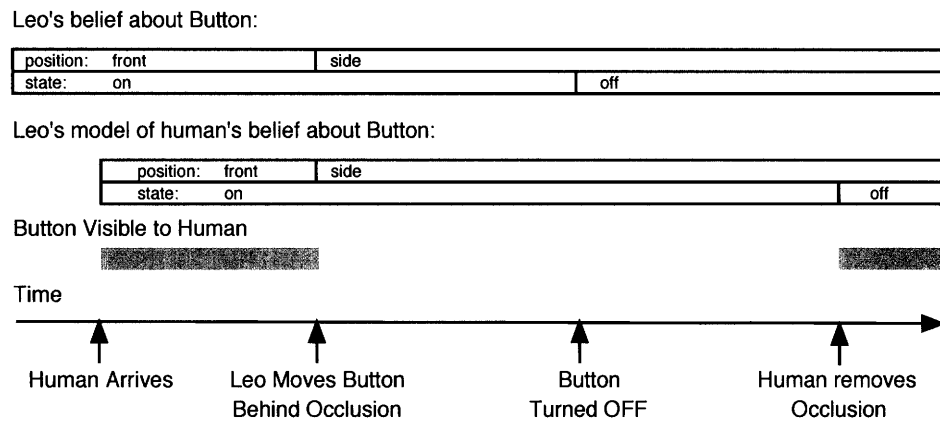


Figure 3-4: Timeline following the progress of the robot's beliefs for one button. The robot updates its belief about the button with any sensor data available - however, the robot only integrates new data into its model of the human's belief if the data is available when the human is able to perceive it.

The robot can also filter out incoming data that it believes is not perceivable to the human, thereby preventing that new data from updating the model of the human's beliefs. This incoming data might come from an object that resides outside the visual cone of the human (determined by the human's position and head orientation). Additionally, if the robot perceives an occlusion, data coming from objects on the opposite side of that occlusion from the human can be filtered before updating the robot's model of the human's beliefs.

Maintaining this parallel set of beliefs is different from simply adding metadata to the robot's original beliefs because it reuses the entire architecture which has mechanisms for

object permanence, history of properties, etc. This allows for a more sophisticated model of the human's beliefs. For instance, Fig. 3-4 shows an example where this approach keeps track of the human's incorrect beliefs about objects that have changed state while out of the human's view. This is important for establishing and maintaining mutual beliefs in time-varying situations where beliefs of individuals can diverge over time.

3.3 Social Attention Mechanisms

In this section, I introduce another important system integrated into the robot's perceptual pipeline: the mechanisms of social attention that help to guide the robot's gaze behavior, action selection, and learning. These mechanisms also help the robot to determine which objects in the environment the teacher's communicative behaviors are *about*.

Shared attention is a critical component for human-robot interaction. Gaze direction in general is an important, persistent communication device, verifying for the human partner what the robot is attending to. Additionally, the ability to share attention with a partner is a key component to social attention [Scassellati, 2001].

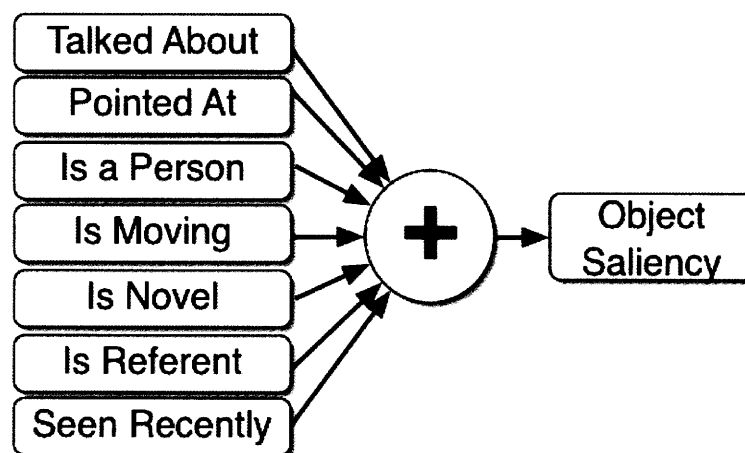


Figure 3-5: Saliency of objects and people are computed from several environmental and social factors.

Referential looking is essentially “looking where someone else is looking”. Shared attention, on the other hand, involves representing mental states of self and other [Baron-

Cohen, 1991]. To implement shared attention, the system models both the attentional focus (what is being looked at right now) and the referential focus (the shared focus that activity is *about*). The system tracks the robot's attentional focus, the human's attentional focus, and the referential focus shared by the two.



Figure 3-6: Leo in his workspace with a human partner. The human's attention on the toy influences the robot's attention as well as the referential focus.

Leo's attentional system computes the saliency (a measure of interest) for objects in the perceivable space. Overall saliency is a weighted sum of perceptual properties (proximity, color, motion, etc.), the internal state of the robot (i.e., novelty, a search target, or other goals), and social cues (if something is pointed to, looked at, talked about, or is the referential focus saliency increases). The item with the highest saliency becomes the current attentional focus of the robot, and determines the robot's gaze direction [Breazeal, 2002].

The human's attentional focus is determined by what he or she is currently looking at. Assuming that the person's head orientation is a good estimate of their gaze direction, the robot follows this gaze direction to determine which (if any) object is the attentional focus.

The mechanism by which infants track the referential focus of communication is still an open question, but a number of sources indicate that looking time is a key factor. This is discussed in studies of word learning [Baldwin and Moses, 1994, Bloom, 2002]. For

example, when a child is playing with one object and they hear an adult say “It’s a modi”, they do not attach the label to the object they happen to be looking at, but rather redirect their attention to look at what the adult is looking at, and attach the label to this object.

For the referential focus, the system tracks a *relative – looking – time* for each of the objects in the robot’s environment (relative time the object has been the attentional focus of either the human or the robot). The object with the most *relative – looking – time* is identified as the referent of the communication between the human and the robot. Fig. 3-6 shows the robot and human sharing joint visual attention. The robot has tracked the human’s head pose and pointing gesture to determine the human’s attentional focus, which in turn made this object more salient and thus the robot’s own attentional focus, thereby casting it as the referential focus of the communication.

While these attentional saliency mechanisms do not play an important role in the learning algorithms described at the end of this chapter, they are a critical part of the interactive demonstrations discussed in chapter 4. The saliency mechanisms, in conjunction with the learning algorithms, directly influence the robot’s gaze and choice of action as it responds to the cues provided by the human teachers. In particular, the robot’s gaze is a crucial feedback channel in these interactions, making the state of the robot’s cognition and learning more transparent and guidable for the human teacher.

3.4 Unified Social Activity Framework

The final important element of the self-as-simulator cognitive architecture is its unified framework for social action recognition and behavior generation. In the activity framework, the actions of the other are interpreted as learning-directed communications which constrain the exploratory behavior of the robot. As an example, imagine that a person in the environment points to a box. They might be initiating any of a number of different interactions. Do they:

- want to play an imitative game? (imitation)

- want me to engage with the box or its contents? (teaching)
- want the object in the box for themselves? (collaboration)

How I interpret the person's actions will have a profound effect upon my subsequent behavior. Gauging their intent correctly is important for the success of the interaction. I seek to build robots that can flexibly and adaptively engage in this process of interpretation, making good initial guesses and recovering gracefully from mistakes. By responding sensibly to the actions of the human, the robot will be well situated to take advantage of the natural guidance that the human can provide.

3.4.1 Action Representation: Process Models

In this section I introduce a new action representation for the robot, motivated by the need for a probabilistic, integrated framework for action recognition, production, and expectation. I see such a framework as a key piece of the simulation theoretic approach to social interaction, collaboration, and learning, where it is critical to establish a common ground between the behavior of the self and the behavior of the other.

At the heart of the action representation is the *action tuple*, a learning-friendly representation based on [Blumberg et al., 2002]. An action tuple represents a basic, temporally-extended action undertaken by the robot or another agent. An action tuple A is fully described by the tuple (t, a, o, d, g) , where

- t is the *triggering* context, an expectation about the state of the world when the action begins,
- a is the physical action,
- o is the object in the world that the physical action relates to,
- d is the *do-until* context, an expectation about the state of the world during the action and when the action finishes, and

- g indicates the agent who performs the action, important in a collaborative scenario.

In order to create a flexible framework which encompasses action recognition, production, and expectation, I embed the action tuple within a more sophisticated representation called a *process model*. A process model M is described by the tuple (S, n) , where

- $S = \{A_1, A_2, \dots, A_K\}$ is a set of action tuples, and
- n is a next-state expectation function.

The function n is defined such that $n(\cdot) = (F_n, p_n)$, where

- $F_n = \{M_1, M_2, \dots, M_J\}$ is a set of possible future process models, and
- $p_n : F_N \rightarrow [0, 1]$ is a probability distribution function, $\sum_{j=1}^J p(M_j) = 1$.

I denote the simplest process model, which specifies a single action tuple A' and which is agnostic about future states, as $M_{A'}, M_{A'} = (\{A'\}, *)$.

The process model representation is strongly related to other action representations for learning which support temporal abstraction, such as options [Sutton et al., 1999, Singh et al., 2005], and indeed parts of the process model framework have been implemented using the options formalism. The representational framework is also inspired by temporally-sophisticated action recognition techniques such as past-now-future networks [Pinhanez, 1999] and event logic [Siskind, 2001], and is clearly related to representations from the planning literature [Allen, 1994, Weld, 1994]. The process model framework is a relatively simple implementation based on these established techniques; the technical details and terminology are introduced here to support the discussion of my novel approach to exploration and social guidance.

I now provide a sketch of how process models can be used for action generation, which will be useful for my subsequent discussion of how recognized actions can be used to constrain the robot's exploration of its environment. Fig. 3-7 contains a description of the core

action selection algorithm for process models. The action selection mechanism maintains a set of core process models C , along with a working set of process models W . The core process models represent the robot's built-in, learned, or remembered models of its own behavior and the behavior of others in its environment. Each core process model is a seed for generating an extended interaction with the world. The working set represents the process models currently under consideration for describing the state of the environment and the present interaction.

Our action selection mechanism maintains a set of core process models C , along with a working set of process models W . Action selection proceeds as follows:

- 1: initialize the working set to the core set: $W \leftarrow C$
- 2: compute the probability distribution function $r : W \rightarrow [0, 1]$ based on the applicability of each $M \in W$ in the current sensorimotor context
- 3: **for** each time step **do**
- 4: compute the *biased* probability distribution function $b : W \rightarrow [0, 1]$ based on r and the system's goals and other biases
- 5: execute process model M' by selecting probabilistically over b
- 6: collect the *selection set* $Q = \{\text{the } l \text{ models } M \in W \text{ that are most consistent with } M'\}$, where l is a constant that can vary with processing constraints
- 7: initialize the set of expected future models $E = \emptyset$ and the probability distribution function $s : E \rightarrow [0, 1]$
- 8: **for** each $M \in Q$ **do**
- 9: compute $n(\cdot) = (F_n, p_n)$
- 10: $E \leftarrow E \cup F_n$
- 11: $s(f) \leftarrow r(M) \cdot p_n(f)$ for each $f \in F_n$
- 12: **end for**
- 13: renormalize s over E
- 14: update the working set: $W \leftarrow E \cup C$
- 15: receive new perceptual information from the world
- 16: update r based on s and the applicability of each $M \in W$ in the current sensorimotor context
- 17: **end for**

Figure 3-7: Action selection algorithm for process models.

On every time step, a probability distribution function r is computed over the process models in the working set which encodes the applicability of each model given the current sensorimotor context. This distribution is then biased by the system's current goals, and the resulting biased distribution is sampled to select a process model M' for execution. The system then identifies the set of models, called the *selection set*, that are still applicable given the choice of action. As an example, imagine that the robot has decided to open a box. Multiple extended processes might be consistent with this behavior: opening the box might facilitate the retrieval of a tool from the box, or it might enable the human to store a toy in the box, and so on. All of these process models would be included in the selection set, as long as they are not explicitly ruled out by the current context.

Once the selection set is assembled, it is used to generate an expectation about the next state of the world. This expectation consists of a set of possible future process models, along with a probability distribution over these models. This expectation set becomes the working set for the next time step, after any missing core models are added back in. The action selection mechanism makes sure that the working set always contains each core process model, including those that are highly unlikely given the current scenario. This guarantees that while the system devotes most of its attention to the most likely models, it can still respond to unexpected events that may necessitate a change of strategy.

3.4.2 Exploration Representation: Interpretation Modes

In this section, I introduce my approach to using the perceived actions of the other as dynamic constraints on exploration. As described in the previous section, the process model representation supports an integrated framework for action production and recognition. Using the same representation for both production and recognition greatly simplifies the problem of shaping the robot's behavior through perceived social action.

The robot's exploration mechanism maintains a set of *interpretation modes* D , where each $I \in D$ corresponds to a particular exploration strategy, generating possible behavioral responses to the perceived action of the other. Specifically, each $I \in D$ is a mapping such that $I(M) = V$, where V is a set of process model variants on M .

The robot's exploration system operates by observing the actions of the human and building process models of their behavior. When a model M' is recognized, the system selects an interpretation mode $I' \in D$, evaluates $I'(M') = V'$, and incorporates V' into the robot's current set of working models W . If pursuing one of these new process models results in a favorable outcome, the new model may be added to the robot's set of core process models for later use, exploration, and refinement.

Each interpretation mode is an implementation of a different learning question. In selecting a particular mode, the system takes a stance about how it should focus its exploration of the environment at the current time. In my implementation, each interpretation

mode produces variants of the perceived action that are focused around different, specific components of the action tuple representation. This implementation induces a hierarchy of learning questions roughly as follows:

- trigger: why?
- action: how?
- object: where?
- do-until: to what end?
- agent: who?

A related idea is the taxonomy of learning strategies that often appears in the imitation literature. Placed into a similar framework, it might look something like this:

- action: mimicry
- object: stimulus enhancement
- do-until: goal emulation

These strategies, however, are based on a direct mapping of the behavior of the other onto the behavior of the self. My productive, exploration-driven framework, while consistent with these strategies, aims to be more flexible. The strategies undertaken through my system might be better described as follows:

- trigger: context-space exploration
- action: motor-space exploration
- object: object-space exploration
- do-until: goal-space exploration

- agent: agent-space exploration

Thus, each interpretation mode focuses the robot’s exploration around a particular aspect of the human’s behavior. In this framework, the human’s actions are viewed as intentional communications aimed at drawing the robot’s attention to important features of the interactive environment. The human is not a teacher or collaborator per se, but rather a dynamic guide for the robot’s behavior.

As an example, imagine that the robot has two interpretation modes, I_{action} and I_{object} . The first focuses the robot’s exploration around the human’s physical activity, while the second focuses exploration around the object implied by that activity. Imagine that the human in the environment starts to point at a box. The system, perceiving this activity, builds a partial action tuple representation $A' = (*, pointing, box, *, agent-A)$ and embeds this action tuple within process model $M_{A'} = human-points-at-box$. Then possible results for applying the two interpretation modes might be:

$$\begin{aligned}
 I_{action}(M_{A'}) &= \{leo-points-same, leo-points-mirrored, \\
 &\quad leo-points-at-box\} \\
 I_{object}(M_{A'}) &= \{leo-opens-box, leo-lifts-box, \\
 &\quad leo-slides-box\}
 \end{aligned}$$

Thus the I_{action} interpretation mode produces process models exploring different aspects of the pointing behavior, while the I_{object} interpretation mode produces models exploring different interactions with the target of the pointing behavior.

3.5 Learning Algorithms

In this section, I describe the two algorithms that I developed within the self-as-simulator architecture for learning tasks from human teaching behavior. The first algorithm was a schema-based goal learning algorithm that was augmented to take advantage of visual

perspective cues to learn tasks both from the perspective of the robot as well as from the perspective of the human teacher. The second algorithm was a simple, Bayesian, constraint learning algorithm that was designed to take advantage of the action timing and spatial scaffolding cues that were identified through the second study. These algorithms supported both the quantitative evaluation of the utility of the identified cues, as well as a number of demonstrations of interactive learning from human teachers, as described in the following chapter.

3.5.1 Task and Goal Learning

I believe that flexible, goal-oriented, hierarchical task learning is imperative for learning in a collaborative setting from a human partner, due to the human's propensity to communicate in goal-oriented and intentional terms. Hence, I employed a hierarchical, goal-oriented task representation, wherein a task is represented by a set, S , of schema hypotheses: one primary hypothesis and n others. A schema hypothesis has x executables, E , (each either a primitive action a or another schema), a goal, G , and a tally, c , of how many seen examples have been consistent with this hypothesis.

Goals for actions and schemas are a set of y goal *beliefs* about what must hold true in order to consider this schema or action achieved. A goal belief represents a desired change during the action or schema by grouping a belief's percepts into i criteria percepts (indicating features that holds constant over the action or schema) and j expectation percepts (indicating an expected feature change). This yields straightforward goal evaluation during execution: for each goal belief, all objects with the criteria features must match the expectation features.

Schema Representation:

$$S = \{[(E_1 \dots E_x), G, c]_P, [(E_1 \dots E_x), G, c]_{1 \dots n}\}$$

$$E = a|S$$

$$G = \{B_1 \dots B_y\}$$

$$B = p_{C_1} \dots p_{C_i} \cup p_{E_1} \dots p_{E_j}$$

For the purpose of task learning, the robot can take a snapshot of the world (i.e. the state of the Belief System) at time t , $Snp(t)$, in order to later reason about world state changes. Learning is mixed-initiative such that the robot pays attention to both its own and its partner's actions during a learning episode. When the learning process begins, the robot creates a new schema representation, S , and saves belief snapshot $Snp(t_0)$. From time, t_0 , until the human indicates that the task is finished, t_{end} , if either the robot or the human completes an action, act , the robot makes an action representation, $a = [act, G]$ for S :

- 1: For action act at time t_b given last action at t_a
- 2: G = belief changes from $Snp(t_a)$ to $Snp(t_b)$
- 3: append $[act, G]$ to executables of S
- 4: $t_a = t_b$

At time t_{end} , this same process works to infer the goal for the schema, S , making the goal inference from the differences in $Snp(t_0)$ and $Snp(t_{end})$. The goal inference mechanism notes all changes that occurred over the task; however, there may still be ambiguity around which aspects of the state change are the goal (the change to an object, a class of objects, the whole world state, etc.). My approach uses hypothesis testing coupled with human interaction to disambiguate the overall task goal over a few examples.

Once the human indicates that the current task is done, S contains the representation of the seen example $[(E_1...E_x), G, 1]$. The system uses S to expand other hypotheses about the desired goal state to yield a hypothesis of all goal representations, G , consistent with the current demonstration (for details of this expansion process see [Lockerd and Breazeal, 2004]; to accommodate the tasks described here the system additionally expands hypotheses whose goal is a state change across a simple disjunction of object classes). The current best schema candidate (the primary hypothesis) is chosen through a Bayesian likelihood method: $P(h|D) \propto P(D|h)P(h)$. The data, D , is the set of all examples seen for this task. $P(D|h)$ is the percentage of the examples in which the state change seen in the example is consistent with the goal representation in h . For priors, $P(h)$, hypotheses whose goal states apply to the broadest object classes with the most specific class descriptions are preferred

(determined by number of classes and criteria/expectation features, respectively). Thus, when a task is first learned, every hypothesis schema is equally represented in the data, and the algorithm chooses the most specific schema for the next execution.

3.5.2 Perspective Taking and Task Learning

In a similar fashion, in order to model the task from the demonstrator's perspective, the robot runs a parallel copy of its task learning engine that operates on its simulated representation of the human's beliefs. In essence, this focuses the hypothesis generation mechanism on the subset of the input space that matters to the human teacher.

At the beginning of a learning episode, the robot can take a snapshot of the world in order to later reason about world state changes. The integration of perspective taking means that this snapshot can either be taken from the robot's (R) or the human's (H) belief perspective. Thus when the learning process begins, the robot creates two distinct schema representations, S_{Robot} and S_{Hum} , and saves belief snapshots $Snp(t_0, R)$ and $Snp(t_0, H)$. Learning proceeds as before, but operating on these two parallel schemas.

Once the human indicates that the current task is done, S_{Robot} and S_{Hum} both contain the representation of the seen example. Having been created from the same demonstration, the executables will be equivalent, but the goals may not be equal since they are from differing perspectives. Maintaining parallel schema representations gives the robot three options when faced with inconsistent goal hypotheses: assume that the human's schema is correct, assume that its own schema is correct, or attempt to resolve the conflicts between the schemas. The evaluation discussed in the following chapter focuses on the simplest approach: take the perspective of the teacher, and assume that their schema is correct.

3.5.3 Constraint Learning from Embodied Cues

In order to give the robot the ability to learn from the embodied cues identified in the second study, I developed a simple, Bayesian learning algorithm. This algorithm was es-

entially a fuzzy, probability-based variant of the schema learning mechanism described above. The algorithm was designed to learn rules pertaining to the color and shape of the foam blocks used in study tasks 1 and 2.

The learning algorithm maintained a set of classification functions which tracked the relative odds that the various block attributes were good or bad according to the teacher's secret constraints. In total, ten separate classification functions were used, one for each of the four possible block colors and six possible block shapes.

Each time the robot observed a salient teaching cue, these classification functions were updated using the posterior probabilities presented in the previous chapter - the odds of the target block being good or bad given the observed cue. For example, if the teacher moved a green triangle away from the student, the relative odds of *green* and *triangular* being good block attributes would decrease. Similarly, if the teacher then moved a red triangle towards the student, the odds of *red* and *triangular* being good would increase.

At the end of each interaction, the robot would attempt to identify the secret constraint being taught by the human teacher. First, the robot identified the single block attribute with the most significant good/bad probability disparity. If this attribute was a color attribute, the constraint was classified as a *color* constraint. If it was a shape attribute, the constraint was classified as a *shape* constraint. Next, all of the block attributes associated with the classified constraint type were ranked from "most good" to "most bad." Thus, if *red* was identified as the attribute with the most pronounced probability disparity, then all of the color attributes - red, green, blue, and yellow - would be ranked according to their relative probabilities of being good vs. bad.

It should be noted that this learning algorithm imposed significant structural constraints on the types of rules that the robot could learn from the interactions. For example, the robot could learn rules about colors or rules about shapes, but not rules combining both color and shape information. The space of rules that the robot considered was thus much smaller than the space of rules entertained by the human learners. However, this space was still large enough to present a significant learning challenge for the robot, with low

chance performance levels. Most importantly, this learning problem was hard enough to represent an interesting evaluation of the usefulness of the teaching cues identified in the previous chapter. The core question was: would these teaching cues be sufficient to support successful learning? The following chapter presents the evaluation that I conducted to answer this question, along with a number of demonstrations of the Leonardo robot learning interactively from natural human teaching behavior.

Chapter 4

Evaluations and Demonstrations

In this chapter, I compare the learning performance of Leonardo’s cognitive architecture against that of human learners on benchmark tasks drawn from the two studies. This comparison provides quantitative evidence for the utility of visual perspective, action timing, and spatial scaffolding as attention-direction cues for robotic learning systems. As a secondary contribution, this evaluation process supported the construction of a pair of demonstrations of the Leonardo robot making use of embodied cues to learn in novel ways from natural human teaching behavior. In the first demonstration, the Leonardo robot takes advantage of perspective taking to learn from ambiguous task demonstrations involving colorful foam blocks. The second demonstration features Leo making use of action timing and spatial scaffolding to learn secret constraints associated with a number of construction tasks, again involving foam blocks. Leonardo is the first robot to make use of visual perspective, action timing, and spatial scaffolding to learn from human teachers.

4.1 Visual Perspective Taking Demonstration and Benchmarks

The first evaluation centers around attention direction via visual perspective taking. In the demonstration, the Leonardo robot takes advantage of perspective taking to learn from

ambiguous task demonstrations involving colorful foam blocks. As an evaluation, I compared the robot's task performance against the performance of study participants on identical learning tasks.

The tasks from the first study were used to create a benchmark suite for the robot. In the robot's graphical simulation environment, the robot was presented with the same task demonstrations as were provided to the study participants (Fig. 4-1). The learning performance of the robot was analyzed in two conditions: with the perspective taking mechanisms intact, and with them disabled.

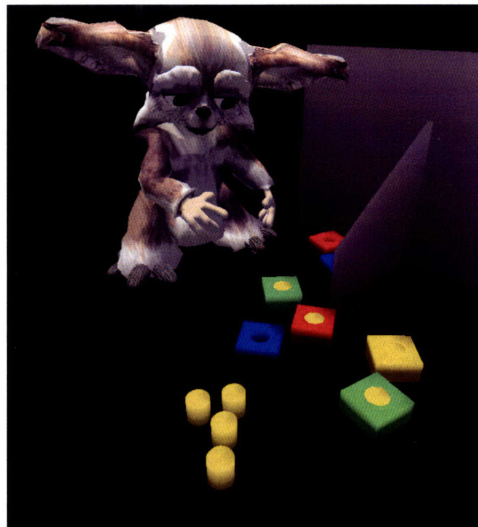


Figure 4-1: The robot was presented with similar learning tasks in a simulated environment.

The robot was instructed in real-time by a human teacher. The teacher delineated task demonstrations using verbal commands: "Leo, I can teach you to do task 1," "Task 1 is done," etc. The teacher could select and move graphical building blocks within the robot's 3D workspace via a mouse interface. This interface allowed the teacher to demonstrate a wide range of tasks involving complex block arrangements and dynamics. For the benchmark suite, the teacher followed the same task protocol that was used in the study, featuring identical block configurations and movements. For the purposes of perspective taking, the teacher's visual perspective was assumed to be that of the virtual camera through which the scene was rendered.

Table 4.1: High-likelihood hypotheses entertained by the robot at the conclusion of benchmark task demonstrations. The highest likelihood (winning) hypotheses are highlighted in bold.

Task	Condition	High-Likelihood Hypotheses
Task 1	with PT	<i>all; all but blue</i>
	without PT	<i>all but blue</i>
Task 2	with PT	<i>all red and green; shape preference</i>
	without PT	<i>shape preference</i>
Task 3 & 4	with PT	<i>rotate figure; mirror figure</i>
	without PT	<i>mirror figure</i>

Table 4.2: Hypotheses selected by study participants following task demonstrations. The most popular rules are highlighted in bold.

Task	Condition	Hypotheses Selected
Task 1	social	<i>all; number; spatial arrangement</i>
	nonsocial	<i>all but blue; spatial arrangement; all but one</i>
Task 2	social	<i>all red and green; shape preference; spatial arrangement</i>
	nonsocial	<i>shape preference; all red and green</i>
Task 3 & 4	social	<i>rotate figure; mirror figure</i>
	nonsocial	<i>mirror figure</i>

As the teacher manipulated the blocks, the robot attended to the teacher’s movements. The robot’s task learning mechanisms parsed these movements into discrete actions and assembled a schema representation for the task at hand, as detailed in previous sections. At the conclusion of each demonstration, the robot expanded and revised a set of hypotheses about the intended goal of the task. After the final demonstration, the robot was instructed to perform the task using a novel set of blocks arranged in accordance with the human study protocol. The robot’s behavior was recorded, along with all of the task hypotheses considered to be valid by the robot’s learning mechanism.

Table 4.1 shows the highest-likelihood hypotheses entertained by the robot in the various task conditions at the conclusion of the demonstrations. In the perspective taking

condition, likely hypotheses included both those constructed from the teacher’s perspective as well as those constructed from the robot’s own perspective; however, as described above, the robot preferred hypotheses constructed from the teacher’s perspective. The hypotheses favored by the learning mechanism (and thus executed by the robot) are highlighted in bold. For comparison, Table 4.2 displays the rules selected by study participants, with the most popular rules for each task highlighted in bold.

For every task and condition, the rule learned by the robot matches the most popular rule selected by the humans. This strongly suggests that the robot’s perspective taking mechanisms focus its attention on a region of the input space similar to that attended to by study participants in the presence of a human teacher. It should also be noted, as evident in the tables, that participants generally seemed to entertain a more varied set of hypotheses than the robot. In particular, participants often demonstrated rules based on spatial or numeric relationships between the objects — relationships which are not yet represented by the robot. Thus, the differences in behavior between the humans and the robot can largely be understood as a difference in the scope of the relationships considered between the objects in the example space, rather than as a difference in this underlying space. The robot’s perspective taking mechanisms are successful at bringing the agent’s focus of attention into alignment with the humans’ in the presence of a social teacher.

4.2 Emphasis Cues Benchmarks

Tasks 2a and 2b from the second study study were used to evaluate the ability of Leo’s cognitive architecture to learn from cues that human teachers naturally provide. The robot was presented with the recorded behavioral data from these tasks, and its learning performance was measured. Care was taken to make sure that the robot could only use the cues provided by the teacher and not those of the learner (and thus possibly “piggy-back” on the human learner’s success).

The robot processed the recorded study data using the same analysis pipeline as described in section 2.3.9. The foam blocks and the heads and hands of the study participants

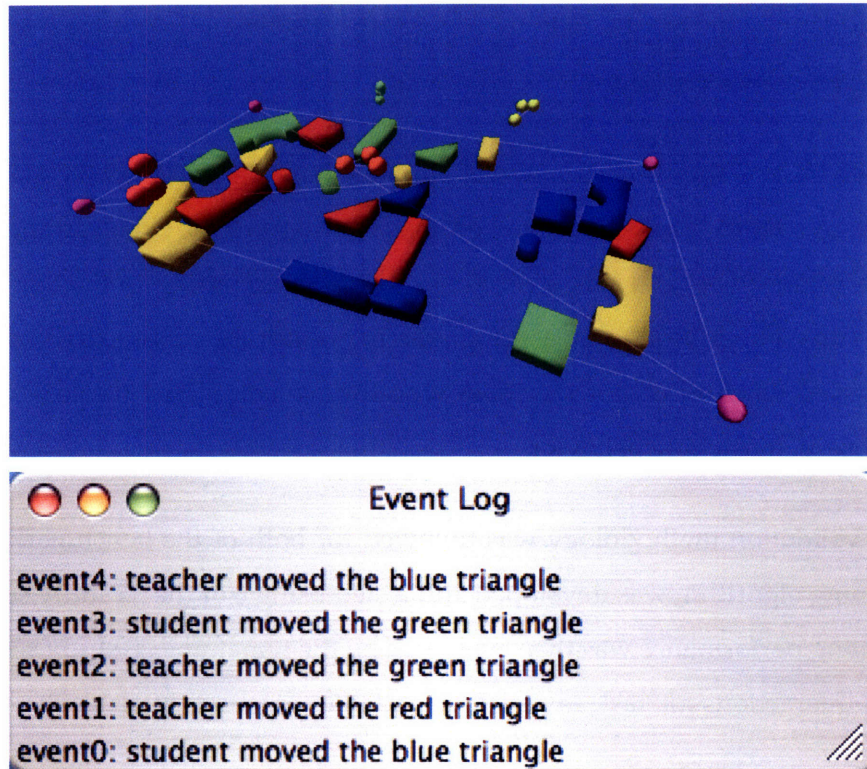


Figure 4-2: The robot was presented with the recorded study data, from which the teachers' cues were extracted.

were tracked and mapped into the same three-dimensional coordinate system. Salient events such as block movement and hand contact were identified, and agency was assigned for each event to either the teacher or the learner. From this stream of event information, two types of teaching cues were extracted: movements by the teacher towards and away from the body of the student, and movements by the teacher following movements by the student.

The robot employed a Bayesian constraint learning algorithm to learn from these cues, as described in section 3.5.3. Each time the robot observed a salient teaching cue, the algorithm updated the classification functions which tracked the relative odds of each block attribute being good or bad. At the end of each interaction, the robot identified the single block attribute with the most significant good/bad probability disparity. Based on this attribute, the secret constraint was classified as either a color-based constraint or a shape-based constraint. Next, all of the block attributes associated with the classified constraint

type were ranked from “most good” to “most bad.” The resulting ranking was recorded for each observed task interaction.

After each observed task, the robot was simulated “re-performing” the given task in a non-interactive setting. The robot followed the rules extracted by its learning algorithm, and its performance was gauged as correct or incorrect according to the teacher’s secret constraint. Thus, in addition to measuring whether or not the robot extracted the correct rules, I assessed whether or not the robot would have completed the task successfully given the observed teaching behavior.

A cross-validation methodology was followed for both of the benchmark tasks. The robot’s learning algorithm was developed and tested using 6 of the 36 study sessions. The robot’s learning performance was then evaluated on the remaining 30 study sessions, with 30 recorded interactions for Task 2a and 30 recorded interactions for Task 2b.

The performance of the human learners and the robot on the benchmark tasks is presented in tables 4.3 and 4.4, respectively. Human performance was gauged based on the guesses that the learners wrote down at the end of each task about the secret constraint. For both tasks, the secret constraint involved two rules. In Task 2a (building a sailboat), the rules were that only red blocks and blue blocks could be used in the construction of the figure (or equivalently, that green and yellow blocks could not be used). In Task 2b (building a truck), the rules were that the square blocks could not be used in the construction of the figure, while the triangular blocks had to be used. The performance of the human learners was gauged using three metrics: whether or not they correctly identified the rules as being color-based or shape-based (Rule Type Correct), whether or not they correctly specified either of the two rules (One Rule Correct), and finally, whether or not they correctly specified both rules (Both Rules Correct).

As can be seen in table 4.3, the performance of the human learners was quite high for both tasks. The rule type was identified correctly nearly 100% of the time, with both rules specified correctly 87% of the time for the Sailboat task, and at least one rule specified correctly 87% of the time for the Truck task. Interestingly, for the Truck task, both rules

Table 4.3: Performance of the human learners on study tasks 2a (Sailboat) and 2b (Truck).

Task	Rule Type Correct (color / shape)	One Rule Correct	Both Rules Correct
Sailboat	30 (100%)	27 (90%)	26 (87%)
Truck	29 (97%)	26 (87%)	4 (13%)

were specified correctly in only 4 instances (13%). In this task, the two rules were partially redundant: not being able to use the square blocks required the use of some, but not all, of the triangular blocks. Similarly, using the triangular blocks was most easily accomplished by covering some, but not all, of the square-shaped regions of the figure. Thus, the teachers could guide the learners most of the way towards the correct completion of the figure by either encouraging the use of the triangular blocks or discouraging the use of the square blocks. This may explain some of the disparity between the success rate for specifying both rules and the success rate for specifying either one of the rules.

The performance of the robot is presented in table 4.4. As described in section 3.5.3, after observing the task interaction, the robot tried to identify the most salient block attribute - the attribute with the most significant good/bad disparity. This attribute was then used to classify the constraint type as either *color* or *shape*. Then, the robot ranked all of the attributes associated with the classified constraint type, ordering them from most good to most bad. The robot's performance was assessed via a number of measures. Rule Type Correct assessed whether or not the robot classified the constraint type correctly. One Rule Correct assessed whether or not the attribute that the robot identified as the most salient was indeed one of the task rules (and whether or not that attribute was correctly identified as good vs. bad). Both Rules Correct assessed whether or not both rules were ranked correctly in the ranking of salient block attributes. For the Sailboat task, this required *red* and *blue* to be the highest-ranked attributes, and *yellow* and *green* to be the lowest-ranked attributes. For the Truck task, this required *triangular* to be the highest-ranked attribute, and *square* to be the lowest-ranked attribute. Finally, the table presents how often the robot completed the task successfully, when it was simulated re-performing the task following its observation of the interaction.

Table 4.4: Learning performance of the robot observing benchmark task interactions. Performance is measured against rules that participants were instructed to teach. Truck* adjusts for a number of instances of rule misunderstanding by the human teachers (see accompanying text).

Task	Rule Type Correct (color / shape)	One Rule Correct	Both Rules Correct	Correct Performance
Sailboat	24 (80%)	22 (73%)	21 (70%)	21 (70%)
Truck	28 (93%)	20 (67%)	10 (33%)	23 (77%)
Truck*	28 (93%)	23 (77%)	14 (47%)	23 (77%)

The robot’s performance results are very exciting. The robot was able to identify the constraint type 80% of the time in the Sailboat task and 93% of the time in the Truck task. The attribute that the robot identified as most salient correctly matched one of the task rules 73% of the time in the Sailboat task and 67% of the time in the Truck task. The robot’s attribute ranking correctly matched both task rules 70% of the time for the Sailboat task and 33% of the time for the Truck task. The robot’s re-performance of the task was successful 70% of the time for the Sailboat task and 77% of the time for the Truck task.

Additionally, in the Truck task, the robot was able to correctly match the rules identified by the human learners in a number of instances where the human teacher misunderstood the specified rule. In four cases, the teacher taught the rule “all rectangles are bad” instead of “all squares are bad.” This rule was identified by both the human learner and the robot in all four instances. The Truck* row in the table credits the robot with success in these cases, with correspondingly higher performance numbers. Essentially, this compares the robot’s performance to the rules that the learner identified, rather than the rules that the teacher was instructed to teach.

Taken together, the results suggest that the robot was able to learn quite successfully by paying attention to a few simple cues extracted from the teacher’s observed behavior. This is an exciting validation both of the robot’s learning mechanisms as well as of the usefulness of the cues themselves. These dynamic, embodied cues are not just reliable at predicting whether blocks are good and bad in isolation. They are prevalent enough and

consistent enough throughout the observed interactions to support successful learning.

4.3 Emphasis Cues Demonstration

Finally, I created a demonstration which featured Leo making use of action timing and spatial scaffolding to learn from live interactions with human teachers, in a similar, secret-constraint task domain. A mixed-reality workspace was created so that the robot and the human teacher could both interact gesturally with animated foam blocks on a virtual tabletop.



Figure 4-3: The robot took advantage of action timing and spatial scaffolding cues to learn through a live interaction with a human teacher.

The human and the robot interacted face-to-face, with a large plasma display positioned in between them with the screen facing upwards, as shown in figure 4-3. Twenty-four animated foam blocks were displayed on the screen, matching the colors and shapes of the blocks used in the study. The dimensions of the screen and the virtual blocks also closely matched the dimensions of the tabletop and blocks in the study.

The robot could manipulate the virtual blocks directly, by procedurally sliding and rotating them on the screen. To provide feedback to the human teacher, the robot gesturally tracked these procedural movements with its hand outstretched, giving the impression that the robot was levitating the blocks with its hand. The robot also followed its own movements and those of the human with its gaze, and attended to the teacher's hands and head. Additionally, the robot provided gestures of confusion and excitement at appropriate moments during the interaction, such as when the robot was interrupted by the human or when the figure was completed successfully.

The human teacher wore the same gloves and baseball cap as were worn by the participants in the study. The teacher's head and hands were tracked using the same motion-capture based object tracking pipeline. The human manipulated the virtual blocks via a custom-built gestural interface, which essentially turned the plasma display into a very large, augmented touch screen (see figure 4-4). The interface allowed the teacher to use both hands to pick up, slide, and rotate the blocks on the screen. The teacher could select a particular block by touching the screen with their fingertips, at which point the selected block would become "stuck" to their hand. In the 3D graphics environment, the selected block would follow the translation and rotation of the teacher's hand, and would continue to do so until the teacher put down the block by again touching the screen. In this manner, the teacher could manipulate any block on the screen, including blocks being moved by the robot (an interruption which would often be followed by a gesture of confusion by the robot).

The robot could be instructed to build any of a number of different figures. The silhouettes of these target figures were specified ahead of time via a simple graphical interface. During the interaction, the robot selected which blocks to use to complete the figures, incorporating the guidance of the human teacher. An example of a successfully completed figure is shown in figure 4-4, which should be recognizable as the truck figure from study Task 2b. As is suggested in the figure, the teacher has moved the square blocks away from the robot (towards the bottom of the screen), causing the robot to use the triangular and small rectangular blocks to complete the figure.

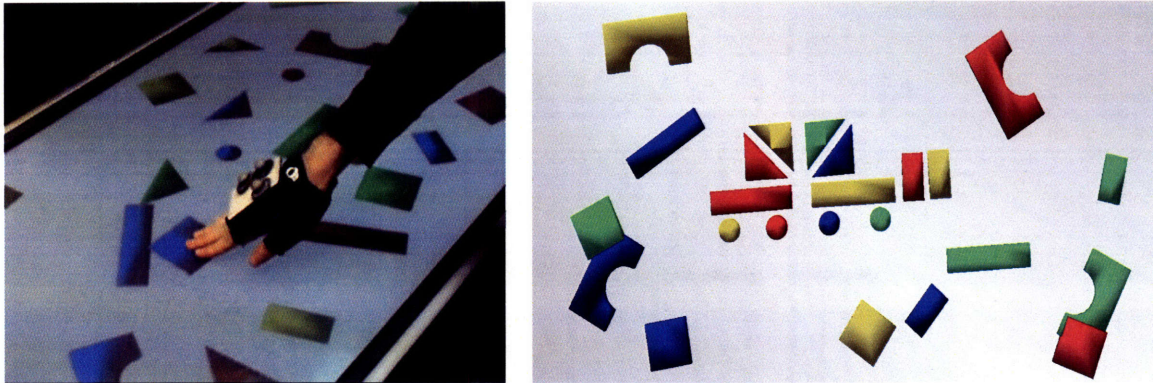


Figure 4-4: A gestural interface (left) allowed the human and the robot to interact with virtual foam blocks. A figure planning algorithm allowed the robot to use a simple spatial grammar to construct figures (right) in different ways, incorporating the guidance of the teacher.

A figure planning algorithm allowed the robot to use a simple spatial grammar to construct the target figures in different ways. This allowed for flexibility in the shapes as well as the colors of the blocks used in the figures. The spatial grammar was essentially a spatially-augmented context-free grammar. Each rule in the grammar specified how a particular figure region could be constructed using different arrangements of one or more blocks. For example, one rule in the grammar specified three alternatives for constructing square figure regions: (1) using one square block, (2) using two triangular blocks, or (3) using two small rectangular blocks. Similar rules specified the alternatives for constructing rectangular figure regions, circular regions, triangular regions, and so on.

The use of such a grammar imposed some strong restrictions on how the target figures could be constructed, and more open-ended planning techniques are certainly imaginable. However, this approach allowed the robot to be quite flexibly guided by the teacher's behavior, and allowed it to clearly demonstrate its understanding of the constraints it learned through the interaction.

For each rule in the grammar, a preference distribution specified an initial bias about which alternatives the robot should prefer. For the example of square figure regions, this distribution specified that the robot should slightly prefer the solution involving just one square block over the solutions involving two triangular or rectangular blocks. During the

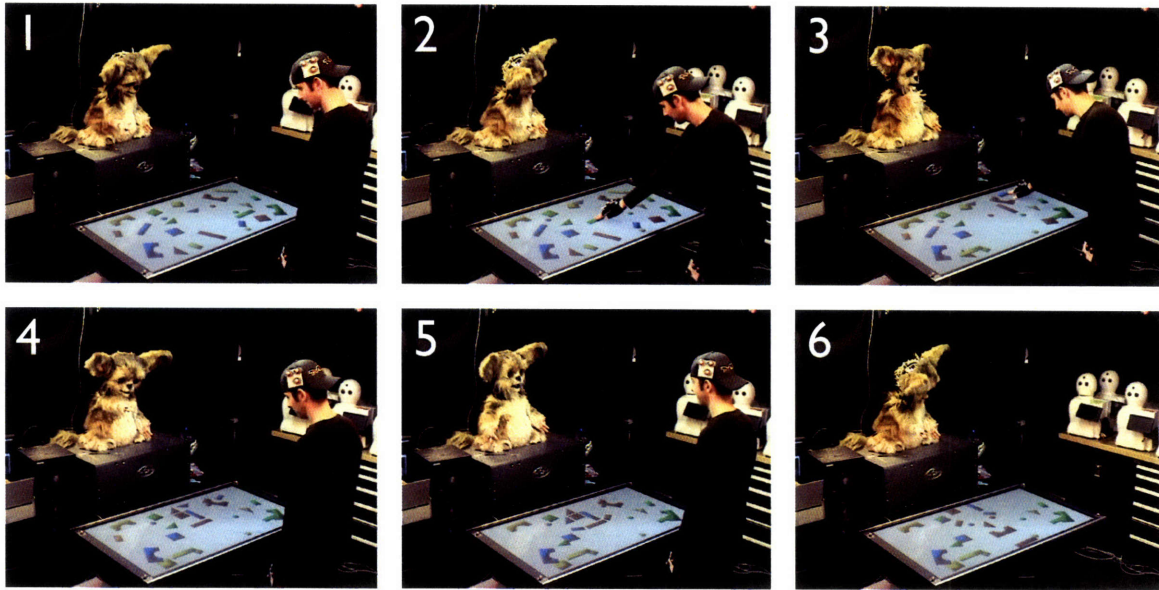


Figure 4-5: Interaction sequence between the robot and a human teacher. (1) The robot starts to construct a sailboat figure as the teacher watches. (2) The teacher interrupts an incorrect addition to the figure, and then (3) moves preferable blocks closer to the robot. (4) The teacher watches as the robot continues to construct the figure, and finally (5) completes the figure successfully, using only blue and red blocks. (6) After the teacher leaves, the robot constructs a new, smiley-face figure following the learned constraints.

interaction, the figure planning algorithm multiplied these distributions by the estimated probability of each available block being a good block, as inferred from the teacher’s embodied cues. The resulting biased probability distribution governed the robot’s choice of which block to use at each step in constructing the figure.

The preference distributions in the grammar were typically skewed only slightly, and thus could be easily overridden by the teacher’s guidance. For example, in the absence of any instruction, the robot would use single square blocks to complete square regions in the figures. However, if the robot estimated that triangular blocks were, for example, twice as likely to be good as square blocks given the teacher’s behavior, the triangular blocks would be used instead.

The robot employed the same learning algorithm as was used to learn from the recorded study data, and employed the same tracking algorithms to identify the teacher’s embodied cues. As the interaction progressed, the robot updated its estimates of whether or not

each block attribute was good or bad, and used these estimates to bias its action selection, as discussed above. Thus, if the teacher slid a green triangle closer to the robot, the odds of the robot attending to and using green blocks and triangular blocks would increase. If the teacher slid the block away from the robot, the odds of the robot ignoring such blocks would increase.

Figure 4-5 shows off an interaction sequence between the robot and a human teacher. The robot, instructed to build a sailboat figure, starts to construct the figure as the teacher watches. The teacher's goal is to guide the robot into using only blue and red blocks to construct the figure. As the interaction proceeds, the robot tries to add a green rectangle to the figure. The teacher interrupts, pulling the block away from the robot. As the robot continues to build the figure, the teacher tries to help by sliding a blue block and a red block close to the robot's side of the screen. The teacher then watches as the robot completes the figure successfully. To demonstrate that the robot has indeed learned the constraints, the teacher walks away, and instructs the robot to build a new figure. Without any intervention from the teacher, the robot successfully constructs the figure, a smiley-face, using only red and blue blocks.

Chapter 5

Conclusion

In this chapter, I present a brief summary of the contributions of my work. I then present some concluding thoughts and a discussion of plans and possibilities for future research.

5.1 Contributions

My thesis research resulted in a number of specific contributions:

I conducted two novel studies that examined the use of embodied cues in human task learning and teaching behavior. To carry out these studies, I created a novel data-gathering system for capturing teaching and learning interactions at very high spatial and temporal resolutions. Through the studies, I observed a number of salient attention-direction cues, the most promising of which were visual perspective, action timing, and spatial scaffolding. In particular, spatial scaffolding, in which teachers use their bodies to spatially structure the learning environment to direct the attention of the learner, was identified as a highly valuable cue for robotic learning systems.

I constructed a number of learning algorithms to evaluate the utility of the identified cues. I situated these learning algorithms within a large architecture for robot cognition, augmented with novel mechanisms for social attention and visual perspective taking.

I evaluated the performance of these learning algorithms in comparison to human learning data, providing quantitative evidence for the utility of the identified cues. As a secondary contribution, this evaluation process allowed me to construct a number of demonstrations of the Leonardo robot taking advantage of embodied cues to learn from natural human teaching behavior. Leonardo is the first robot to make use of visual perspective, action timing, and spatial scaffolding to learn from human teachers.

5.2 Future Work

Finally, I present some thoughts on plans and possibilities for future research stemming from this work.

5.2.1 Additional Embodied Emphasis Cues

My analysis of the second study focused on just one of the three study tasks, and on only a handful of the embodied emphasis cues observed during the task. I am interested in using the automated analysis tools that I created to generate quantitative data about some of the other cues enumerated in chapter 2, as well as about the cues observed during tasks 1 and 3. This would be interesting both to better understand the use of these cues in the learning interactions, as well as to improve the robot’s ability to learn from the recorded data.

I am interested in looking more closely at how the teachers spatially cluster the blocks in tasks 1 and 2, drawing the learner’s attention to the salient block features. I am also interested in looking at head shaking and nodding cues, hand gestures such as tapping and finger-wagging, and cues communicated through gaze and eye contact between the teacher and the learner.

Additionally, the recorded data for task 3 contains a great deal of information about how teachers and learners communicate in a sequence learning domain. Cues of interest here include how teachers guide the actions taken by the learners, and how they convey constraints in the space of available actions and identify salient perceptual events.

5.2.2 Cues for Regulating the Learning Interaction

The recorded data set also seems to contain some very interesting observations of how learners and teachers regulate the pacing and turn taking in learning interactions. A fluid continuum seems to exist between teacher-guided demonstration and learner-guided exploration, and it might be quite valuable to better understand the cues that can establish and modify where a particular interaction lies in this continuum.

Some simple cues in the data set seemed to regulate the pacing of the learner's actions and how much attention was paid by the learner to the teacher. In task 3, for example, an important cue seemed to be the proximity of the teacher's hands to the box controls. If the teachers kept their hands close to their own bodies, the learners would often proceed quite quickly with their actions. As the teachers raised their hands and moved them towards the controls, the learners actions would tend to slow, and their attention to the teacher would seem to increase. More generally, an important cue across many of the task interactions seemed to be how much the teacher was intruding into the learner's "space" versus how much they were just observing and leaving the learner alone.

5.2.3 Robots as Teachers

I used the data about the types of useful, embodied cues that human teachers naturally provide to build a robotic system that could learn in novel ways from human teaching behavior. You could also imagine using this very same data to build a better robot teacher, one that could provide appropriate, non-verbal cues and that could structure its demonstrations in a way that was highly understandable to human learners.

One could imagine a compelling demonstration of such teaching based on a "telephone" style series of interactions. The robot could be taught something new by a human expert, and then could proceed to teach that same skill or task to human novices. There is quite a bit of interesting work that could be done to frame this interaction and develop methods for evaluating its success.

5.2.4 Dexterity and Mobility

Finally, I am excited to extend the demonstrations of robot social learning presented in this thesis to a robotic platform featuring both dexterity and mobility. The ability to move around and grasp objects would allow the interaction to be fully grounded in the physical world, supporting the exploration of a wider range of embodied teaching cues. I am fascinated by the challenges presented in designing a robotic system that can learn from human teachers through interactions in real, physical space.

As robots enter the social environments of our workplaces and homes, it will be important for them to be able to learn from natural human teaching behavior. This thesis has presented some concrete steps towards this goal, by identifying a handful of simple, information-rich cues that humans naturally provide through their visual perspective, action timing, and use of space, and by demonstrating a robotic learning system that can take advantage of these cues to learn tasks through natural interactions with human teachers. By continuing to explore the information contained within our nonverbal teaching cues, we will not only make progress towards a better understanding of ourselves as social actors and learners, but also enable the creation of robots that fit more seamlessly into our lives.

Bibliography

- [Allen, 1994] Allen, J. (1994). *Readings in Planning*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.
- [Arulampalam et al., 2002] Arulampalam, M., Maskell, S., Gordon, N., Clapp, T., Sci, D., Organ, T., and Adelaide, S. (2002). A tutorial on particle filters for online nonlinear /non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188.
- [Baldwin and Moses, 1994] Baldwin, D. and Moses, J. (1994). Early understanding of referential intent and attentional focus: Evidence from language and emotion. In Lewis, C. and Mitchell, P., editors, *Children’s Early Understanding of Mind*, pages 133–156. Lawrence Erlbaum Assoc., New York.
- [Baron-Cohen, 1991] Baron-Cohen, S. (1991). Precursors to a theory of mind: Understanding attention in others. In Whiten, A., editor, *Natural Theories of Mind*, pages 233–250. Blackwell Press, Oxford, UK.
- [Barsalou et al., 2003] Barsalou, L. W., Niedenthal, P. M., Barbey, A., and Ruppert, J. (2003). Social embodiment. *The Psychology of Learning and Motivation*, 43.
- [Bloom, 2002] Bloom, P. (2002). Mindreading, communication and the learning of names for things. *Mind and Language*, 17(1 and 2):37–54.
- [Blumberg et al., 2002] Blumberg, B., Downie, M., Ivanov, Y., Berlin, M., Johnson, M., and Tomlinson, B. (2002). Integrated learning for interactive synthetic characters. In *Proceedings of the ACM SIGGRAPH*.
- [Bolt, 1980] Bolt, R. (1980). “Put-that-there”: Voice and gesture at the graphics interface. *Proceedings of the 7th annual conference on Computer graphics and interactive techniques*, pages 262–270.
- [Breazeal, 2002] Breazeal, C. (2002). *Designing Sociable Robots*. MIT Press, Cambridge, MA.
- [Breazeal et al., 2004] Breazeal, C., Hoffman, G., and Lockerd, A. (2004). Teaching and working with robots as collaboration. In *Proceedings of the AAMAS*.
- [Carberry, 2001] Carberry, S. (2001). Techniques for plan recognition. *User Modeling and User-Adapted Interaction*, 11(1-2):31–48.

- [Carpenter et al., 1999] Carpenter, J., Clifford, P., and Fearnhead, P. (1999). Improved particle filter for nonlinear problems. *Radar, Sonar and Navigation, IEE Proceedings-*, 146(1):2–7.
- [Cassell, 2000] Cassell, J. (2000). *Embodied Conversational Agents*. MIT Press.
- [Davies and Stone, 1995] Davies, M. and Stone, T. (1995). Introduction. In Davies, M. and Stone, T., editors, *Folk Psychology: The Theory of Mind Debate*. Blackwell, Cambridge.
- [Fransen et al., 2007] Fransen, B., Morariu, V., Martinson, E., Blisard, S., Marge, M., Thomas, S., Schultz, A., and Perzanowski, D. (2007). Using vision, acoustics, and natural language for disambiguation. In *Proceedings of the 2007 ACM Conference on Human-Robot Interaction (HRI '07)*.
- [Gallese and Goldman, 1998] Gallese, V. and Goldman, A. (1998). Mirror neurons and the simulation theory of mind-reading. *Trends in Cognitive Sciences*, 2(12):493–501.
- [Ghidary et al., 2002] Ghidary, S., Nakata, Y., Saito, H., Hattori, M., and Takamori, T. (2002). Multi-Modal Interaction of Human and Home Robot in the Context of Room Map Generation. *Autonomous Robots*, 13(2):169–184.
- [Gopnik et al., 2001] Gopnik, A., Sobel, D., Schulz, L., and Glymour, C. (2001). Causal learning mechanisms in very young children: Two-, three-, and four-year-olds infer causal relations from patterns of variation and covariation. *Developmental Psychology*, 37(5):620–629.
- [Gray, J. et al., 2005] Gray, J., Breazeal, C., Berlin, M., Brooks, A., and Lieberman, J. (2005). Action parsing and goal inference using self as simulator. In *14th IEEE International Workshop on Robot and Human Interactive Communication (ROMAN)*, Nashville, Tennessee. IEEE.
- [Hanna et al., 2003] Hanna, J., Tanenhaus, M., and Trueswell, J. (2003). The effects of common ground and perspective on domains of referential interpretation. *Journal of Memory and Language*, 49(1):43–61.
- [Iverson and Goldin-Meadow, 2005] Iverson, J. and Goldin-Meadow, S. (2005). Gesture Paves the Way for Language Development. *Psychological Science*, 16(5):367.
- [Johnson and Demiris, 2005] Johnson, M. and Demiris, Y. (2005). Perceptual perspective taking and action recognition. *International Journal of Advanced Robotic Systems*, 2(4):301–308.
- [Jones and Hinds, 2002] Jones, H. and Hinds, P. (2002). Extreme work teams: using swat teams as a model for coordinating distributed robots. In *Proceedings of the 2002 ACM conference on Computer supported cooperative work*, pages 372–381. ACM Press.
- [Kahn et al., 1996] Kahn, R. E., Swain, M. J., Prokopowicz, P. N., and Firby, R. J. (1996). Gesture recognition using the perseus architecture. In *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*, page 734, Washington, DC, USA. IEEE Computer Society.

- [Kalman, 1960] Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45.
- [Kendon, 1997] Kendon, A. (1997). Gesture. *Annual Review of Anthropology*, 26:109–128.
- [Keysar et al., 2000] Keysar, B., Barr, D., Balin, J., and Brauner, J. (2000). Taking perspective in conversation: the role of mutual knowledge in comprehension. *Psychological Science*, 11(1):32–8.
- [Kipp, 2004] Kipp, M. (2004). *Gesture Generation by Imitation: From Human Behavior to Computer Character Animation*. PhD thesis, Boca Raton, Florida: Dissertation.com.
- [Kobsa et al., 1986] Kobsa, A., Allgayer, J., Reddig, C., Reithinger, N., Schmauks, D., Harbusch, K., and Wahlster, W. (1986). Combining deictic gestures and natural language for referent identification. In *Proceedings of the 11th International Conference on Computational Linguistics*.
- [Langton, 2000] Langton, S. (2000). The mutual influence of gaze and head orientation in the analysis of social attention direction. *The Quarterly Journal of Experimental Psychology: Section A*, 53(3):825–845.
- [Lockerd and Breazeal, 2004] Lockerd, A. and Breazeal, C. (2004). Tutelage and socially guided robot learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- [Martin et al., 2005] Martin, B., Lozano, S., and Tversky, B. (2005). Taking the Actor’s Perspective Enhances Action Understanding and Learning. In *Proceedings of the 27th Annual meeting of the Cognitive Science Society*, Stresa, Italy.
- [McNeill, 1992] McNeill, D. (1992). *Hand and Mind: What Gestures Reveal about Thought*. University of Chicago Press.
- [McNeill, 2005] McNeill, D. (2005). *Gesture and thought*. University of Chicago Press.
- [Meltzoff and Decety, 2003] Meltzoff, A. N. and Decety, J. (2003). What imitation tells us about social cognition: a rapprochement between developmental psychology and cognitive neuroscience. *Philosophical Transactions of the Royal Society: Biological Sciences*, 358:491–500.
- [Morency et al., 2002] Morency, L.-P., Rahimi, A., Checka, N., and Darrell, T. (2002). Fast stereo-based head tracking for interactive environment. In *Int. Conference on Automatic Face and Gesture Recognition*.
- [Neal et al., 1998] Neal, J., Thielman, C., Dobes, Z., Haller, S., and Shapiro, S. (1998). Natural language with integrated deictic and graphic gestures. *Readings in Intelligent User Interfaces*, pages 38–51.
- [Nehaniv et al., 2005] Nehaniv, C., Dautenhahn, K., Kubacki, J., Haegele, M., Parlitz, C., and Alami, R. (2005). A methodological approach relating the classification of gesture

- to identification of human intent in the context of human-robot interaction. *IEEE International Workshop on Robot and Human Interactive Communication (ROMAN 2005)*, pages 371–377.
- [Nicolescu and Matarić, 2003] Nicolescu, M. N. and Matarić, M. J. (2003). Natural methods for robot task learning: Instructive demonstrations, generalization and practice. In *Proceedings of the 2nd Intl. Conf. AAMAS*, Melbourne, Australia.
- [Oviatt et al., 1997] Oviatt, S., DeAngeli, A., and Kuhn, K. (1997). Integration and synchronization of input modes during multimodal human-computer interaction. *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 415–422.
- [Pappu and Beardsley, 1998] Pappu, R. and Beardsley, P. A. (1998). A qualitative approach to classifying gaze direction. In *Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition*.
- [Perrett and Emery, 1994] Perrett, D. and Emery, N. (1994). Understanding the intentions of others from visual signals: neurophysiological evidence. *Cahiers de Psychologie Cognitive*, 13(5):683–694.
- [Pinhanez, 1999] Pinhanez, C. (1999). *Representation and Recognition of Action in Interactive Spaces*. PhD thesis, Massachusetts Institute of Technology.
- [Roy et al., 2004] Roy, D., Hsiao, K., and Mavridis, N. (2004). Mental Imagery for a Conversational Robot. *IEEE Transactions On Systems, Man, and Cybernetics—Part B: Cybernetics*, 34(3).
- [Scassellati, 2001] Scassellati, B. (2001). Foundations for a theory of mind for a humanoid robot. *Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, PhD Thesis*.
- [Schaal, 1999] Schaal, S. (1999). Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences*, 3:233242.
- [Schulz and Gopnik, 2004] Schulz, L. and Gopnik, A. (2004). Causal learning across domains. *Developmental Psychology*, 40(2):162–176.
- [Sebanz et al., 2006] Sebanz, N., Bekkering, H., and Knoblich, G. (2006). Joint action: Bodies and minds moving together. *Trends in Cognitive Sciences*, 10(2):70–76.
- [Severinson-Eklundh et al., 2003] Severinson-Eklundh, K., Green, A., and Hüttenrauch, H. (2003). Social and collaborative aspects of interaction with a service robot. *Robotics and Autonomous Systems*, 42(3-4):223–234.
- [Singer and Goldin-Meadow, 2005] Singer, M. and Goldin-Meadow, S. (2005). Children Learn When Their Teacher’s Gestures and Speech Differ. *Psychological Science*, 16(2):85.
- [Singh et al., 2005] Singh, S., Barto, A. G., and Chentanez, N. (2005). Intrinsically motivated reinforcement learning. In *Proceedings of Advances in Neural Information Processing Systems 17 (NIPS)*.

- [Siskind, 2001] Siskind, J. (2001). Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic. *Journal of Artificial Intelligence Research*, 15(1):31–90.
- [Smith et al., 2007] Smith, L., Maouene, J., and Hidaka, S. (2007). The body and children’s word learning. In Plumert, J. M. and Spencer, J., editors, *Emerging landscapes of mind: Mapping the nature of change in spatial cognitive development*. Oxford University Press, New York.
- [Stiefelbogen, 2002] Stiefelbogen, R. (2002). Tracking focus of attention in meetings. In *Proceedings of the Fourth IEEE International Conference on Multimodal Interfaces (ICMI ’02)*.
- [Sutton et al., 1999] Sutton, R., 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.
- [Trafton et al., 2005] Trafton, J. G., Cassimatis, N. L., Bugajska, M. D., Brock, D. P., Mintz, F. E., and Schultz, A. C. (2005). Enabling effective human-robot interaction using perspective-taking in robots. *IEEE Transactions on Systems, Man, and Cybernetics*, 35(4):460–470.
- [Weld, 1994] Weld, D. (1994). An Introduction to Least Commitment Planning. *AI Magazine*, 15(4):27–61.
- [Wilson and Bobick, 1999] Wilson, A. D. and Bobick, A. F. (1999). Parametric hidden markov models for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9).
- [Wimmer and Perner, 1983] Wimmer, H. and Perner, J. (1983). Beliefs about beliefs: Representation and constraining function on wrong beliefs in young children’s understanding of deception. *Cognition*, 13:103–128.
- [Zukow-Goldring, 2004] Zukow-Goldring, P. (2004). Caregivers and the education of the mirror system. In *Proceedings of the International Conference on Development and Learning*.