Toward Helpful Robot Teammates: A Simulation-Theoretic Approach for Inferring Mental States of Others

Jesse Gray, Cynthia Breazeal
MIT Media Lab
20 Ames St., Cambridge MA, 02139
jg, cynthiab @media.mit.edu

Abstract

As robots enter the human environment they must be able to communicate and cooperate with novice users. Towards this goal, understanding human nonverbal behavior is a critical skill. This includes not only recognizing human's actions, but also inferring mental states from observable behavior. This capability would allow a robot to offer predictive and relevant assistance to a human. Simulation Theory argues in favor of an embodied approach for how humans infer mental states of others (e.g., intents, beliefs, affect, etc.). This theory argues that humans reuse parts of their cognitive structure used for generating behavior to simulate and detect the behavior of others. Inspired by this theory, we describe our simulation-theoretic approach and implementation that enables a robot to monitor the human by simulating their behavior within the robot's own generative mechanisms on the motion, action, and perceptual levels. This grounds the robot's information about the user in the robot's own systems, allowing it to make inferences about the human's goals and knowledge that are immediately useful for providing helpful behavior such as helping to complete an action or pointing out an occluded object. We feel that designing individual systems of the robot to allow for this type of dual use, and reusing them in this manner, is a powerful technique for designing robots that interact with humans.

Introduction

Today, robots are largely considered to be complex tools operated by highly trained specialists; some, like assembly line robots, need initial programming to carry out a repetitive task, while others, such as planetary explorers, require the constant supervision of tele-operators. While requiring a trained operator to manage each robot may be feasible for these situations, as robots enter into the human environment, they will need to serve as helpers and teammates for untrained humans (care for the elderly, domestic assistant, etc.).

An important element of cooperative behavior is the need to understand a participant's actions and infer their goals from their external appearance and surrounding context. Experiments have show that human infants develop the ability to understand actions according to inferred goals as early as 6 months (Woodward, Sommerville, & Guajardo 2001). At 18 months they are able to imitate the goal of an unsuccessful action (Meltzoff 1995). Understanding the actions of others in terms of their goals is a natural level of representation of behavior in humans. For instance, 3 year old children have been shown to imitate actions based on inferred goals rather than perceived movements (Gleissner, Meltzoff, & Bekkering 2000).

Understanding actions in intentional terms through observation will enable the robot to see through surface level observable behavior to understand what the person is trying to do rather than just what they are physically doing at that very moment. This will allow the robot to provide assistance that is relevant to the person's goal, an important skill to participate in cooperative interactions with humans.

In order to approach the problem of action recognition and goal inference, we are using ideas drawn from the development of this capability in human children. Simulation theory holds that certain parts of the brain can have a 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. More specifically, to understand another person's mental process, we step into their shoes to determine the consequences. That is, we use our own similar brain structure to simulate the thoughts of the other person (Davies & Stone 1995).

In order to place oneself in the situation of another, it is critical to be able perceive situational information about another and use it as input into one's own cognitive architecture. Perceiving the position and motion of another person and mapping it onto your own body (skills required for imitating), then, become important requirements for applying these theories.

(Williams *et al.* 2001; Gallese & 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. These neurons show similar activity both when a primate carries out a goal directed action, and when it observes it performed by another (Rizzolatti *et al.* 1996; Gallese *et al.* 1996). (Barsalou *et al.* 2003) presents additional evidence 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.

This work provides a working model for how a simulation theoretic system could infer from external perception the goals and beliefs of others by utilizing its own action, motor production, and belief modeling systems. We believe that re-using these modules (and in the future, possibly others) is a useful technique to help understand the mental state of the human in terms useful to a robot. The work described here is of course situated within our own behavior architecture, however we believe that the idea of re-using these systems (and designing them with their potential re-use in mind) is an idea that could be applied to any similar system. Not only does a simulation theoretic design aid in the implementation of the system by allowing wide re-use of components, but it also grounds perceptions and inferences about the human in the robot's own representations, making it easier for the robot to perform useful behavior based on this data.

This paper first describes the techniques for the re-use of three separate systems, then describes the benefit of that re-use in the Demonstration section.

Platform

Robot This implementation runs on the Leonardo robot (Leo), a 63 degree of freedom humanoid robot. (figure 4)

Sensors Leo sees the world through two environmentally mounted stereo vision cameras and two eye cameras. One stereo camera is mounted behind Leo's head for detecting humans near Leo. The second stereo camera looks down from above, and detects objects in Leo's space as well as human hands pointing to these objects. Leo can use his eye cameras for fine corrections to look directly at objects. Leo also has access to data from the Axiom ffT facial feature tracker. The system provides data as the 2 dimensional image coordinates of various facial features, such as the inner corner of the eyebrow or the top of the mouth.

In order for Leonardo to perceive the pose and motions of a human interacting with him, we use a Gypsy motion capture suit. This suit has potentiometers mounted near the joints of the human to determine their joint angles. This suit is worn by the human interacting with Leo (figure 4). In the future, we plan to replace the use of this suit with a vision solution to allow for more natural interactions.

Background on Simulation Theory for Robots

Mapping demonstrated human movements into the robot's motor space has been discussed in Programming by Demonstration (PdB) applications (Kang & Ikeuchi 1997; Friedrich, Holle, & Dillmann 1998). PdB is used as a programming shortcut for quick, intuitive training of robots using human demonstration of tasks. This work focuses on motor skill acquisition rather than detection of known motor skills. (Lieberman & Breazeal 2004) explores mapping movements into the space of a robot and detecting boundaries between actions, but from the perspective of teaching new motor skills.

A number of motor learning efforts for robotic systems have looked to mirror neurons for their biological inspiration, ex (Schaal 1999); some even use simulated mirror neurons to recognize movements of both the robot and the human during interactions (Zukow-Goldring, Arbib, & Oztop 2002). These systems examine motor learning and motor trajectory recognition, but have not yet addressed using these abilities to infer hidden mental information about the human. (Buchsbaum 2004) uses a similar simulation theoretic action inference but for instrumental referencing.

There is a rich literature on plan recognition and using recognized plans to infer goals (see for example (Carberry 2001) for a review). Our system uses a simple hierarchical task structure similar to what is used in many plan recognitions systems; however we combine multiple types of information (motion, context, and a model of the human's beliefs) to infer the human's intent in a non-verbal task, then use this same task structure to determine helpful behavior for the robot to perform.

Architecture for Simulation Theory

Our implementation addresses mapping perceived body positions of another into the robot's own joint space, then using that information along with contextual clues to simulate the human's actions and infer their goals. Further, Leo can reuse his perceptual systems to model the beliefs of the human and provide helpful hints based on this information. See Fig. 2 for a system diagram. Each of these levels of inference operates independently, and can provide useful information on its own. Simple motor mapping can allow for imitation. Action and goal inference can help the robot monitor the human's success and provide help if needed. Belief modeling allows the robot to attempt to track what the human knows. These systems operating together, however, allow the robot to perform more complicated inferences. Combining motion matching (to help determine the human's action) with goal inference and belief modeling allows the robot to provide more sophisticated assistance such pointing out information unknown but necessary to the human and their current goals.

Movement Level Simulation

This section describes the process of using Leo's perceptions of the human's movements to determine which motion from the robot's repertoire might be being performed. The technique described here allows the joint angles of the human to be mapped to the geometry of the robot even if they have different morphologies, as long as the human has a consistent sense of how the mapping should be and is willing to go through a quick, biologically inspired calibration process. Once the perceived data is in the joint space of the robot, the robot tries to match the movement of the human to one of its own movements; representing the human's movements as one of the robot's movements is more useful for further inference than a collection of joint angles.

The motor system of the robot is based on a pose-graph structure (Blumberg *et al.* 2002), consisting of a directed graph of poses. Each pose is a collection of joint angles that represents one frame of an animation. Paths through this directed graph of poses form the motions that the robot can produce.

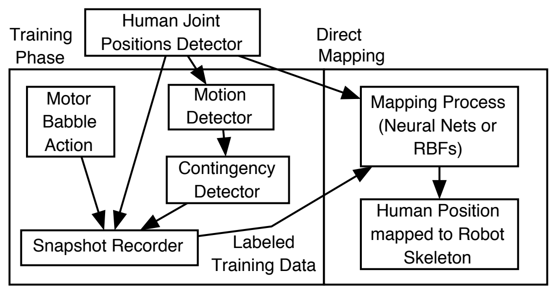

Figure 1: Mapping perceived human joints onto the robot's skeleton to allow for comparison between joint configurations of robot and human

Intermodal Mapping The mechanism described here to discover the mapping from perceptions about the human's position into the joint space of the robot is inspired by Meltzoff's Active Intermodal Mapping (AIM) hypothesis (Meltzoff & Moore 1997) which addresses the mapping process for the facial imitation in infants by proposing that they share a common representation for the perception and production of motor movements. We use the joint space of the robot as this representation and thus need no conversion to output motor commands. However, incoming sensor data of the robot must be converted to this format. The AIM hypothesis provides inspiration for this step, by theorizing that infants refine a similar mapping through an imitative interaction with a caregiver. In this interaction the infant motor babbles, or simply tries out new motor configurations. When the caregiver imitates the infant, the infant observes this and uses that imitation as a mirror to refine its own mapping from perception of facial expressions to their production.

Facial Mapping The robot's face perception comes from the Axiom ffT system, which tracks facial features and outputs their position as 2d coordinates. These 2d coordinates must be mapped onto the robot's joint space. This process is complicated by the fact that there is not a one to one correspondence between the tracked facial features and Leo's joints. A number of algorithms exist that can be applied to map data from one space to another given sufficient labeled training examples to train a model. To acquire this training data, the robot participates in an imitative interaction with the human.

Within the framework of the imitative interaction, the robot identifies when the human participant is imitating it, and then stores a snapshot of the current facial feature data and the robot's own current joint configuration. Before the mapping is trained the robot cannot detect an exact correspondence between the human's facial features and its own pose, thus identifying when the robot is being imitated is tricky at this stage. However, a good metric is provided by the timing of raw movement in the perceptual data; if the perceptual data exhibits significant movement right after the robot moves its joints, it is likely that the human is imitating, and a snapshot can be taken.

Once enough data has been acquired, a mapping between the perceived feature coordinates and the robot's joint space can be trained. In this case we found a simple 2 layer neural net was sufficient to approximate the function. Separating the input data based on the different parts of the face allowed us to train a separate network for each part of the face. This greatly reduced the number of examples needed to train an effective mapping, since each network is responsible for fewer independent degrees of freedom. We found that in this case, about 2 examples per pose and about 4 poses per facial feature (i.e. left eye, right eye, mouth) were sufficient. Once these networks are trained, new perceived facial coordinate data can be mapped into the robot's pose space, enabling the robot to imitate the human. For a more complete treatment of this facial mapping process see (Breazeal *et al.* 2005).

Body Mapping A similar technique is employed in mapping perceived arm and torso movement into the robot's joint space. In this case, joint positions of the human are observed using a mechanical motion capture suit described above. The robot motor babbles in different regions (each arm and the torso) with the human imitating. When enough poses have been recorded, a mapping is created (in this case using radial basis functions), and the robot can mimic the movements of the human.

Matching Joint Angles to Trajectories The above sections describe how to directly map from observations to the robot's own joint space, which can be useful for mimicry, but is somewhat limited in its usefulness for drawing inferences. To gain more representational power, we match this joint angle data against the trajectories through the robot's pose-graph, and thus determine which of the robot's motions is most closely related to the motion the human is producing. This is a much more concise representation of the data than sets of joint angles, and it will prove useful in the following section.

Early work for this system classified incoming trajectories using a domain specific metric that worked well with the limited number of animations the robot was using; however, we are moving towards a more general system based on Morphable Models (Brooks, A.G. *et al.* 2005), which are designed for comparison of spatio-temporal data such as these joint trajectories.

Goal Level Simulation

The action representation described here is intended to not only allow the robot to perform a set of actions, but also introspect over them to determine which actions a human is performing and even infer certain mental states (most notably their goal, based on what the robot's goal would be if it were performing the action in their position). Ideally the same information is needed to detect actions and make inferences as to perform them, so when actions are designed (and in the future, learned) only accomplishing the forward task need be taken into account, and the detection and simulation will come for free. This design goal is followed except for one concession, which comes in determining which object a human is acting upon - this mechanism requires a special detector (a Parameter Generation Module, or PGM) described below.

Action Structure The robot's actions are composed of a hierarchy of Action Segments and Conditions. Action Segment modules perform a task (intended to be atomic, on the level of a single motion), while Condition modules make an evaluation about some state of the world or the creature's internal state. A Condition that follows an Action Segment is a goal of that segment - it is expected that after the completion of that segment, the Condition will evaluate to true. A Condition that precedes an Action Segment is its precondition - it is necessary that that Condition evaluate to true before the action is attempted (See Figure 3).

This relationship between Action Segments and Conditions allows larger structures to be created where the goal of one Action Segment becomes the precondition of a later Action Segment. Compound tasks are specified as a hierarchy of Actions Segments and Conditions. To achieve some desired state, only the desired Condition need be activated, and the system will traverse the hierarchy completing preconditions and Action Segments as necessary until the Condition is fulfilled (or it becomes known that the Condition cannot be fulfilled).

Conditions have two modes of operation. If queried, they simply determine if they have been satisfied ("Is the button on?"). They can also be activated, in which case they try to achieve themselves. In this case, the condition will activate each of its action segments in turn until it is achieved, or until it runs out of segments and still is not achieved. A standard condition relates to some state of the world; it uses the belief system of the robot to determine if it is satisfied (see the next section for a brief description of the robot's belief system).

Action Segments contain a list of preconditions. When an Action Segment is activated, it first verifies that all its preconditions are met. Any Conditions that are not met are also activated. If the Action Segment can achieve all of its preconditions by activating them, it will do so and then activate itself. If any of its preconditions fail to complete, the Action Segment will terminate in failure and it will not attempt to perform itself.

The actual inner behavior invoked by an Action Segment structure is often a single motion performed by the robot, although it could be something more sophisticated. Oftentimes these motions have additional parameters, such as where or on what object the robot should perform the motion. The Action Segment gets this information in forward operation from a higher level decision process (not covered here); when observing a human's action this information is not directly available in the same way, so these parameters come from an associated PGM (see next section).

Simulating the actions of others to infer their goals The system for simulating the actions of others in order to infer their goals begins by first attempting to determine which action segment they are performing. One of the key inputs into this system is the motion classifier described in a previous section. Whenever the robot classifies a motion of the human as one of its own, it triggers a search through its set of action segments to see what action segments might produce that motion (Each action segment can be associated with a movement it produces - in the current architecture, only actions with an associated movement can be recognized). The set of matching action segments are further narrowed by examining the context required to perform each one. The context refers to the necessary conditions in the world state for the action to be performed - for example, in order push a button, there must be a button nearby. For the robot, the context required for an action segment is represented as the set of parameters necessary to perform that segment, as described below.

In normal operation, when the robot decides to accomplish some condition, that decision process will supply any parameters needed for that condition (such as which object to act on). However, when detecting the action in an observed human, the robot is not privy to that hidden information. Instead, each action is provided with a PGM that attempts to generate the relevant parameters. The most common module is one that detects objects that are close to the human's hand. For example, if the robot is determining which button a human is attempting to press, it will guess that it is the button closest to the human hand. This module works by examining data from the robot's belief system, where the position of the human and other objects in the world are tracked. These modules can also indicate a failure to generate the appropriate parameters, for example if there is no button near the human's hand. This failure indicates that the context for the given action segment is invalid; other action segments with similar motions will be considered and if none have an appropriate context there will be no match for the human's current activity.

For an action segment to be eligible for consideration, it must produce the motion currently being observed, and also have currently relevant parameters as determined by its PGM. Once an action has been selected, the robot infers that the human's goal would be the same as its own, were it currently performing the action. Since it has mechanisms to determine its own success or failure (by evaluating the goal condition for that action segment), it can monitor the success of the human. Currently the robot helps out by re-performing the action of the human if they fail (as it is easily able to do, since the structure used to detect the action is the same used by the robot to perform the action - also any relevant action parameters have been filled in by the PGM). In the case of a compound action, it also examines the preconditions of the action segment that the human is attempting. If a precondition is not satisfied, the robot determines that this might be the reason the human has failed and tries to accomplish the precondition for them. These are relatively simple responses to these inferred states, but in the future these inferences could be used to inform more sophisticated behavior.

Belief Level Simulation

We believe that for successful cooperative interaction modeling the human's beliefs about the task state is important. For example, a human may not be able to see important information due to occlusion, or a new arrival may need to be brought up to speed.

Leo keeps track of information about the world using his

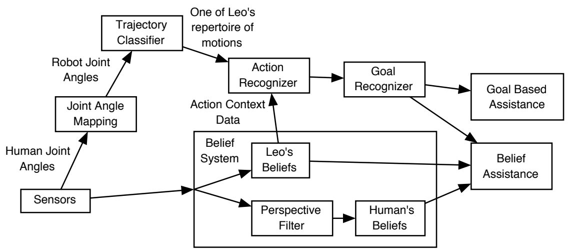

Figure 2: System Architecture Overview - Inferring beliefs allows for additional kinds of assistance over only inferring actions and goals.

Belief System (Blumberg *et al.* 2002). Data comes in from Leo's sensors, and passes through his perception system. In the belief system, the robot sorts and merges data from the perception system to attempt to determine what data corresponds to new information about known objects, and what data indicates the presence of a new object. The object representation consists of a history of properties related to each perceived object.

In the spirit of simulation theory, we reuse that mechanism to model the beliefs of the human participant. For this, the robot keeps a second set of beliefs that are produced through the same mechanism - however, these beliefs are processed from the visual perspective of the human. This is different from simply flagging the robot's beliefs as human visible or not, because it reuses the entire architecture which has mechanisms for object permanence, history of properties, etc. By reusing the entire system you can get more complicated modeling: for example, this approach can keep track of the human's incorrect beliefs about objects that have changed state while out of their view.

This technique has the advantage of reusing a lot of architecture, but also of keeping the model of the human's beliefs in the same format as the robot's. This is important for using the information, because all the mechanisms the robot uses with its own beliefs can now be applied, in simulation, to work with the model of the human's beliefs. Currently we use exactly the same processing to produce the human's beliefs as the robot's, except that we apply a visual perspective filter for the human's beliefs based on what they should and shouldn't be able to see - this allows the robot to keep a model of what the human believes about the local world. Further changes could easily be made to the human's perceptual system - for example, if the robot believed that the human was colorblind, it could remove the color percept from its processing of the human perceptual information, and then none of the human beliefs would contain any color information. You could even imagine modeling what the robot thinks the human thinks the robot knows using this mechanism, by maintaining a third set of beliefs that go through the humans filter as well as one additional filter.

This model of the human's beliefs is used by the robot to determine how best to help the human accomplish tasks. In order to do this, the robot first uses the mechanism described above to determine what action segment the human is performing. It then determines if the human has enough information to successfully complete the action, and if the human does not it points out the relevant information to the human in an attempt to get them to discover it. For example, if an object is hidden behind an occlusion, the robot can point to it in an attempt to get the human to look around the obstruction and discover the object.

As described above, the robot has a mechanism to help the human by re-performing a failed action, or by performing another section of a compound action. To do this it uses its PGM to determine the appropriate parameters (generally a target object) for repeating a human's action. This same mechanism is used in this context to determine if the human has enough information. For each necessary action segment that is part of the compound action the human is performing, Leo attempts to generate the appropriate parameters to perform that action using the action's PGM(without actually performing the action). However, instead of using its own beliefs as input to the PGM, the system attempts to generate the parameters based on its model of the human's beliefs. Thus, the robot is trying to see if it could accomplish each action segment with the beliefs that the human has. So for example, if as part of a compound action "Button A" must be pressed, the action segment's PGM will fail if no "Button A" is present. If the human has no belief for "Button A", then when the robot puts itself in the position of the human and tries to determine if it could complete the action, it will determine that it could not, and that the action segment would fail. The robot can then point out the object to human and try to get them to learn about it.

The mechanism described here for belief modeling does come with significant scalability limitations. Maintaining the entire set of beliefs of the participant is costly to the robot's behavior system, and becomes infeasible with more than a few humans. To handle this, a good solution might be to attend fully to only one or two humans at a time, and only keep a full model of the humans that are currently being attended to.

Demonstration

To demonstrate the concepts described here, we have created an example action system for the robot that can show off many of the ideas discussed here. The structure of its actions can be seen in figure 3. The actions are somewhat contrived because they are designed to use existing button pushing skills of the robot, but they allow demonstration of the key points.

Motion Classification

Although there is no specific way to observe the output of the motion classification system in this scenario, it is running all the time in these demonstrations. The motion classification is a critical piece of the data that allows the next two demonstration sections to function properly.

Action/Goal Inference

The robot's action and goal inference is shown here when the human attempts to push one of the buttons. If the human per-

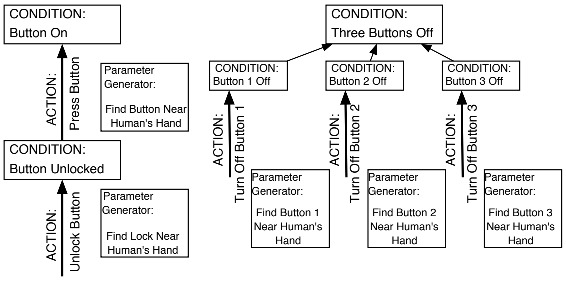

Figure 3: Structure of Leo's Action Segments and Conditions for this demonstration. In forward operation, Leo (recursively) performs each connected action segment until a desired condition is satisfied. Leo detects a human doing one of his action segments if they are performing that trajectory and the parameter generator successfully finds appropriate action parameters (i.e., context is correct).

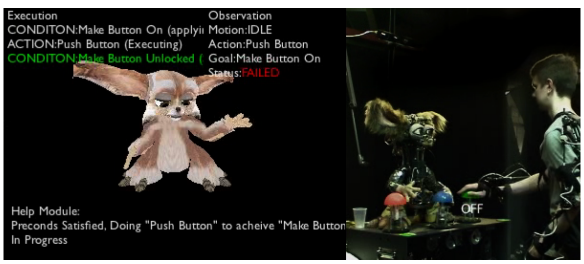

Figure 4: Human fails to completely press button. Leo notices and starts to achieve human's goal, turning the button on.

forms a pushing motion (detected through the motion classification system) and their hand is near a button (detected through the parameter estimation system), the robot determines that they are trying to push that button. Based on the structure of its own action, it infers that they are trying to accomplish the goal of turning the button on. Further it evaluates their success by monitoring the state of the button, and if they fail, it will accomplish their goal for them.

In the case of a locked button, Leo demonstrates that he can perform a helpful action that is not simply repeating the action of the human. In this case, if they perform a push motion near a locked button, Leo still infers that their goal is to turn on that button. However, if that button is indeed locked, instead of repeating their action (and also failing, as the button is locked), he achieves the precondition which the human overlooked, allowing them to proceed to their goal and demonstrating that he is goal oriented rather than action oriented.

Belief Inference

To demonstrate Leonardo's belief inference capabilities we move to a simulated task since we can more readily manipulate the environment by adding and removing occlusions that Leo can easily detect with simulated data. For this demonstration, we add an occlusion to the simulated environment of the robot that blocks the human's view of one of the buttons. Leo is keeping track of the human's beliefs through the mechanism described above, and because of his visual perspective taking his model of the human's beliefs is lacking one button. When the human begins the Turn All Buttons Off task, Leo notices the actions using the above mechanism. He considers each required action segment and determines if the human has enough information. Noticing that the human cannot complete the compound action because of the obscured button, Leo performs an action to help them acquire the missing information - in this case, pointing at the button to encourage the human to look around the occlusion.

Discussion

The system described here was built in an architecture that evolved from c4 (Blumberg *et al.* 2002). In many ways this architecture is modular by design, separating behavior processing into discrete pieces such as gathering perceptual input, belief maintenance, action selection, and motor control. However the modular concept being discussed here is the idea of Simulation Theory, and how it can be applied to existing systems to add mental state inference capabilities. To a certain extent the parts of the system to be reused in this manner must be designed with this re-use in mind - they must be modular enough that they can be isolated and reused, and they must have a data representation that supports enough introspection to work backwards from results to cause. However, we believe that this idea can be applied to other behavior architectures to help begin to model human mental states.

We have been pleased with the success of employing these simulation theoretic ideas within our architecture. Not only have the techniques been useful at accomplishing our stated tasks, but we keep finding that we get additional beneficial behavior "for free" because of these design decisions. The idea of simulation theory may or may not hold up as we gain more evidence about the functioning of the human brain, but it does seem to be a useful tool for the design of robotic behavior.

References

Barsalou, L. W.; Niedenthal, P. M.; Barbey, A.; and Ruppert, J. 2003. Social embodiment. *The Psychology of Learning and Motivation* 43.

Blumberg, B.; Downie, M.; Ivanov, Y.; Berlin, M.; Johnson, M. P.; and Tomlinson, B. 2002. Integrated learning for interactive synthetic characters. *ACM Transactions on Graphics* 21(3: Proceedings of ACM SIGGRAPH 2002).

Breazeal, C.; Buchsbaum, D.; Gray, J.; and Blumberg, B. 2005. Learning from and about others: Toward using imitation to bootstrap the social competence of robots. *Artificial Life* 11.

Brooks, A.G.; Berlin, M.; Gray, J.; and Breazeal, C. 2005. Untethered robotic play for repetitive physical tasks. In *ACM International Conference on Advances in Computer Entertainment*. (Submitted, under review).

Buchsbaum, D. 2004. Imitation and social learning for synthetic characters. Master's thesis, MIT.

Carberry, S. 2001. Techniques for plan recognition. *User Modeling and User-Adapted Interaction* 11(1-2):31–48.

Davies, M., and Stone, T. 1995. Introduction. In Davies, M., and Stone, T., eds., *Folk Psychology: The Theory of Mind Debate*. Cambridge: Blackwell.

Friedrich, H.; Holle, J.; and Dillmann, R. 1998. Interactive generation of flexible robot programs. In *Proceedings of the IEEE International Conference on Robotics and Automation*.

Gallese, V., and Goldman, A. 1998. Mirror neurons and the simulation theory of mind-reading. *Trends in Cognitive Sciences* 2(12):493–501.

Gallese, V.; Fadiga, L.; Fogassi, L.; and Rizzolatti, G. 1996. Action recognition in the premotor cortex. *Brain* 119:593–609.

Gleissner, B.; Meltzoff, A. N.; and Bekkering, H. 2000. Children's coding of human action: cognitive factors influencing imitation in 3-year-olds. *Developmental Science* 3(4):405–414.

Kang, S. B., and Ikeuchi, K. 1997. Toward automatic robot instruction from perception - mapping human grasps to manipulator grasps. *IEEE Transactions on Robotics and Automation* 13(1).

Lieberman, J., and Breazeal, C. 2004. Improvements on action parsing and action interpolation for learning through demonstration. In *Proceedings of the International Conference on Humanoid Robots*.

Meltzoff, A. N., and Moore, M. K. 1997. Explaining facial imitation: a theoretical model. *Early Development and Parenting* 6:179–192.

Meltzoff, A. N. 1995. Understanding the intentions of others: re-enactment of intended acts by 18-month-old children. *Developmental Psychology* 31:838–850.

Rizzolatti, G.; Fadiga, L.; Gallese, V.; and Fogassi, L. 1996. Premotor cortex and the recognition of motor actions. *Cognitive Brain Research* 3:131–141.

Schaal, S. 1999. Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences* 3:233–242.

Williams, J.; Whiten, A.; Suddendorf, T.; and Perrett, D. 2001. Imitation, mirror neurons and autism. *Neuroscience and Biobehavioral Reviews* 25:285–295.

Woodward, A. L.; Sommerville, J. A.; and Guajardo, J. J. 2001. How infants make sense of intentional actions. In Malle, B.; Moses, L.; and Baldwin, D., eds., *Intentions and Intentionality: Foundations of Social Cognition*. Cambridge, MA: MIT Press. chapter 7, 149–169.

Zukow-Goldring, P.; Arbib, M. A.; and Oztop, E. 2002. Language and the mirror system: A perception/action based approach to cognitive development. *Working draft*.