Recognizing Engagement Behaviors in Human-Robot Interaction

By
Brett Ponsler

A Thesis

Submitted to the faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Master of Science

in

Computer Science

January 2011

APPROVED:

Professor Charles Rich, Thesis Advisor

Professor Joseph Beck, Thesis Reader

Professor Michael Gennert, Head of Department

Abstract

Based on analysis of human-human interactions, we have developed an initial model of engagement for human-robot interaction which includes the concept of connection events, consisting of: directed gaze, mutual facial gaze, conversational adjacency pairs, and backchannels. We implemented the model in the open source Robot Operating System and conducted a human-robot interaction experiment to evaluate it.

Acknowledgements

First and foremost I would like to thank Professors Chuck Rich and Candy Sidner. I thank you both for the opportunity to do research with you, and to learn from you. In the short time that I have known you, I have learned more about conducting research, computer science, and human interaction than I could ever express and for that I will be forever thankful. You have helped turn me into the computer scientist, and person, that I am today.

I would also like to thank Professor Beck without whom I would not have significant results. Thank you for your comments and suggestions on my thesis as well as your invaluable statistical advice.

Aaron, thank you for all of your help debugging and testing, as well as for helping to trick Melvin into making a tangram. It has been a pleasure working with you, and I wish you the best of luck as you pursue your PhD.

Thanks to all of the members of the Interaction Lab, especially Paulo, Jia, Will, Ben, and Tonje. Thanks for listening to Melvin talk for hours on end without complaining, and for all the help you provided whenever asked.

I would also like to thank all of the participants who played tangrams with Melvin, and I hope you had fun.

Finally, thanks to the WPI Computer Science Department professors and secretaries for all that you have contributed to my studies over the years.

This work is supported in part by the National Science Foundation under awards IIS-0811942 and IIS-1012083.

Table of Contents

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 Related work . . . 2

2 Connection events in human-human interaction **6**
- 2.1 Directed gaze . . . 7
- 2.2 Mutual facial gaze . . . 9
- 2.3 Adjacency pair . . . 9
- 2.4 Backchannels . . . 10
- 2.5 Summary statistics . . . 11

3 Human-robot architecture **12**
- 3.1 Human-robot setting . . . 12
- 3.2 Information flow . . . 13
- 3.3 Engagement recognition module . . . 15

4 Human-robot implementation **17**
- 4.1 Recognition . . . 17
- 4.2 Testing module . . . 21
- 4.3 ROS details . . . 22

5 Human-robot demonstration **25**
- 5.1 Humanoid robot . . . 25
- 5.2 Task description . . . 26

6 Evaluation **29**
- 6.1 Experimental design . . . 29
- 6.2 Results . . . 31
- 6.3 Discussion . . . 31

7 Conclusion **35**
- 7.1 Future work . . . 35

A Tangram questionnaire **38**
- A.1 Likert scale questions . . . 38
- A.2 Personal experience questions . . . 39

B Pre-study tangram explanation **40**

C Transcript of a tangram game **42**

D Questionnaire results **44**

List of Tables

2.1 Summary statistics for human engagement study 11

4.1 Maximum delay timeouts for each connection event. 21

6.1 Relevant questionnaire results 31

D.1 Questionnaire results using 2 tail, unpaired t-test 44

List of Figures

2.1 Two camera views of participants in human engagement study (during directed gaze event). . . . 6
2.2 Detailed coding of the human interaction videotapes. . . . 7
2.3 Time line for directed gaze (numbers for reference in text). . . . 8
2.4 Time line for mutual facial gaze (numbers for reference in text). . . . 9
2.5 Time line for adjacency pair (numbers for reference in text). . . . 9
2.6 Time line for backchannels (numbers for reference in text). . . . 10

3.1 Internal architecture of the engagement recognition module. . . . 12
3.2 Internal architecture of the engagement recognition module. . . . 13
3.3 Internal architecture of the engagement recognition module. . . . 15

4.1 State machine for recognizing directed gaze. . . . 18
4.2 State machine for recognizing mutual facial gaze. . . . 19
4.3 State machine for recognizing adjacency pair. . . . 19
4.4 State machine for recognizing backchannel. . . . 20
4.5 The engagement stack. . . . 22
4.6 Example ROS message pertaining to human directed gaze. . . . 23
4.7 Example ROS service pertaining to a robot directed gaze goal. . . . 23

5.1 Melvin and author making a tangram. . . . 25
5.2 Start and completion of the anchor tangram puzzle. . . . 27

1 Introduction

Engagement is "the process by which two (or more) participants establish, maintain and end their perceived connection during interactions they jointly undertake" [23]. Engagement is a fundamental process that governs human interaction and consists of three phases: initiation, maintenance, and termination. For example, humans can initiate engagement by saying "Hello," maintain engagement by using eye contact, and terminate engagement by saying "Goodbye."

Through analysis of human-human interactions, we identified four types of *connection events* that contribute to engagement: directed gaze, mutual facial gaze, adjacency pairs, and backchannels. We also developed a computational model for automatically recognizing these events during human-robot interaction. We implemented this model as a reusable Robot Operating System (ROS) [20] module which is available under an open source license at `ros.org`.

This thesis focuses on engagement *recognition*, while Holroyd's coordinated thesis [11] focuses on engagement *generation*. We have conducted a joint human-robot study to evaluate both the recognition and generation models.

Our methodology was first to analyze human interactions from which a model of human engagement behavior was extracted. Then we implemented the model in a reusable robotic software system able to successfully interact with an untrained individual.

Engagement is a multimodal process involving speech, body position and orientation, and gesture. Thus in this paper, we will refer to *actions* as a behavior taken to achieve a specific task. A *gesture* is a specific type of action which is intended to symbolize something. Lastly, *behaviors* include both actions and gestures.

1.1 Motivation

We believe that engagement is a fundamental process which governs all human interaction and has common features across a very wide range of interaction circumstances. This implies that it is crucial that humanoid robots be able to recognize and generate engagement behaviors if we hope to develop robots able to interact effectively with untrained humans.

This argument motivates the two main goals of this work, which are to develop a computational model for recognizing engagement behaviors, and to implement the model such that it can be reused across different robots and applications. There is no reason that every project

should need to re-implement the engagement process. Along with the creators of ROS [20], and others, we share the vision of increasing code reuse in the robotics research and development community.

1.2 Related work

Researchers have approached the study of engagement in several different manners. While some have chosen to study engagement as a whole, others have studied the behaviors which comprise the three phases of engagement, i.e., initiation, maintenance and termination. Research has been done regarding these behaviors with respect to interactions with robots and virtual agents in both controlled and uncontrolled environments. Our work investigates engagement maintenance behaviors, namely: directed gaze, mutual facial gaze, adjacency pairs, and backchannels, which have been well researched by others. Our contribution is the unifying concept of connection events, and the development of a reusable engagement software module. The research most related to our own work is discussed below.

1.2.1 Engagement and collaboration

Collaboration is a process by which two or more participants coordinate their actions toward achieving shared goals [10]. Engagement supports collaboration in that a collaborator depends on engagement behaviors, for example, to determine when to continue or end the collaboration. Collaboration also affects engagement, e.g., when involved in a complicated task, it is an acceptable engagement behavior to make less eye contact than otherwise.

Engagement and collaboration between a human and a robot has been previously investigated by Sidner *et al.* [23, 22]. They performed a study wherein a human interacted with a robot that was able to track faces (mutual facial gaze and directed gaze) while explaining a collaborative task. They found that the number of times the human looked back at the robot was significantly greater when the robot tracked the human's face and performed pointing gestures versus when it did not. These results show the effect engagement has on an interaction. The robot in this research used rudimentary initiation and termination behaviors and was only able to recognize the human's nodding as opposed to having a more complete model of behaviors that contribute to engagement, such as what we have developed.

Bohus and Horvitz [2, 3] investigated how to automatically learn engagement behaviors in multi-party situations in which a virtual agent is able to initiate, terminate, or suspend en-

gagement with multiple humans in real time. They have developed predictive models from observations, and implemented the models in a system which uses machine learning to adapt to behaviors of participants, as well as to the environment. They have also shown experimental results which validate the approach with respect to learning to detect engagement intentions. Furthermore, they have created components able to sense engagement state, actions, and intentions, make engagement decisions, and render behaviors in a virtual agent. To evaluate these components, they conducted a preliminary observational study using an on-screen virtual agent which interacts with participants to play a trivia game. This system was able to track the engagement of multiple participants in an uncontrolled environment.

Bohus and Horvitz have not included directed gaze as an engagement behavior in their work, possibly because it is not as natural with a virtual agent, but they have performed experiments in uncontrolled environments, and used machine learning in their research both of which we have not yet done. We share a similar theoretical framework, but differ in the fact that we are modeling engagement for use with a humanoid robot rather than a virtual agent.

1.2.2 Engagement initiation behaviors

Researchers are also concerned with behaviors that initiate interactions. These behaviors consists of greetings, waving at someone, or saying "hi".

Participant roles and gaze cues

Mutlu *et al.* [18] are interested in the role that gaze cues play in forming participant roles during an interaction, specifically how a humanoid robot can use different gaze behaviors to establish the participant's role in the interaction. They have made use of three participant roles: *addressee*, *bystander*, and *overhearer*. An addressee is addressed by the speaker during the interaction, and is also a contributing member of the interaction. In an interaction where a bystander is present, the robot acknowledges the presence of a bystander; however, it does not directly speak to him. An overhearer is a participant that is not addressed, or acknowledged by the robot. The authors conducted a three-arm experiment using different gaze behaviors in an interaction with two humans and one robot where the robot treated participants as different roles depending on the condition. This experiment showed that the gaze behaviors altered the subject's participation during the interaction with respect to their attention to the conversation, feelings of belonging to the group, and feelings toward the robot.

Despite the fact that Mutlu *et al.* focus on these aspects of engagement, their research has

no concept of connection events, and focused mainly on the initiation phase of engagement while this work focuses on the maintenance phase.

1.2.3 Engagement maintenance behaviors

Researchers have also investigated behaviors that maintain interactions due to the fact that the maintenance phase is the longest phase of engagement. This phase consists of behaviors such as head nods, eye contact, and utterances.

Humanoid museum tour guide

Faber *et al.* [8] have developed a system for use in a mobile full-body humanoid robot which functions autonomously as a museum tour guide. This robot interacts with humans using gesture, and speech, and is able to focus on different people involved in the interaction. The robot has a probabilistic belief about the participants which is based on face tracking and head-pose estimation. The robot can also recognize the human's gestures such as nodding, shaking, waving, and pointing using hidden Markov models to calculate the trajectories of the hands. The participants can speak to the robot using a small vocabulary. Using several factors, such as recent speech, distance from the robot, and relative body angle, the robot can compute an importance value for each participant which is used to determine where to focus. The robot also performs arm gestures to greet the participant, ask the participant to come closer, draw the participant's attention toward a specific object (directed gaze), or to emphasize an utterance. The robot displays emotion through its use of facial expressions and emotional speech. The robot is able to do all of this while walking and avoiding obstacles.

Faber *et al.* have a probabilistic model of engagement, but they do not have a notion of connection events which this work develops. Humans are also able to interact with the robot using a small vocabulary, while we do not provide any vocabulary for participants. Their robot is also able to present emotions in both speech and facial expression, while our model is not concerned with the meaning of utterances–only the timing. Lastly, their work also differs from our work due to our focus on developing reusable engagement modules to allow other researchers to make use of our work.

Direction of attention

Peters [19] investigated two areas of research: direction of attention, and theory of mind. Direction of attention is important to collaboration because humans use their eyes, and gaze, for

social purposes, e.g., to convey emotions, or relay information. Theory of mind investigates how mental states, such as, beliefs, goals, and desires, can be attributed to oneself and to others. This models information such as if the human has seen the agent or seen where the agent is looking, as well as the human's interest level. We share a similar theoretical framework with Peters, but differ in the fact that we are modeling engagement for use with a humanoid robot rather than a virtual agent.

Spatial model of engagement

Michalowski *et al.* [16] have developed a spatial model of engagement for a virtual agent in the form of a robotic receptionist with the goal of engaging visitors and maintaining their interest. Their model of engagement includes four categories for engagement: present, attending, engaged, and interacting. These categories model the distance of people to the agent, i.e., near and far, as well as their level of interaction with the agent, i.e., observing or interacting. We share a similar theoretical framework with Michalowski *et al.*, but differ in the fact that we are modeling engagement for use with a humanoid robot rather than a virtual agent.

1.2.4 Other architectures

Flippo *et al.* [9] have developed a reusable multimodal framework to aid development of applications. This framework provides developers with an interface to multimodal fusion techniques. The framework can be used to fuse the data from eye tracking, speech, and other input devices in order to manage the overall task. Flippo *et al.* share similar concerns of developing extensible, reusable software, and fusing verbal and nonverbal behaviors. They have focused on the input data fusion problem for multimodal interfaces, while we focus on the concept of engagement and connection events for robots.

2 Connection events in human-human interaction

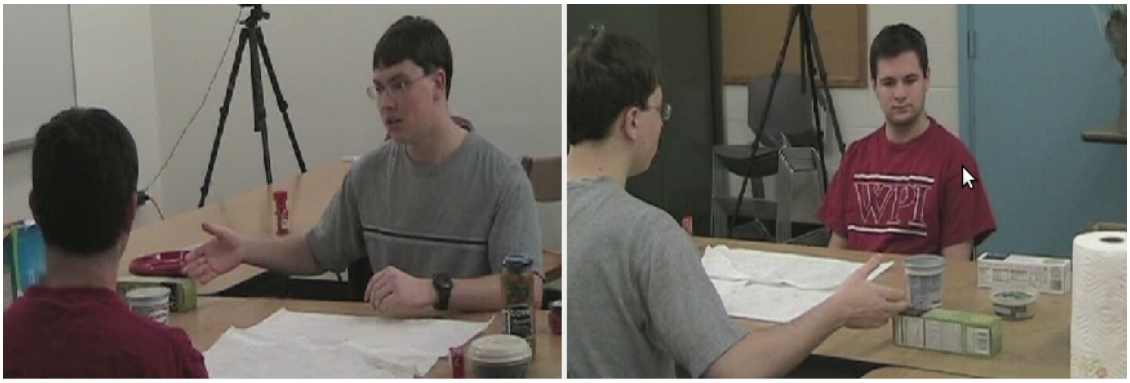


Figure 2.1: Two camera views of participants in human engagement study (during directed gaze event).

We have conducted an observational study of human engagement behavior [12] in which two participants sat across a table from each other to prepare canapés together (see Fig. 2.1). Each of the four sessions involved an experimenter and two study participants and lasted about 15-20 minutes. During the first portion of the experiment, the experimenter taught the first study participant how to prepare various types of canapés using crackers, toppings, and spreads that were arranged on the table. For the second portion, the experimenter left the room and the first participant taught the second participant how to prepare the various canapés that he or she learned about in the first portion. There were eight study participants in total, six males and two females, who were all college students at Worcester Polytechnic Institute (WPI). Each session was videotaped using two cameras.

We then analyzed the videos with respect to the engagement maintenance behaviors. The behaviors for initiating engagement (meeting, greeting, etc) and terminating engagement (standing up from the table, leaving the room, etc) were recorded, but not analyzed and will be useful for further analysis.

During the periods of interest, we coded where each person was looking (at the other person's face, at specific objects, or "away"), when each person pointed at objects, and the beginning and end of each person's speech. Each tick mark at the bottom of Fig. 2.2 indicates the start or end of an event for either the teacher or the participant. Based on this analysis and other human

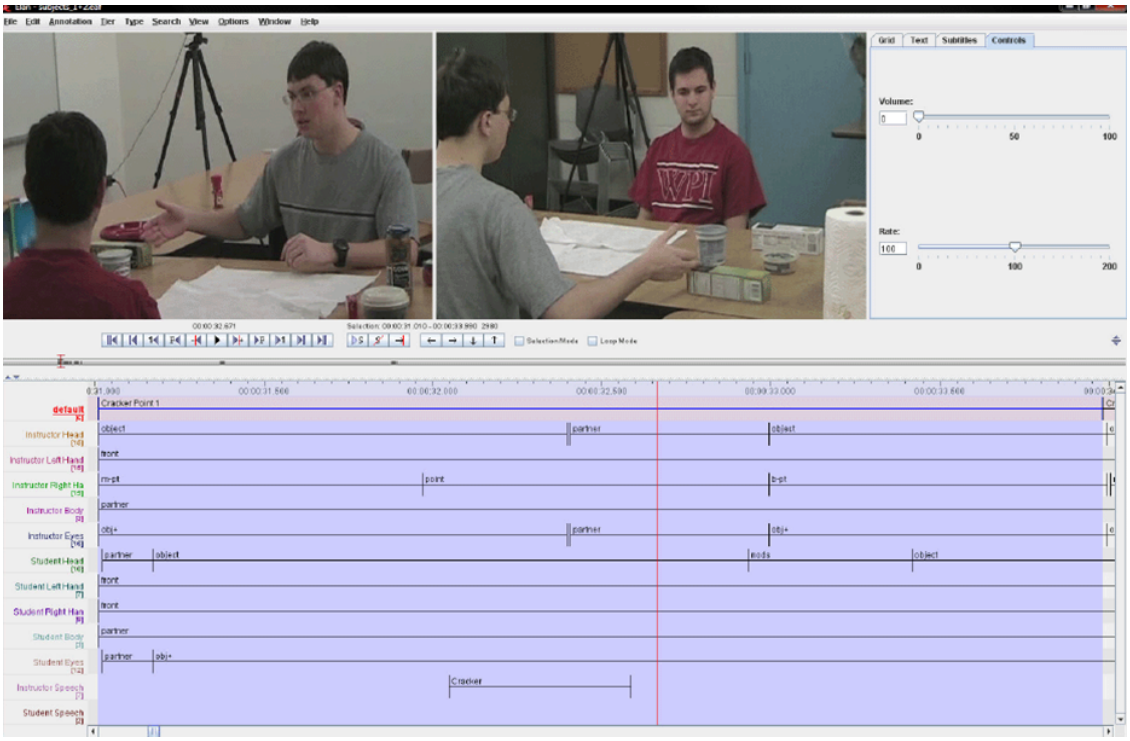


Figure 2.2: Detailed coding of the human interaction videotapes.

interaction research [1, 14, 7, 18], we have identified four types of, what we call, *connection events*. Our hypothesis is that these events, occurring at some minimum frequency, are the process mechanism for maintaining engagement.

Abstracting from all our coded data, figures 2.3 through 2.6 show the time lines for each type of connection event we have analyzed. Dotted lines indicate optional behaviors. Each connection event has an *initiator*, and a *responder*, where the initiator performs an action with the intention of eliciting a behavior from the responder. Note that each event type can be either human- or robot-initiated and that gesture and speech often overlap.

Connection events, excepting backchannels, can either *succeed*, or *fail*. Success occurs when the responder performs his intended response, and failure occurs when the responder does **not** respond. Failures are common for various reasons not considered in this work, e.g., a task in which the responder is distracted, or does not wish to respond.

2.1 Directed gaze

In directed gaze [14], one the initiator looks and optionally points at some object or group of objects in the immediate environment, following which the responder looks at the same

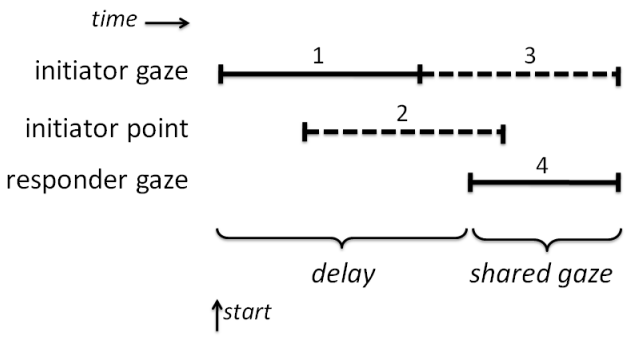


Figure 2.3: Time line for directed gaze (numbers for reference in text).

object(s). We hypothesize that the initiator intends to bring the indicated object(s) to the responder's attention, i.e., to make the object(s) more salient in the interaction. This event is often synchronized with the initiator referring to the object(s) in speech, as in "now spread the *cream cheese* on the cracker." By turning his gaze where directed, the responder intends to be cooperative and thereby signals his desire to continue the interaction (maintain engagement).

In more detail (see Fig. 2.3), notice first that the pointing behavior (2), if it is present, begins after the initiator starts to look (1) at the indicated object(s). This is likely because it is hard to accurately point at something without looking to see where it is located–although, one can easily imagine an exception to this situation, and other situations, where a person can point without looking. Furthermore, we observed several different configurations of the hand in pointing, such as extended first finger, open hand (palm up or palm down – see Fig. 2.1), and a circular waving motion (typically over a group of objects). An interesting topic for future study (that will contribute to robot generation of these behaviors) is to determine which of these configurations are individual differences and which serve different communicative functions. After some delay, the responder looks at the indicated object(s) (4). The initiator usually maintains the pointing (1), if it is present, at least until the responder starts looking at the indicated object(s). However, the initiator may stop looking at the indicated object(s) (2) before the responder starts looking (4), especially when there is pointing. This is often because the initiator looks at the responder's face, assumedly to check whether the responder has directed his gaze yet (such a moment is captured in Fig. 2.1).

Finally, there may be a period of shared gaze, i.e., a period when both the initiator (3) and responder (4) are looking at the same object(s). Shared gaze has been documented [4] as an important component of human interaction.

2.2 Mutual facial gaze

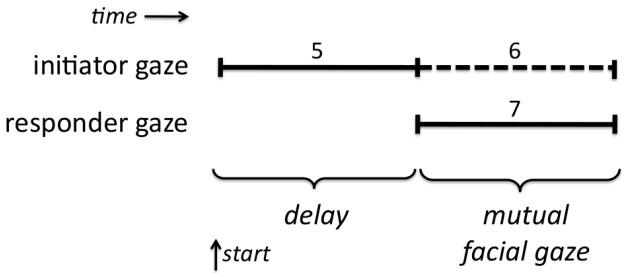


Figure 2.4: Time line for mutual facial gaze (numbers for reference in text).

Mutual facial gaze [1] has a time line (see Fig. 2.4) similar to directed gaze, but simpler, since it does not involve pointing. The event starts when the initiator looks at the responder's face (5). After a delay, the responder looks at the initiator's face, which starts the period of mutual facial gaze (6,7). Notice that the delay can be zero, which occurs when both parties simultaneously look at each other. The intentions underlying mutual facial gaze are less clear than those for directed gaze. We hypothesize that both the initiator and responder in mutual facial gaze engage in this behavior because they intend to maintain the engagement process. Mutual facial gaze does however have other interaction functions. For example, it is typical to establish mutual facial gaze at the end of a speaking turn.

Finally, what we are calling mutual facial gaze is often referred to informally as "making eye contact." This latter term is a bit misleading since people do not normally stare continuously into each other's eyes, but rather their gaze roams around the other person's face, coming back to the eyes from time to time.

2.3 Adjacency pair

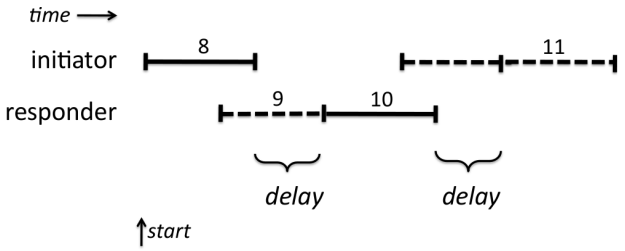


Figure 2.5: Time line for adjacency pair (numbers for reference in text).

In linguistics, an adjacency pair [5] consists of two utterances by two speakers, with minimal overlap or gap between them, such that the first utterance provokes the second utterance. A question-answer pair is a classic example of an adjacency pair. We generalize this concept slightly to include both verbal (utterances) and non-verbal communication acts. So for example, a nod could be the answer to a question, instead of a spoken "yes." Adjacency pairs, of course, often overlap with the gestural connection events, such as directed gaze and mutual facial gaze.

The simple time line for an adjacency pair is shown in Fig. 2.5. First the initiator communicates what is called the first turn (8). Then there is a delay, which could be zero if the responder starts talking before the the initiator finishes (9). Then the responder communicates what is called the second turn (9,10). In some conversational circumstances, this could also be followed by a third turn (11) in which the initiator, for example, repairs the responder's misunderstanding of his original communication.

2.4 Backchannels

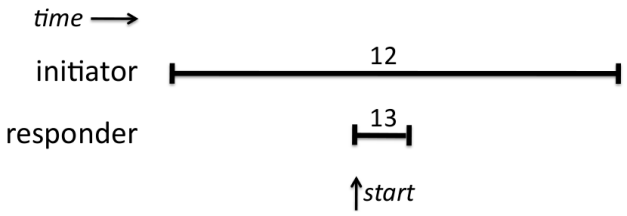


Figure 2.6: Time line for backchannels (numbers for reference in text).

A backchannel [5] is an event (see Fig. 2.6) in which one party (the responder) directs a brief verbal or gestural communication (13) back to the initiator during the primary communication (12) from the initiator to the responder. Typical examples of backchannels are nods and/or saying "uh, huh." Backchannels are typically used to communicate the responder's comprehension of the initiator's communication (or lack thereof, e.g., a quizzical facial expression) and/or desire for the initiator to continue. Unlike the other three connection event types, the start of a backchannel event is defined as the start of the responder's behavior and this event has no concept of delay.

2.5 Summary statistics

Table 2.1 shows the statistics gathered from a detailed quantitative analysis of approximately nine minutes of engagement maintenance time in our observational study. The time between connection events is defined as the time between the *start* of successive events, which properly models overlapping events. We hypothesize that the mean time between connection events (MTBCE) captures something of what is informally called the "pace" of an interaction [6]:

$$pace \propto \frac{1}{MTBCE}$$

In other words, the faster the pace, the less the time between connection events. Furthermore, our initial implementation of an engagement recognition module (see Section 3) calculates the MTBCE on a sliding window and considers an increase as evidence for the weakening of engagement.

Two surprising observations in Table 2.1 are the relatively large proportion of failed mutual facial gaze (13/24) and adjacency pair (15/45) events and the 70 second maximum time between connection events. Since we do not believe that engagement was seriously breaking down anywhere during the middle of our sessions, we take these observations as an indication of missing factors in our model of engagement. In fact, reviewing the specific time intervals involved, what we found was that in each case the (non-)responder was busy with a detailed task on the table in front of him, e.g., spreading a condiment on a cracker.

Table 2.1: Summary statistics for human engagement study

		count	delay (sec)		
			min	mean	max
directed gaze	succeed	13	0	0.3	2.0
	fail	1	1.5	1.5	1.5
mutual facial gaze	succeed	11	0	0.7	1.5
	fail	13	0.3	0.6	1.8
adjacency pair	succeed	30	0	0.4	1.1
	fail	14	0.1	1.2	7.4
backchannel		15	n/a	n/a	n/a
mean time between connection events (MTBCE) = 5.7 sec max time between connection events = 70 sec					

3 Human-robot architecture

In general in software development, the key to making a reusable component is careful attention to the setting in which it will be used and the "division of labor" between the component and the rest of the computational environment in which it is embedded.

3.1 Human-robot setting

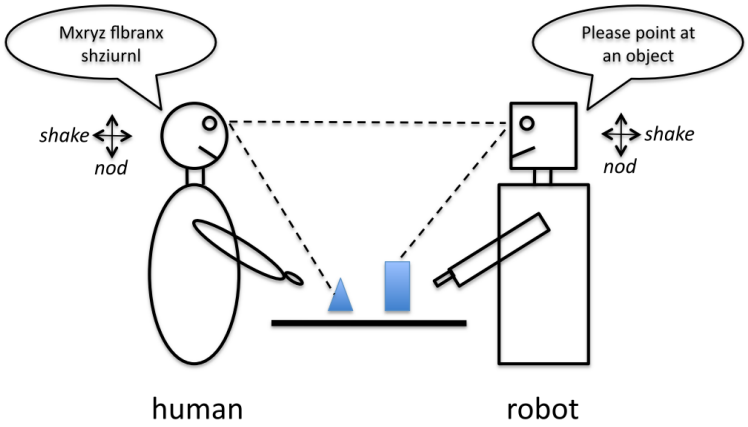


Figure 3.1: Internal architecture of the engagement recognition module.

Fig. 3.1 shows the setting of our current architecture and implementation, which mirrors the setting of the human engagement study, namely a human and a humanoid robot with a table of objects between them. Either the robot or the human can be the initiator (or responder) in the connection event time lines shown in the previous section. Like the engagement maintenance part of the human study, mobility is not part of this setting. Unlike the human study, we are not dealing here with manipulation of the objects or changes in stance (e.g., turning the body to point to or manipulate objects on the side part of the table). Both the human and the robot can perform the following behaviors and observe them in the other:

1. Look at the other's face, objects on the table or "away",
2. Point at objects on the table,
3. Nod the head (up and down), and
4. Shake the head (side to side).

The robot can generate speech that is understood by the human. However, since our demonstration system (see Section 5) does not include natural language understanding, the robot can only detect the beginning and end of the human's speech.

3.2 Information flow

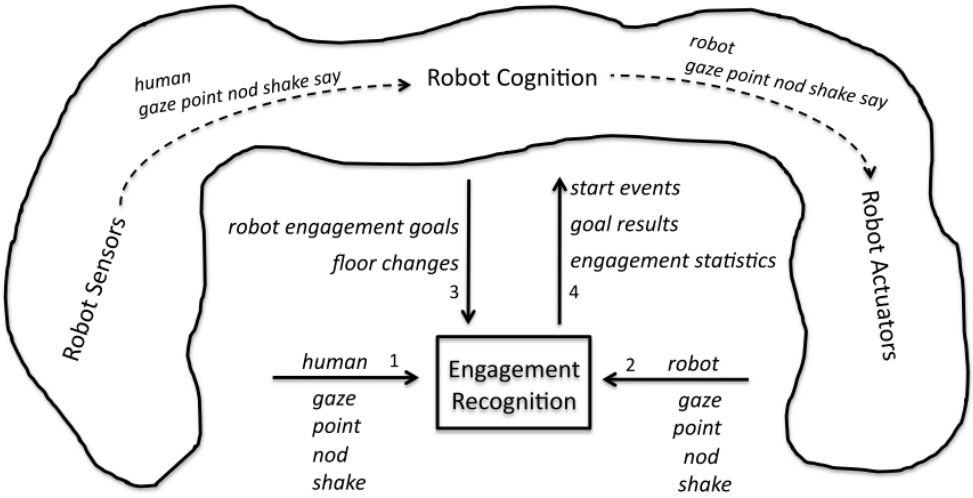


Figure 3.2: Internal architecture of the engagement recognition module.

Fig. 3.2 shows the information flow between the engagement recognition module and rest of the software that operates the robot. In ROS, this information flow is implemented via message passing, as described in the next section. The next section also specifies the state machine for recognizing each connection event type.

Notice first in Fig. 3.2 that the rest of the robot architecture, not including the engagement recognition module, is shown as a big cloud. This vagueness is intentional in order to maximize the reusability of the engagement module. This cloud typically contains sensor processing, such as computer vision and speech recognition, cognition, including planning and natural language understanding, and actuators that control the robot's arms, head, eyes, etc. However, the exact organization of these components does not matter to the engagement module. Instead we focus on the solid arrows in the diagram, which specify what information the rest of the robot architecture must supply to the engagement module. Starting with arrow (1), the engagement module needs to receive information about where the human is looking and pointing in order to recognize human-initiated directed gaze and mutual facial gaze events. It also needs to be notified of the human's head nods and shakes in order to recognize human backchannel events

and human gestural turns in adjacency pair events. The engagement module also needs to be notified (2) of where the robot is looking (in order to recognize the completion of a human-initiated directed gaze or mutual facial gaze), pointing and when the robot nods or shakes. This may seem a bit counterintuitive at first. For example, would not the engagement module be more useful if it took responsibility for making the robot automatically look where the human directs it to look? The problem with this potential modularity is that the decision of where to look can depend on a deep understanding of the current task context. You may sometimes ignore an attempt to direct your gaze–suppose you are in the midst of a very delicate manipulation on the table in front of you when your partner points and says "look over here." Such decisions need to be made in the cognitive components of the robot. Similarly, only the cognitive components can decide when the robot should point and whether it should backchannel comprehension (nod) or the lack thereof (shake).

Robot engagement goals (3) trigger the engagement recognition module to start waiting for the human response in all robot-initiated event types, except backchannel (which does not have a delay structure). For example, suppose the (cognitive component of the) robot decides to direct the human's gaze to a particular object. After appropriately controlling the robot's gaze and pointing, a directed gaze engagement goal is then sent to the engagement component. The floor in a conversational interaction simply refers to who is the (primary) person currently speaking (communicating). Floor change information (3) is needed to support recognition of adjacency pair events. In natural spoken conversation, people signal that they are done with their turn via a combination of intonation (dropping tone), gesture (mutual facial gaze) and utterance semantics (e.g., a question). The engagement module thus relies on the rest of the robot architecture, such as speech recognition and natural language understanding, to decide when the human is beginning and ending his/her turn. Similarly, only the cognitive component of the robot can decide when/whether to take and/or give up the robot's turn. Arrow (4) summarizes the information that the engagement recognition module provides to the rest of the robot architecture to coordinate and monitor the engagement process. First, the module provides notification of the start of human-initiated connection events, so that the robot can respond. The module also provides real-time feedback on the successful or unsuccessful completion of robot-initiated connection events (engagement goals). For example, if the robot directs the user's gaze to an object and the user does not look, the engagement module notifies the rest of the architecture, so that the robot can try again, if necessary. Finally, the engagement module provides various ongoing statistics, similar to those in 2.1, which the robot can use to gauge the health of the

engagement process and decide, for example, to initiate more connection events.

3.3 Engagement recognition module

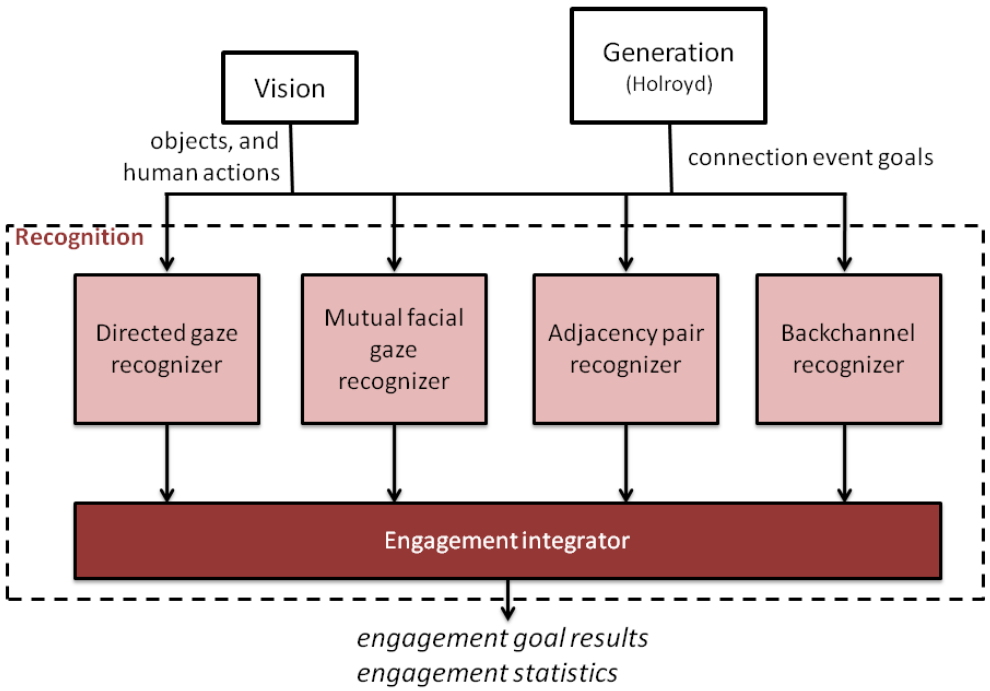


Figure 3.3: Internal architecture of the engagement recognition module.

Fig. 3.3 shows the internal architecture of the engagement recognition module, which consists of four parallel recognizers that feed information to an integrator process. More than one recognizer may be active at one time to allow for overlapping connection events. Each recognizer responds to a subset of the information coming into the recognition module. The state machine for each recognizer is shown in the next section. Each recognizer reports its start time, end time and, except for backchannel, its delay duration and whether it successfully completed its time line or failed (typically because the delay exceeded a threshold). The integrator process incrementally calculates the mean and maximum time between connection events, the mean and maximum delay times and the number of failed events per unit time, over both a recent time window and the whole interaction (baseline). All of these statistics are available to the rest of the robot architecture to provide an adaptive estimate of the current strength of engagement. For example, increases in recent versus baseline mean time between connection events, delay time and/or failure rate may indicate the human's desire to disengage. Exactly how to weigh these factors along with other information, such as the content of what the human says, is beyond

the scope of the engagement recognition module. Future experimentation with the system may yield further insight into this issue.

4 Human-robot implementation

Now that the human-robot architecture has been described, we will describe how the architecture was implemented as a reusable software module. Specifically the two modules contributed by this thesis: a module for engagement recognition, and a module for testing the engagement recognition module.

4.1 Recognition

The recognition module is responsible for recognizing robot and human-initiated connection events. It performs this task through parallel state machines each responsible for detecting one of the connection events (depicted in Fig. 3.3). Below, the implementation of the recognition module is discussed in more detail, specifically: each one of the four connection event recognizers, available statistics, and selected timeouts. Each state machine diagram includes a path for a robot-initiated event (above) and a path for a human-initiated event (below). The state machines are event driven and thus transition to new states when certain messages are received. The **bold** text in the diagrams represents a main transition for the particular arc, while the *italicized* text is included to show the remaining transitions. The term *unengaged* is used to indicate the point in time when the human leaves the table.

4.1.1 Directed gaze

In Fig. 4.1, recognition of a human-initiated directed gaze event is triggered by the human looking and optionally pointing at an object. After transitioning to the Human Waiting state, the recognizer waits until either the robot decides to respond by looking at the same object (in which case the Shared Gaze state is entered), time runs out, or the robot decides it wants to make eye contact or direct the human's gaze to another object instead (in which case the event fails). The Shared Gaze state always transitions to Succeed, which occurs when either the human or robot stops looking at the directed object. The state transition path for recognizing a robot-initiated directed gaze event is similar, except that the directed gaze goal (robot intention) triggers the transition from Start to Robot Waiting. At this point, the robot has already been commanded to look, and optionally point, at the directed object. As before, the recognizer

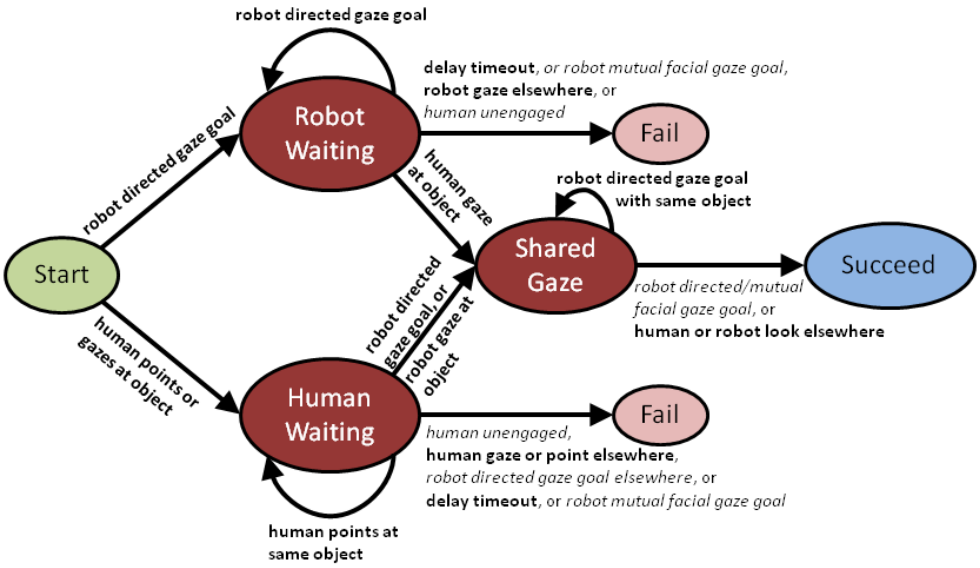


Figure 4.1: State machine for recognizing directed gaze.

waits, in this case until the human looks at the directed object, before entering the Shared Gaze state. If time runs out or the robot decides to make eye contact and is not also pointing, then the event fails. In order to properly handle directed gaze at multiple objects, it was necessary to have a one-to-one relationship between the directed gaze recognizer and specific objects in the environment (rather than having a single recognizer handle multiple objects). Thus, when a directed gaze is performed to a group of objects, an instance of the directed gaze recognizer is started for each one of the objects referenced, and succeeds as long as at least *one* of the recognizers succeeds.

4.1.2 Mutual facial gaze

Fig. 4.2 has a similar state structure to directed gaze, with a Mutual Facial Gaze state instead of Shared Gaze. The Mutual Facial Gaze state transitions to Succeed when either the robot or the human breaks eye contact. As in directed gaze, the Human Waiting and Robot Waiting states correspond to the recognition of human-initiated and robot-initiated events, respectively, and each of these states may lead to failure due to timeout. Also, at the point that the mutual facial gaze goal message arrives, the robot has already been commanded to look at the human's face. Finally, if the robot decides to look at another object (directed gaze goal) during either the Human Waiting or Robot Waiting state, the event fails (because the robot cannot both make eye contact and look at an object at the same time).

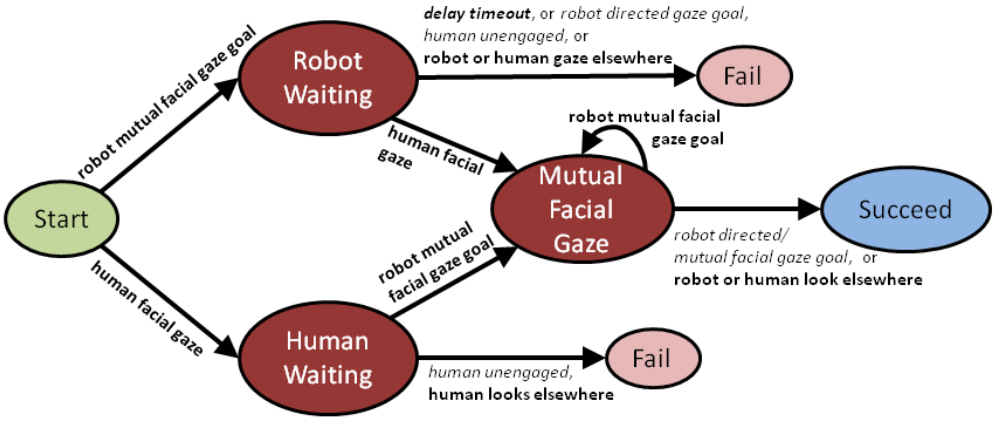


Figure 4.2: State machine for recognizing mutual facial gaze.

4.1.3 Adjacency pair

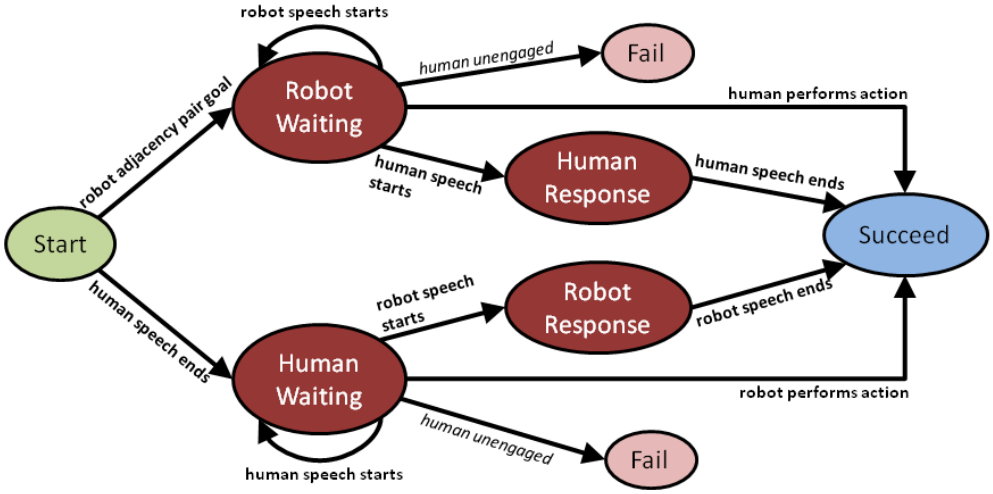


Figure 4.3: State machine for recognizing adjacency pair.

The state machine in Fig. 4.3 for recognizing adjacency pair events also has Human Waiting and Robot Waiting states (with timeouts to failure), on the human-initiated and robot-initiated recognition paths, respectively. The majority of transitions in this recognizer depend on the start and end of utterances, but the recognizer can also succeed when the robot or human perform actions. Unlike the previous two recognizers, this state machine could in fact be written more compactly in terms of an initiator and responder, but for consistency of understanding we have expanded out separate paths for the human and robot. We have not yet implemented the handling of third turns or barge-in (when one party starts taking a turn–not just a backchannel–without the other party first yielding the floor).

4.1.4 Backchannel

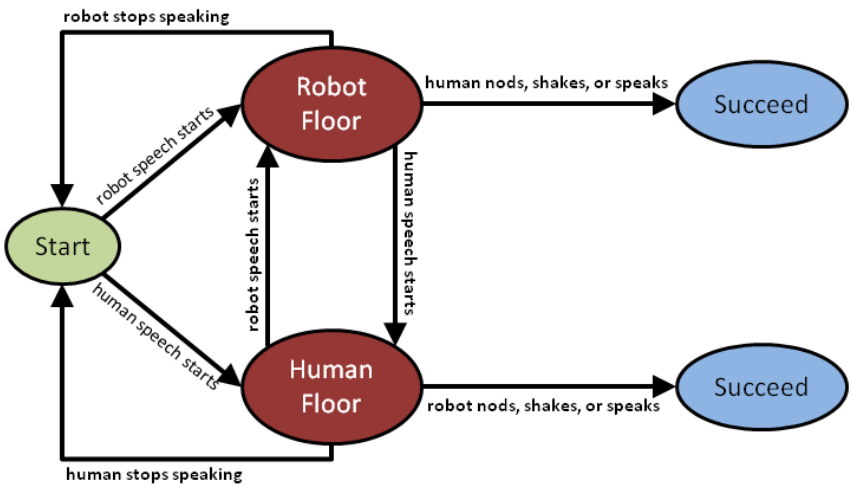


Figure 4.4: State machine for recognizing backchannel.

The state machine in Fig. 4.4 has no delays or failure states. Basically, the machine keeps track of who has the floor so that it can recognize a backchannel nod or shake by the other party. A human-initiated backchannel begins when the human starts speaking at which time it enters the Human Floor state. A backchannel occurs at any point during the human's speech when the robot nods or shakes its head, or speaks (e.g., "mhm", or "okay"). The recognizer returns to the start state once the human stops speaking. The robot-initiated path is symmetrical. It begins when the robot starts speaking, succeeds when the human nods or shakes his head, or speaks, and returns to the start state when the robot stops speaking.

4.1.5 Statistics

The recognition module performs several calculations during the course of the interaction. These calculations are gathered for the entire interaction, as well as for a sliding window of the interaction. These specific calculations are: the mean and max time between connection events, the mean and max delay for connection events, and the failure rate. The time between connections events, as discussed in Section 2.5, is defined as the time between the *start* of successive events, while the delay is the time between the start of the initiator's action and the start of the responder's action–except for adjacency pair which starts with the *end* of the responder's action. Refer to Section 2 for the specific definition of delay for each connection event type.

4.1.6 Delay timeout

One aspect of each of the connection event time lines, excepting backchannels, is the *delay*, or, more generally, the time between the initiator's action and the responder's action (see Section 2 for specifics). From the standpoint of an observer, there must be a point at which the connection event fails and the initiator moves on. Take for instance directed gaze: the initiator will stop pointing after a some time if the responder never responds. The values we used for maximum delay were gathered through analysis of human-human interactions (see Section 2.5) and then tweaked through testing until we felt it was the most natural. The maximum delay timeouts used for the robotic application can be seen Table 4.1. In the future, it would be ideal to discover these values in a more scientific way, such as machine learning.

	robot max delay (sec)
directed gaze	3.0
mutual facial gaze	1.8
adjacency pair	3.1
backchannel	n/a

Table 4.1: Maximum delay timeouts for each connection event.

4.2 Testing module

The testing module was developed as a utility for testing the recognition module's ability to recognize robot-initiated connection events. For example, the robot will initiate a directed gaze by instructing the human to look at a specific object, e.g., "Please look at the blue-plate." The robot will then tell the human whether or not they successfully looked at the desired object, e.g., "You successfully looked." This series of events will repeat until the testing module is terminated which allows the functionality of each of the connection event recognizers to be repeatedly tested in a simple manner. The testing module is equipped to test all four of the connection events.

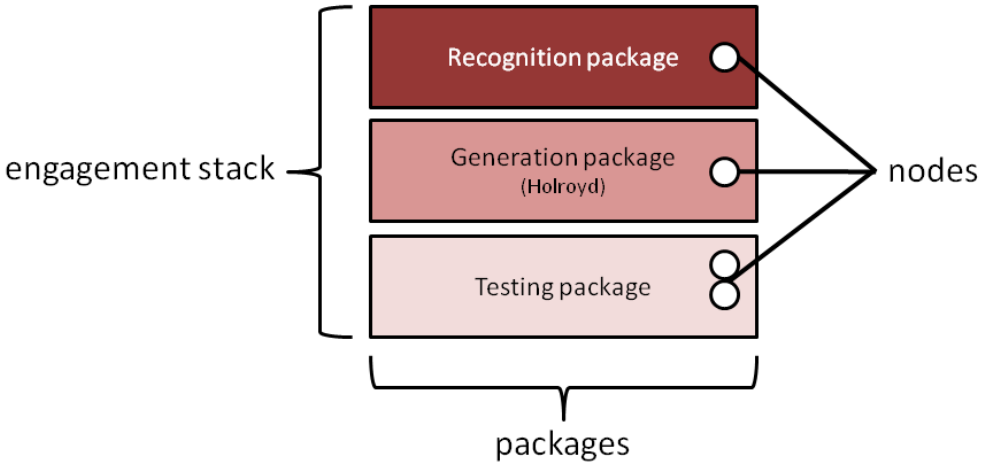


Figure 4.5: The engagement stack.

4.3 ROS details

To implement the architecture described in the preceding sections, we chose the Robot Operating System (ROS) framework [20], because it offered the highest likelihood that our work could be easily shared with other robot researchers and developers. ROS is a distributed framework of processes (called *nodes*) that communicate via message passing. Nodes are grouped into *packages*, which can be easily shared and distributed. Information flows into and out of an ROS node via messages (called *topics*) and *services*. Services are a higher-level abstraction that uses messages to implement return values (similar to a remote procedure call). Each type of information flowing into the engagement recognition module, or node, except for the robot engagement goals, is a separate ROS topic (message type).

We have contributed an ROS *stack*, i.e., a collection of packages, called "engagement" (depicted in Fig. 4.5). Within the stack are the recognition, generation, and testing packages which contain one or more nodes. For specific details regarding the implementation of the engagement stack, the source code and documentation can be found at: `http://ros-engagement.sourceforge.net/`.

This thesis contributed the recognition package, a coordinated thesis [11] contributed the generation package, and both theses contributed to the testing package. The recognition module was implemented as an ROS node using C++ while the testing module was implemented as an ROS node using Java.

4.3.1 Example message

```
# Unique id for the actor that initiated this event
ActorID actor
# The timeout for this human-initiated connection event
float32 timeout
# The objects associated with this connection event
Entity[] objects
# What action the human used
int8 action
# Declare constants for the different actions
int8 POINT=3
int8 LOOK=4
```

Figure 4.6: Example ROS message pertaining to human directed gaze.

Fig. 4.6 contains the definition of an ROS message pertaining to a human-initiated directed gaze. The *actor* field contains the identifier specific to the human performing the directed gaze. The *timeout* field contains the maximum delay timeout value for directed gaze (see Section 4.1.6). The *objects* field contains the set of objects the human is directing gaze toward. The *action* field specifies the action used to direct the gaze, which can be set to either the *POINT* or *LOOK* constants. Lines beginning with # are taken as comments. All of the messages received by the recognition module's state machines are implemented as ROS messages.

4.3.2 Example service

```
# Unique string identifier for the actor
engagement_msgs/ActorID actor
# A flag which determines between DONE and NOT_DONE
engagement_msgs/Flag done
# The sequence of entities
engagement_msgs/Entity[] objects
---
# Use a Flag message as a response
engagement_msgs/Flag result
```

Figure 4.7: Example ROS service pertaining to a robot directed gaze goal.

Fig. 4.7 contains the definition of and ROS service pertaining to a robot-initiated directed gaze goal. The *actor* field contains the identifier specific to the human whose gaze the robot wishes to direct. The *done* field determines whether the robot wishes to start a directed gaze

(*NOT_DONE*) or end the directed gaze (*DONE*). The *objects* field contains the set of objects the robot is directing gaze toward. The three dashed lines separate the service request (above) from the service response (below). Thinking of a service as a remote procedure call, the service request are the parameters to the procedure call, and the service response is the return value, or values, of the procedure call. The *result* field contains the response to the directed gaze service, i.e. whether the connection event succeeded or failed. Lines beginning with # are taken as comments.

The recognition module provides four services (one for each type of connection event). The return value of these services informs the caller of whether the goal succeeded or failed (i.e., whether the recognizer machine terminated in the *succeed* or *fail* state). Additionally, the recognition module gathers statistics which it provides at the request of a service call.

5 Human-robot demonstration

We are interested in taking a step toward autonomous robots that can collaborate naturally with humans. Thus, we developed a working demonstration of our engagement models rather than taking a "Wizard of Oz" approach in which the robot would not be autonomous. In this demonstration, described below, the human and robot collaborate to play a game called "tangrams" which will be explained in Section 5.2.

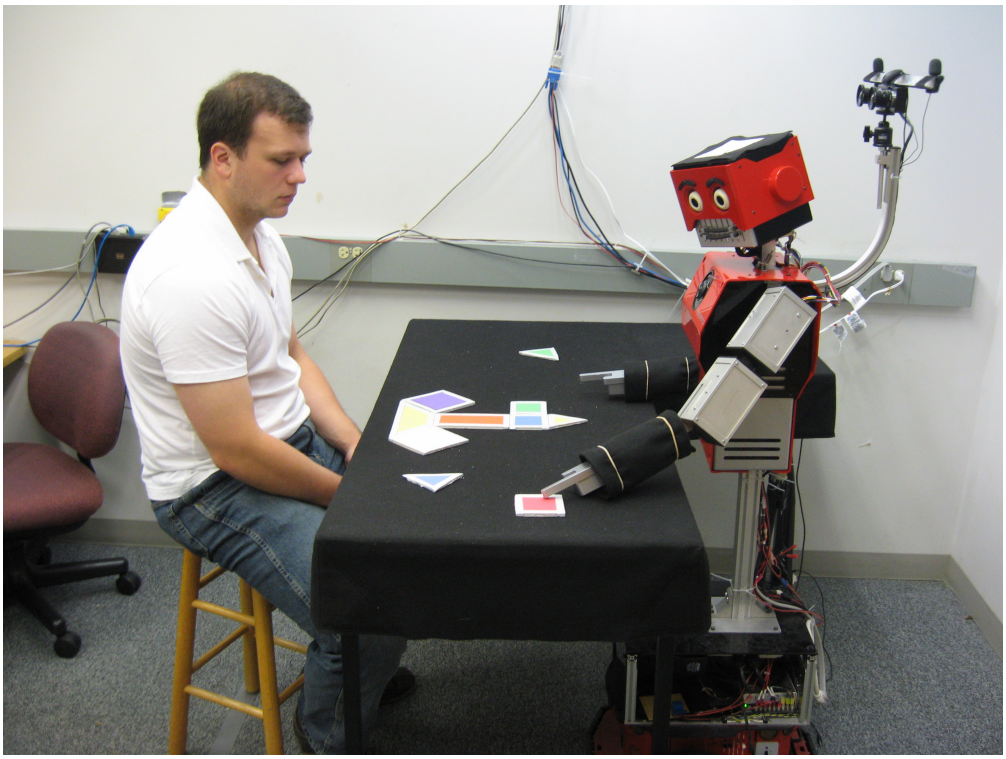


Figure 5.1: Melvin and author making a tangram.

5.1 Humanoid robot

The robot employed in this work is named "Melvin", shown in Fig. 5.1. Melvin was designed by Mitsubishi Electric Research Laboratories (Cambridge, MA) and the University of Sherbrooke (Quebec, Canada), and was manufactured by Robomotio, Inc. (Quebec, Canada). Melvin has many traits that resemble a human: two arms, a torso, and a movable expressive face. Melvin is also attached to a wheeled base (which is not being used for this work). He is able to move his

body through the use of 16 servomechanisms, which control the motors throughout his body. Melvin's hands each contain a single, stationary, finger which allows him to point at objects but does not allow him to grasp any objects.

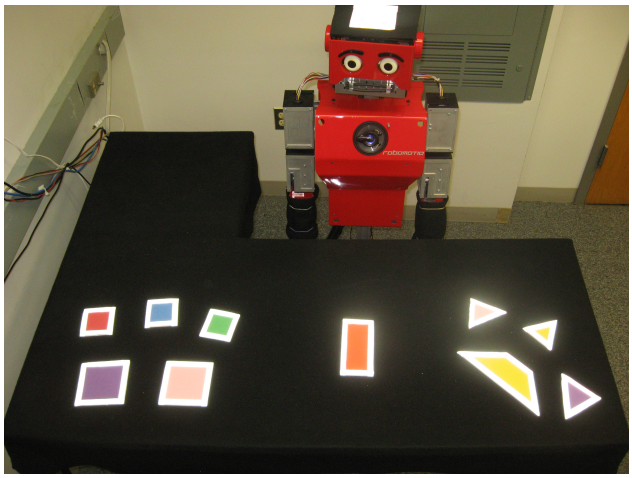
5.2 Task description

Tangrams are puzzles consisting of several colored, small, flat pieces which are placed together to construct a specific shape. Typically the pieces can be oriented in any manner so long as they do not overlap. For the sake of simplifying the vision processing, we've placed an additional constraint on the game such that two touching pieces **must** share an edge of equal length.

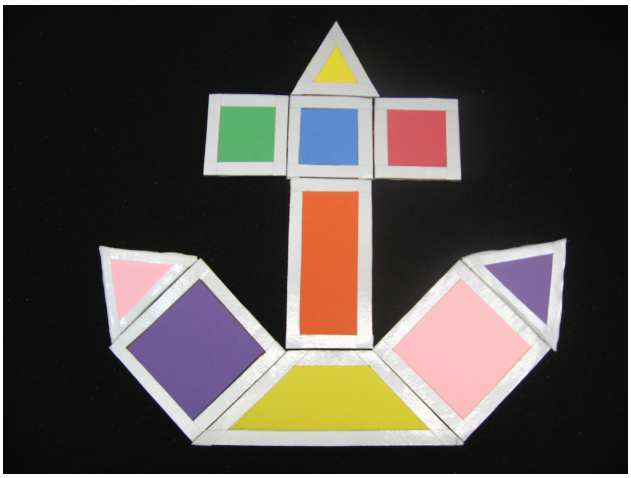
In this human-robot demonstration, the human interacts with Melvin in a setting depicted in Fig. 5.1. There is a table between the human and Melvin on which there are several tangram pieces. Melvin instructs the human to move one piece at a time in order to properly construct the final shape–which Melvin has knowledge of, but the human does not. If the human moves the piece to an incorrect location, Melvin will indicate that it is incorrect by saying "that does not look quite right" and then he will repeat the previous instruction. Also at some points during the conversation, Melvin asks the human, "please point to the piece you would like to move next," and–assuming the piece can be moved at the time–then Melvin tells the human where to move the selected piece. We decided to ask the human to point to a piece in order to make the collaboration more mixed initiative. When all the pieces are correctly placed Melvin will indicate that the game is complete and tell the human what shape they have constructed. For a transcript of a sample interaction see Appendix C.

5.2.1 Anchor puzzle

Melvin instructs each participant how to use the given pieces to construct the final shape which is an **anchor**. This puzzle consists of 10 pieces which can be seen in Fig. 5.2. Fig. 5.2b depicts the anchor in its completed state. This shape was chosen due to the fact that it has many simple shapes (typical tangram puzzles consist of 7 pieces), and that the human should not be able to easily recognize the final shape (which could cause the human to get ahead of Melvin and ignore him altogether).



(a) Start of the anchor puzzle



(b) The completed anchor puzzle

Figure 5.2: Start and completion of the anchor tangram puzzle.

5.2.2 Software modules

There was an extensive amount of work that went into the implementation of the tangram game. Software modules were needed to control the robot, perform computer vision, and keep track of the state of the task. Holroyd and I jointly developed and tested these modules, but I was focused on the modules responsible for computer vision, and Holroyd was responsible for controlling the robot, as well as tracking the state of the task. The software modules that I developed are discussed below.

Vision

While this research is not related to computer vision, a working vision module was required in order to create an autonomous system. Thus we attempted to simplify the vision required for the task as much as possible, e.g., controlled lighting of distinctly colored objects on a black background. Two cameras are used for vision. One camera is located above and behind Melvin's head (face tracking), while the other is located on the ceiling over the table (object tracking). In order to track the human's face in the frame we use Morency's Watson [17] system. It is able to track multiple faces and determine the position and orientation of the faces in the frame of video, while also determining if the face is performing a nod, or shake. OpenCV and C++ are used to locate objects within the range of the table. Blob detection for specific color thresholds is used to find all the colored pieces on the table, for which a series of points is extracted relating to the contour of the piece. Blob detection and the camshift technique are used to track the

human's hand position and orientation. Using configurable thresholds for color, this system can recognize pieces that are red, blue, green, yellow, orange, purple, and pink.

The low level information gathered from the frames of vision is processed to determine what object the human is pointing to, where the human is looking, and the current state of the tangram pieces. For each piece, we must extract the points relating to the corners of the piece, determine what piece(s) it is currently connected to, and then compare the current pieces to the expected final configuration of the pieces to determine which pieces need to be moved.

Human pointing is recognized when a hand is oriented in the direction of one, or more, tangram pieces for a series of frames. An action begins when a hand crosses the table, and ends when the hand leaves the table. The looking behavior occurs when a face is oriented toward one, or more, of the tangram pieces for several frames. A nod or shake occurs, as indicated by Watson using a threshold value for each.

6 Evaluation

To evaluate our approach to recognizing engagement behaviors in human-robot interaction, Holroyd [11] and I conducted a between-subjects three-arm controlled experiment in which we shared a single condition.

6.1 Experimental design

In the shared (or *OPERATIONAL*) condition, the robot recognizes all of the human-initiated connection events and uses the standard timeout values (see Section 4.1.6) for connection events while also properly generating robot-initiated connection events. The control (or *DEGRADED*) condition corresponds to the baseline condition where the robot does not recognize any human-initiated connection events and uses a randomly selected timeout of either 0 or double the standard timeout for robot-initiated connection events. The third condition, which is the control condition for Holroyd, is where the robot recognizes all of the human-initiated connection events but does not generate any connection events of its own (in fact, the robot does not move at all).

33 male and 10 female graduate and undergraduate students from WPI, between the ages of 17 and 26, participated in this experiment for a total of 43 participants. Each participant was arbitrarily placed in one of the three conditions. Overall we had 14 participants in the *OPERATIONAL* condition, 14 participants in the *DEGRADED* condition, and 15 participants in the third condition.

6.1.1 Hypotheses

We have posited the following hypotheses comparing the *OPERATIONAL* condition to the *DEGRADED* condition:

1. (a) Participants will report that Melvin is more human like.
   (b) Participants will report that the interaction with Melvin is fluent and natural.
2. Participants will report that Melvin understands what they did more often.
3. Participants will make more eye contact.
4. Participants will perform more backchannels.

5. Participants will produce more utterances.

6. The mean time between adjacency pairs will be less.

7. The average adjacency pair delay will be less.

8. The puzzle completion time will be less.

We also hypothesized that in the *DEGRADED* case:

9. Participants will make progressively less eye contact during the interaction.

Hypotheses 1(a), 1(b), and 2 will be tested via a questionnaire (see section 6.1.3). Hypotheses 3, 6, 7, 8, and 9 will be tested via the system statistics reported by the recognition module, and hypotheses 4 and 5 will be tested via coding of the recorded videos of the interaction.

6.1.2 Participant instructions

Prior to the interaction the participant reads a sheet of instructions (see Appendix B) which explain the task and certain restrictions on their behavior. To simply the vision processing used to recognize the human hand, the participant was asked to roll up their sleeves past their elbows. The names of different shapes (trapezoid, rectangle, etc) were also explained in order to ensure that the participant was familiar with these names. Finally, the words Melvin uses to explain where to place pieces (left, right, top, bottom) are explained. Each participant was given the same sheet of instructions.

6.1.3 Questionnaire

After the interaction, each participant also completed a questionnaire comprised of 28 questions on a 7-point Likert scale ranging from "Strongly Disagree" (1) to "Strongly Agree" (7), 4 questions regarding personal experience, and an open space for additional comments. The Likert scale questions can be seen in Appendix A.1, and the additional questions can be seen in Appendix A.2.

The questionniare used was adapted from the questionnaire developed by Sidner *et al.* [23], and from the original Interactive Experiences Questionnaire developed by Lombard *et al.* [15]. The Likert scale questions were presented in the same, randomized, order for all participants.

Table 6.1: Relevant questionnaire results

		Mean		Std. Dev.		
Hypothesis	**Question**	**Oper.**	**Deg.**	**Oper.**	**Deg.**	**p-value**
N/A	**5**	7	6.7	0	0.61	0.092^{+}
2	**13**	5.9	5	1.1	1.2	0.049^{*}
1 (b)	**15**	4.2	3	1.4	1.6	0.038^{*}
1 (b)	**17**	3.6	2.4	1.1	1.3	0.017^{*}
2	**23**	5.6	4.6	1.2	1.7	0.085^{+}
2	**25**	5.9	4.4	1.1	1.4	0.003^{***}

6.2 Results

Table 6.1 depicts the questionnaire results that were at least trending toward significance. For the full results of the experiment see Appendix D. The results contain the mean, standard deviation, and p-value for both the *OPERATIONAL* and *DEGRADED* conditions. A two tailed, unpaired t-test was used to calculate the p-values for each question. There were two questions with trending results, three questions with weakly significant results, and one question with a highly significant result. The results support hypothesis 2 (participants will report that Melvin understands what they did more often), and weakly support hypothesis 1 (b) (participants will report that the interaction with Melvin is fluent and natural).

6.3 Discussion

While the results support one of our hypotheses and weakly support another hypothesis, the remaining hypotheses were not supported. We believe this is due to vision system errors, namely in piece placement and pointing recognition, that occurred during the majority of the interactions. Due to time constraints, we were unable to resolve this issues with enough time to conduct another experiment. We also believe the *DEGRADED* condition should be revised with respect to how pointing is handled, and how timeout values are selected. These issues, and possible improvements are discussed below.

6.3.1 System errors

The majority of the system errors occurred when the participant placed a piece in the correct position, as instructed by Melvin; however, the vision system then deemed this piece to be

$^{+}$Data is trending, *Data is weakly significant, ***Data is highly significant

incorrectly placed and stated to the participant "That does not look quite right". This often resulted in the participant merely rotating the piece to a different edge to appease Melvin, but in some cases the participant became confused and moved the piece to an incorrect location. This error occurred up to 10 times for some participants. One participant said it best:

"Melvin was great to interact with. The one moment that kind of took me out of the feeling of interaction was when I put the piece in what I was sure was the right spot, but Melvin didn't think so. He asked me to fix the piece so I rotated it 90 degrees (which did nothing in effect because it was a square) and put it back where I had placed it before. He then accepted this. Other than that one moment, I felt engaged with Melvin, even grinning and almost laughing at times out of disbelief that I was actually interacting with a robot in this way so naturally."

Other vision related errors included things such as: incorrectly recognizing what objects the human was pointing to, and imperfect color and corner detection of tangram pieces. There were also some issues in the control of the robot. For instance, segments of speech would occasionally be lost, i.e., not synthesized, however, the robot's mouth would still move as if it were speaking. The accuracy of the robot's pointing was also sometimes unreliable, and its movement erratic. All of these different issues detract from the engagement and can cause the human to become confused.

6.3.2 Possible improvements

We have several ideas for improvements that could be made to the experiment, such as: tweaking behaviors and the timeout values used in the *DEGRADED* condition, improving the processing speed for the computer vision related modules and performing a within subjects experiment rather than a between subjects experiment.

Experimental design

One of the most difficult aspects of designing an experiment is defining the control condition–or in this case the *DEGRADED* condition. Because we are testing two separate engagement modules, i.e., recognition and generation, we preferred to degrade each module separately (rather than testing an interaction with engagement and one without). However, one does not have to select only a single control condition. Instead you could degrade each component of a module separately to have multiple control conditions to see the various effects of the isolated compo-

nents. For example, remove only directed gaze from one condition, and only mutual facial gaze from another condition to determine the effects these connection events have on the interaction. This translates to more control conditions, more participants, and much more time. Thus, we decided to have one control condition for each of the engagement modules, but this is made more difficult due to the nature of engagement in interactions. The *OPERATIONAL* condition should, ideally, be similar to a natural interaction, but what about the *DEGRADED* condition? This should be some sort of simple and cheaper interaction, but there are no guidelines stating exactly what that entails for human-robot interaction. Thus, we had to make the choice of what behaviors to degrade without creating a degenerate interaction. Given more time, it would be ideal to have more control conditions, but this is rarely the case.

Pointing

During the *DEGRADED* condition, human-initiated connection events are not recognized, e.g., when the participant points at a piece it is not noticed. When a point is not recognized Melvin simply selects an available piece to move–meaning that Melvin may happen to select the piece the participant pointed to. This would give the illusion that Melvin had, in fact, recognized the point. In some instances, Melvin happened to select each of the pieces the participant pointed to during the interaction, thus they had the opinion that Melvin did see where they were pointing despite it being only random chance. We believe this contributed to the lack of results on the questionnaire. The system could be configured such that Melvin **never** selects the piece the human pointed to, but this would require shared knowledge of the *DEGRADED* condition across modules.

Face and hand tracking

Due to architecture decisions (see Section 3), the generation module is responsible for doing certain behaviors such as tracking the participant's hand or face, and periodically performing a glance at the participant. These behaviors remained enabled during the *DEGRADED* condition, which may dilute the desired effects of the condition. In the future, it may be ideal to remove these behaviors during the *DEGRADED* condition.

Selection of timeouts

Additionally the selection of the timeout to use in the *DEGRADED* condition could be improved. Currently, the timeout selected for a connection event is equally distributed between a timeout

of 0, and twice the standard timeout for the connection event. It may be better to weight the selection such that it is less likely to select a timeout of 0. Thus, on average, it would have a long timeout, but sometimes it would timeout immediately. This may amplify the feeling that Melvin is acting erratically. Optionally, we could remove the ability to have a timeout of 0 which would effectively force the participant to wait an extended period of time throughout the interaction. Ideally, it would be interesting to have multiple conditions to test each of these conditions, but this is not always feasible.

Algorithm processing speed

It is also pertinent to discuss the prevalence of delay in the system operations. The vision processing currently requires too much processing time to determine the available moves, and whether pieces are placed properly. This causes a noticeable delay during the interaction where Melvin is waiting for the processing to complete. This pause can cause the participant to lose interest, i.e., become unengaged, or confuse the person. The *DEGRADED* condition amplifies these effects, however, they are still present in the *OPERATIONAL* condition. This makes the difference between the two conditions much more subtle which may explain why the answers to questions 9, 15, and 17 (see Appendix A.1) are similar for both of the conditions. Improving the functionality of the vision processing such that the processing delay is very minor will greatly improve the interactions and thus make the difference between conditions more apparent.

Within subjects

One final possibility would be to run a *within subjects* experiment. In this type of study, each participant would participate in each of the conditions of the experiment, but the order given to each participant would be randomized. The participants would be asked to complete a questionnaire either after completing each of the conditions, or at the end of the experiment. This should make the degradations of the system more apparent to the human and may improve results with respect to the *DEGRADED* condition. The effect of degraded recognition may be too subtle for the human to realize unless they have a basis for comparison, i.e., the *OPERATIONAL* condition. A within subjects design also requires a unique puzzle for each condition to account for a difference in puzzle difficulty. Thus, more puzzles would need to be created prior to running a within subjects experiment.

7 Conclusion

Engagement is the process that governs human interaction, and thus in order for robots and humans to collaborate in a natural way, robots must have a model of engagement. The robot must be able to both recognize the human's engagement behaviors, and properly generate its own engagement behaviors at the correct times.

Based on analysis of human-human interactions, we have developed a model of engagement recognition [21] and Holroyd has developed a model for engagement generation [13] for use in a robotic system. These models were implemented as open-source ROS nodes with the hope of benefiting the human-robot interaction research community.

Holroyd [11] and I evaluated the validity of our models through a between-subjects three-arm experiment (in which we shared a single condition) where humans interacted with our humanoid robot to create a puzzle consisting of ten tangram pieces. Fortunately, we found two results to be trending toward statistical significance, three results to be weakly significant, and one result to be highly significant.

7.1 Future work

There are many interesting areas for further research with respect to recognizing engagement behaviors. Some of these areas are discussed below.

7.1.1 Engagement initiation and termination

One area that needs further research is behaviors that work to initiate or terminate the engagement. Currently the system recognizes the start of engagement when the human's face appears in the frame and the end of engagement when the human's face disappears from the frame (such as when the human walks away). While these behaviors oftentimes indicate the start and end of an interaction, it would be more beneficial for the robot to be able to reason about the human's *intent* to engage or disengage. Whether this be due to a change in the mean time between connection events, repeatedly failed connection events, or other behaviors, it will take further research to formulate a model of these behaviors. Once a model has been created it could then be translated into software and merged into the recognition module. One could then design a

study such that the robot is able to adapt to the human's intent to engage and disengage, versus another condition where the robot simply continues its task. Ideally, participants would find the adaptive system more engaging and natural when compared to the system which does not recognize the intent to engage or disengage.

7.1.2 Maximum delay timeout

The delay timeout values (see Section 2.5) currently used for each connection event were based on the max delay values gathered from analysis of human-human interactions with some additional tweaking through testing. The selection of these values plays an important role in the naturalness of the interaction. For instance, if the timeout for directed gaze is twenty seconds, the robot could wait for the human to look at an object for twenty seconds prior to continuing with the task. In some situations, and tasks, this may be the appropriate timeout, however, further research is needed to determine exactly how these timeout values should be calculated for specific tasks. One option, is to use machine learning on analyzed human interaction data to find the specific timeout values. Another option could be to use the mean time between connection events to determine the specific timeout values. This necessitates using the guessed timeout values as the starting values for the interaction and then using the mean time between connection events to converge on new values.

Simply calculating timeout values for a particular task may not be enough. Timeout values for connection events are not universal constants, but instead differ between tasks and participants, thus having the ability to adapt the timeouts throughout the interaction should improve the overall sense of engagement. One could also run a study comparing two systems: an adaptive system, and a non-adaptive system. The participants should find that the adaptive system is more natural and human-like, however, the difference between the two conditions is quite subtle and most likely would need additional work prior to executing an experiment.

7.1.3 Multi-party interactions

Another avenue of future research would be to investigate engagement with respect to multi-party interactions, i.e., interactions with more than one participant. The recognition module has been designed with multi-party interactions in mind, but, it has only been tested in single participant interactions. This would include testing the recognition module to determine that it properly keeps track of the engagement between two humans in an interaction, as well as between each human and the robot. One could then design an experiment such that multiple

participants collaborated with a robot to, for example, play a game of cards with one condition where the robot properly reasons about the engagement, and another condition where the robot does not reason about engagement. This study would aim to validate the recognition module's ability to improve a multi-party interaction.

7.1.4 Non-humanoid robots

Engagement is a natural process for humans interacting with other humans, and we are working to validate that it improves the naturalness of interactions with *humanoid* robots as well. What about *non-humanoid* robots? Ideally, interacting with a non-humanoid robot with the ability to reason about engagement should still feel more natural than interacting with one without that ability. Several issues come to mind regarding this implementation. First and foremost, generating engagement behaviors, e.g., pointing, nodding, facial gaze, in a non-humanoid robot could be difficult. What does it mean for a robot without a "face" to make facial gaze, or nod? What actions can non-humanoid robots perform to maintain engagement that humans will understand without training? Also, designing an interaction between a human and a non-humanoid robot in order to elicit engagement behaviors is not readily apparent. Further, the task of performing computer vision to reason about the human's behaviors may be more challenging in a non-humanoid robot–for instance a mobile robot which is close to the ground. Answering these, and other questions, would be an extremely interesting topic to investigate further.

A Tangram questionnaire

A.1 Likert scale questions

1. Melvin looked at the table and the puzzle pieces at appropriate times.

Strongly Disagree	1	2	3	4	5	6	7	Strongly Agree

2. The interaction felt natural all the time.
3. I always knew what object Melvin looked at.
4. Melvin was reliable.
5. Melvin pointed at objects during the interaction.
6. The puzzle piece descriptions were easy to understand.
7. I looked at Melvin's face often.
8. I always knew what object Melvin pointed at.
9. Melvin slowed me down during the interaction.
10. Melvin always looked at me in a natural way.
11. Melvin seemed more like a human than a robot.
12. I like Melvin.
13. Melvin responded appropriately to my actions.
14. I always knew what object Melvin talked about.
15. I spent a lot of time waiting for Melvin to tell me what to do.
16. I could easily identify the objects that Melvin referred to.
17. There were awkward pauses during the interaction when I wasn't sure what was supposed to happen next.
18. I always understood Melvin's instructions.
19. I spoke to Melvin during the interaction.
20. I could easily tell which objects Melvin looked at.
21. Melvin makes me feel comfortable, as if I am with a friend.
22. I easily found the puzzle piece that Melvin described to me.
23. Melvin always understood what I was doing.
24. Melvin looked at me at appropriate times.

25. Melvin always knew what I was doing when I pointed to or moved a piece.
26. I made eye contact with Melvin frequently.
27. I could easily tell the object that Melvin pointed to.
28. Melvin looked at me during the interaction.

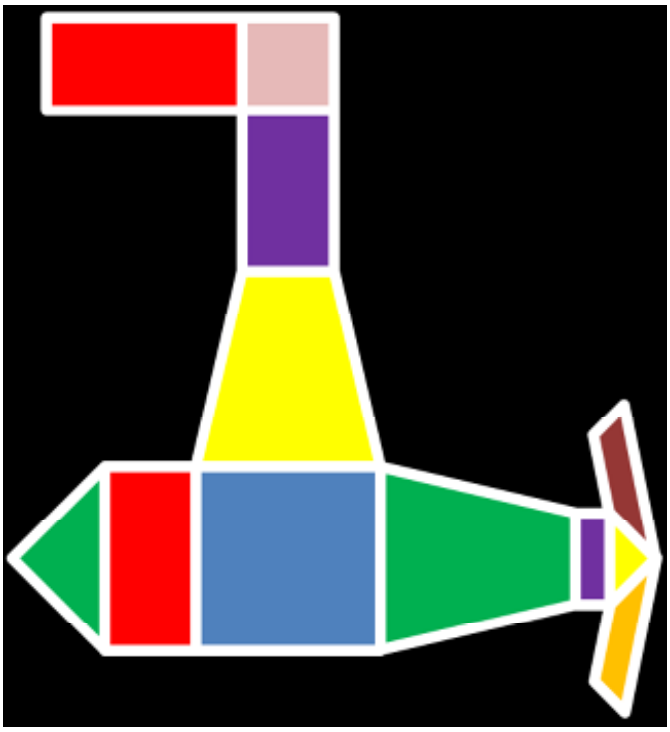
A.2 Personal experience questions

29. How old are you?
30. Please indicate your gender.
31. What is your level of education?
32. How much do you know about robots?
33. Additional comments about Melvin or the interaction.

B Pre-study tangram explanation

Tangrams

You will interact with a humanoid robot, Melvin, to construct a **tangram**. A tangram is a puzzle containing several flat, colored pieces arranged to form a specific picture. For example, the twelve pieces below make a submarine picture (this is **not** what you will be making!).

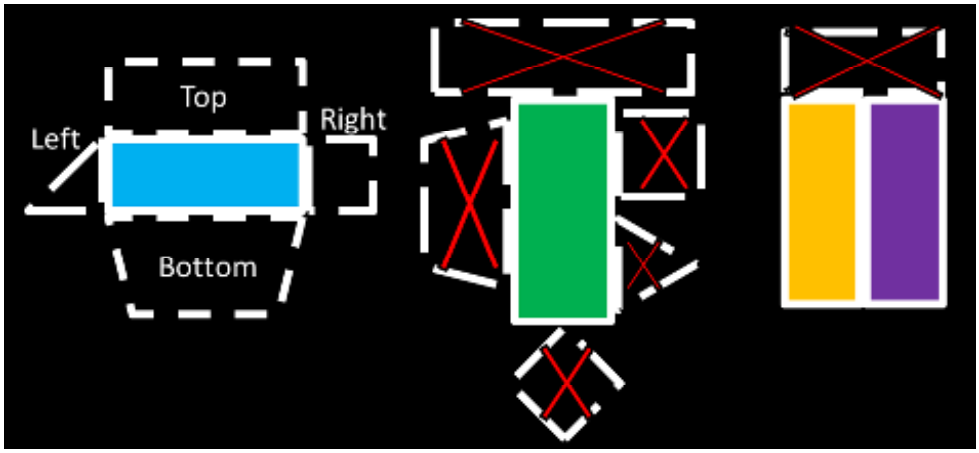


During the interaction, Melvin will **direct you where to place pieces** one-by-one, but will **not** tell you the name of the final picture until you have finished.

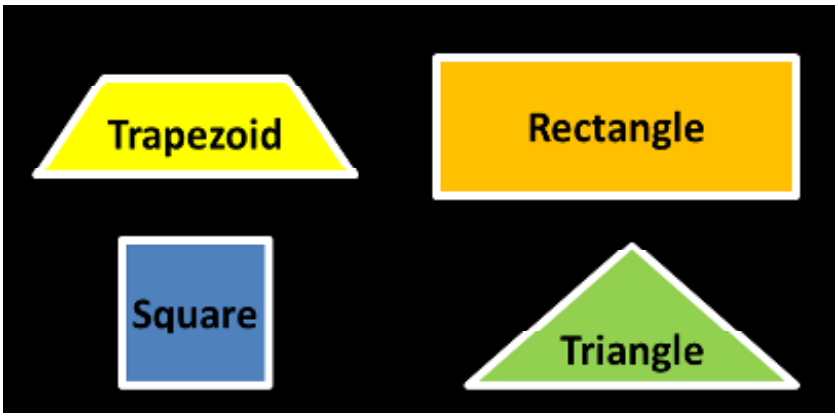
- Melvin can hear you speak, but **cannot understand** what you say.
- Please keep your hands in your **lap** while you are not moving the pieces.

There are some constraints on how pieces may be placed.

- Pieces **must remain flat** on the surface of the table and may not overlap.
- **All** the pieces will be used to construct the final picture.
- Properly aligned pieces should have **exactly one edge** of one piece touching **exactly on edge** of another piece. The edges that touch **must be equal** length. The pieces shown with red X's are i**ncorrectly** placed.



Note: Melvin will use the terms 'top', 'bottom', 'left', and 'right' as they pertain to **your** point of view.



Finally, prior to beginning:

1) Please remove any watches, bracelets, or anything else that is on **either** of your wrists.
2) Please roll up the sleeves of your shirt to your elbow.

C Transcript of a tangram game

- <*Action taken by participant*>, [**action taken by Melvin**], "Melvin's synthesized speech"
- [**DG at ***], [**MFG**], and [**AP**] indicate the start of the respective connection events
- ***** Indicates a failed connection event, + Indicates an optional connection event
- **Note**: robot-initiated adjacency pairs were started frequently during the interaction.

1. <*Looks at the table*> <*sits on stool*> <*facial gaze*> <*looks at the table*> <*facial gaze*>
2. [**AP**][**facial gaze**] "Hello, let's make a tangram." [**MFG+**]
3. [**AP**]"We are going to use the pieces on the table <*scans the table*> to make a picture together. As you can see, <*facial gaze*> on the table we have: squares, triangles, <*scans the table*> [**DG at rectangle**] a <*facial gaze*> rectangle, and <*looks at the rectangle*> [**DG with pointing at trapezoid**] a <*looks at the trapezoid*> trapezoid of <*facial gaze*> different colors [**MFG+**]." <*looks at the table*>
4. [**AP**][**facial gaze**] "Okay, <*scans the table*> let's start." [**MFG+**]
5. [**AP**][**facial gaze**] "Please move [**DG with pointing at blue piece**] the <*looks at the blue piece*> blue piece [**facial gaze**] so <*grabs the blue piece*> that it is touching the top of [**DG at rectangle**] the rectangle." <*looks at the rectangle*> [**MFG+**]
6. <*Moves the blue square to the top of the rectangle*> [**looks at the human's hand**]<*facial gaze*>
7. [**AP**]<*Looks at the table*> [**facial gaze**] "Ummmm, [**looks at the table**] good." <*facial gaze*> [**MFG+**]
8. [**AP**][**facial gaze**] "Please move [**DG with pointing at yellow triangle**] the <*looks at the yellow triangle*> yellow triangle [**facial gaze**] so that it is touching the top of [**DG at blue piece**] the blue piece <*looks at the blue piece*>." [**MFG+**] <*grabs the yellow triangle*>
9. <*Moves the yellow triangle to the top of the blue piece*> [**looks at the human's hand**] <*facial gaze*>
10. [**AP**]<*Looks at the table*> [**facial gaze**] "Ummmm, <*facial gaze*> [**looks at the table**] good." [**MFG+**]

⋮ ⋮ ⋮

11. [**AP**]<*Looks at the table*> [**facial gaze**] "Please point to the piece you would like to move next." [**MFG+**]
12. <*Points to the pink square*> <*facial gaze*> <*looks at the pink square*> <*facial gaze*> [**looks at the pink square**]
13. [**AP**][**facial gaze**] "Please move [**DG with pointing at pink square**] the <*looks at the pink square*> pink square [**facial gaze**] <*facial gaze*> so that it is <*looks at the table*> touching the top right of [**DG at trapezoid**] the trapezoid. Make sure <*grabs the pink square*> the square is touching the slanted edge of the trapezoid." [**MFG+**] <*looks at the table*>
14. <*Moves the pink square to the top right of the trapezoid*> [**looks at the human's hand**]
15. [**AP***][**facial gaze**] "Ummmm, [**looks at the table**] good." [**MFG+**]<*facial gaze*>
16. [**AP**][**facial gaze**] "Please point to the piece you would like to move next." [**MFG+**]
17. <*Looks at the table*> <*facial gaze*> <*looks at the purple triangle*> <*points to the purple triangle*> <*facial gaze*> [**looks at the human's hand**]
18. [**AP**][**facial gaze**] "Please move [**DG* with pointing at purple triangle**] that [**facial gaze**] <*facial gaze*> so that it is touching the top right of <*looks at the table*> [**DG at pink square**] the pink square." <*looks at the purple triangle*> [**MFG+**]
19. <*Grabs the purple triangle*> <*moves the purple triangle to the top right of the trapezoid*>
20. [**AP**]<*Looks at the table*> [**facial gaze**] "Ummmm, <*facial gaze*> [**looks at the table**] good." [**MFG+**]<*looks at the table*>
21. [**AP**][**facial gaze**] "Please <*facial gaze*> move [**DG with pointing at green square**] <*looks at the green square*> this [**facial gaze**] <*facial gaze*> so that it is touching the left side of <*looks at the table*> [**DG at blue piece**] the <*facial gaze*> blue piece <*looks at the table*> ." [**MFG+**]
22. <*Looks at the green square*> <*grabs the green square*> <*moves the green square to the left of the blue piece*> <*facial gaze*> [**looks at the human's hand**]
23. [**AP**][**facial gaze**] <*facial gaze*> "Ummmm, <*looks at the table*> [**looks at the table**] <*facial gaze*> good." [**MFG+**]
24. [**AP**]"Look, we made an <*looks at the table*> anchor." [**MFG+**]
25. [**AP**]"Thanks <*facial gaze*> for playing. That was fun, goodbye." [**MFG+**]
26. <*Looks at the table*> <*facial gaze*> <*looks at the table*> <*leaves the stool*>

D Questionnaire results

Table D.1: Questionnaire results using 2 tail, unpaired t-test

	Mean		Std. Dev.		
Question	**Oper.**	**Deg.**	**Oper.**	**Deg.**	**p-value**
1	6.2	6.4	1.1	0.93	0.588
2	4.8	4.4	1.1	1.0	0.385
3	6.1	5.9	1.6	1.7	0.824
4	5.7	5.1	1.4	1.5	0.297
5	7	6.7	0	0.61	0.092^{+}
6	6.9	6.9	0.3	0.3	1
7	6.1	5.9	1.1	1.1	0.727
8	6.6	6.4	1.3	0.6	0.723
9	4.2	3.3	1.6	1.4	0.121
10	4.6	4.7	1.2	1.6	0.788
11	3.4	3.5	1.4	1.5	0.896
12	6.4	6.1	0.85	0.95	0.409
13	5.9	5	1.1	1.2	0.049^{*}
14	6.6	6.5	1.1	1.3	0.759
15	4.2	3	1.4	1.6	0.038^{*}
16	6.6	6.7	1.1	0.6	0.831
17	3.6	2.4	1.1	1.3	0.017^{*}
18	6.2	6.6	1.3	0.6	0.281
19	1.8	1.6	1.8	0.8	0.794
20	5.7	5.6	1.2	1.4	0.774
21	4.6	3.9	1.5	1.5	0.206
22	6.6	6.9	0.93	0.4	0.429
23	5.6	4.6	1.2	1.7	0.085^{+}
24	6.2	6.2	0.97	0.89	1
25	5.9	4.4	1.1	1.4	0.003^{***}
26	5.4	5.8	1.8	1.3	0.552
27	6.4	6.5	1.2	0.8	0.848
28	6.2	6.5	1.1	1.0	0.487
29	20.1	19.6	1.3	1.6	0.377
32	3.1	3	1.7	2.1	0.842
System error in piece placement	2.6	2.2	2.5	2.2	0.695
System error in pointing recognition	0.3	0.7	0.6	0.8	0.131
Human error in piece placement	0.7	0.5	1.9	0.8	0.693
Total error	2.6	2.2	2.5	2.2	0.695
Completion Time (min)	5:45	6:17	1:36	1:14	0.338
Backchannels	1.1	0.8	1.4	0.8	0.522
Utterances	2.2	1.7	5.2	1.9	0.739

$^{+}$Data is trending, *Data is weakly significant, **Data is significant, ***Data is highly significant

Bibliography

[1] M. Argyle and M. Cook. *Gaze and mutual gaze*. Cambridge University Press, New York, 1976.

[2] D. Bohus and E. Horvitz. Learning to predict engagement with a spoken dialog in open-world settings. In *Proceedings of the SIGDIAL 2009 Conference*, pages 244–252, London, UK, September 2009. Association for Computational Linguistics.

[3] D. Bohus and E. Horvitz. Models for multiparty engagement in open-world dialog. In *Proceedings of the SIGDIAL 2009 Conference*, pages 225–234, London, UK, September 2009. Association for Computational Linguistics.

[4] S. Brennan. How conversation is shaped by visual and spoken evidence. In J. Trueswell and M. Tanenhaus, editors, *Approaches to Studying World-Situated Language Use*, pages 95–129. Cambridge, MA: MIT Press, 1999.

[5] D. Crystal. *The Cambridge Encyclopedia of Language*. Cambridge University, Cambridge, England, 1997.

[6] A. Dix. *Pace and interaction*. Cambridge University Press, 1992.

[7] S. Duncan. Some signals and rules for taking speaking turns in conversation. *Journal of Personality and Social Psychology*, 23(2):293–292, 1972.

[8] Felix Faber, Maren Bennewitz, Clemens Eppner, Attila Görög, Christoph Gonsior, Dominik Joho, Michael Schreiber, and Sven Behnke. The humanoid museum tour guide Robotinho. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 891–896, Toyama, Japan, September 2009.

[9] F. Flippo, A. Krebs, and I. Marsic. A framework for rapid development of multimodal interfaces. In *Proc. 5th Int. Conf. Multimodal Interfaces*, pages 109–116, Nov. 2003.

[10] B. J. Grosz and S. Kraus. Collaborative plans for complex group action. *Artificial Intelligence*, 86(2):269–357, 1996.

[11] A. Holroyd. Generating engagement behaviors in human-robot interaction. Master's thesis, Worcester Polytechnic Institute, Worcester, Mass., USA, December 2010.

[12] A. Holroyd, B. Ponsler, and P. Koakiettaveechai. Hand-eye coordination in a humanoid robot. Major qualifying project, Worcester Polytechnic Institute, Worcester, Mass., USA, 2009.

[13] A Holroyd., C. Rich, C. Sidner, and B. Ponsler. Generating connection events for human-robot collaboration. Submitted to ACM Conf. on Human-Robot Interaction, 2011.

[14] A. Kendon. Some functions of gaze direction in two person interaction. *Act Psychologica*, 26:22–63, 1986.

[15] M. Lombard, T. B. Ditton, D. Crane, B. Davis, G. Gil-Egui, K. Horvath, and J. Rossman. Measuring presence: A literature-based approach to the development of a standardized paper-and-pencil instrument. In *Presence 2000: The Third International Workshop on Presence, Netherlands*, Delft, The Netherlands, 2000.

[16] M. P. Michalowski, S. Sabanovic, and R. Simmons. A spatial model of engagement for a social robot. In *9th IEEE Workshop on Advanced Motion Control*, pages 229–240, 2006.

[17] L.-P. Morency, A. Rahami, and T. Darrell. Adaptive view-based appearance model. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 803–810, Madison, WI, June 2003.

[18] B. Mutlu, T. Shiwa, T. Kanda, H. Ishiguro, and N. Hagita. Footing in human-robot conversations: How robots might shape participant roles using gaze cues. In *Proc. ACM Conf. on Human-Robot Interaction*, San Diego, Calif., USA, 2009.

[19] C. Peters. Direction of attention perception for conversational initiation in virtual environments. In *Intelligent Virtual Agents*, pages 215–228, 2005.

[20] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Y. Ng. Ros: an open-source robot operating system. 2009.

[21] C. Rich, B. Ponsler, A. Holroyd, and C. Sidner. Recognizing engagement in human-robot interaction. In *Proc. ACM Conf. on Human-Robot Interaction*, Osaka, Japan, Mar. 2010.

[22] C. L. Sidner, C. D. Kidd, C. H. Lee, and N. Lesh. Where to look: a study of human-robot engagement. In *ACM International Conference on Intelligent User Interfaces (IUI)*, pages 78–84. ACM, 2004.

[23] C. L. Sidner, C. Lee, C. Kidd, N. Lesh, and C. Rich. Explorations in engagement for humans and robots. *Artificial Intelligence*, 166(1-2):104–164, 2005.