Robots that Stereotype: Creating and Using Categories of People for Human-Robot Interaction

Alan R. Wagner
Georgia Tech Research Institute, Atlanta GA

Psychologists note that humans use categories to simplify and speed the process of person perception. Applications that require a robot to interact with a variety of different people will demand the creation of stereotypes representing different categories of people. This article presents a method for stereotype learning and usage by a robot. Both robot and simulation experiments are used to examine the benefits and challenges associated with stereotype usage. Our results indicate that stereotypes can serve as a means of feature selection and for inferring a partner's appearance from observations of their actions. The results also show that the timing of certain types of errors impact the stereotype creation. This article concludes by describing the limitations, ethical ramifications, and potential applications of this research.

Keywords: Human-robot interaction, stereotype, user model, recognition, concept learning

1. Introduction

A variety of research has come to the conclusion that creating mental models of humans is critical for behavior prediction (Rilling, et al., 2002; Trafton, et al., 2005; Briggs & Scheutz, 2011). For instance, one's model of an interactive partner has been found to play a larger role in behavior prediction than facial expressions (Todorov, Gobbini, Evans, & Haxby, 2007). Creating partner models may therefore be necessary for developing robots that can act as a teammate or assistant.

This article examines possibility of using ideas from the psychology of stereotyping to develop methods that would allow a robot create categories of people. Psychologists note that humans regularly use categories to simplify and speed the process of person perception (Schneider, 2004). Macrae and Bodenhausen suggest that categorical thinking influences a human's evaluations, impressions, and recollections of the target person (Macrae & Bodenhausen, 2000). The influence of categorical thinking on interpersonal expectations is commonly referred to as a stereotype. For better and for worse, stereotypes have a profound impact on interpersonal interaction (Bargh, Chen, & Burrows, 1996). The research question investigated here is whether and how stereotypes can be used to aid the process of modeling a robot's interactive partner.

This question is potentially critical for robots operating in dynamic social environments, such as search and rescue. In environments such as these, a person's uniform often reflects his or her role. Ideally, a general purpose robotic teammate would leverage its perception of a person's uniform with previously obtained role-specific knowledge to quickly evaluate how it could help. For people, learned categories of different types of individuals bootstrap the process of modeling a newly encountered person and provide knowledge about how to initially interact. Information processing models of human cognition suggest that the formation and use of stereotypes is critical for this type of quick assessment of new interactive partners (Macrae & Bodenhausen, 2000). Our goal is to understand if this approach can similarly assist robots which need to quickly assess the needs of their human teammate.

Journal of Human-Robot Interaction, Vol. X, No. X, 20XX, Pages XX-XX, DOI 10.5898/JHRIX.X.XX

Our algorithm for stereotyping is not complex. Still several aspects of this work are new, important, and powerful. First, although the computational methods used in our algorithm are not original, their application to the learning of categories of people by a robot is novel. The creation of categorical models of people is an important step towards developing robots that can use prior experience with similar individuals to reason about newly encountered people. Moreover, the methods described are powerful in that the stereotypes that are created reflect the robot's underlying interactive experience, not a programmer's preconceived classification scheme. The methods are also extremely general. We have applied these techniques to several types of robots and social environments. Finally, and perhaps most importantly, the computational steps that we use to create the robot's stereotypes are grounded on a larger and diverse literature ranging from social psychology, to cognitive science, to game theory. We therefore believe that any general purpose technique for stereotyping by a robot will have similar computational underpinnings.

This article represents a large effort to develop and test methods by which a robot could create and use stereotypes to bootstrap the process of interacting with a new person. Previously published related work details the development of this algorithm and a portion of the experiments (Wagner A. R., 2012a). In addition to dozens of simulation experiments, to date over 4.5 hours of filmed experiments have been conducted. Video clips for these experiments can be found at: www.cc.gatech.edu/~alanwags/Stereotypes/indexStereotypes.html.

Before presenting these experiments, we review related work. Next, the problem of partner modeling is briefly described and this problem is couched with respect to our framework for social action selection. A description of the different types of stereotyping error and their impact on behavior prediction follows. The experimental methods used, the experiments conducted, and the results of those experiments are then discussed. This paper concludes by examining the assumptions, limitations, and ethical ramifications of this work.

2. Related Work

Stereotypes and stereotyping have long been a topic of investigation for social psychologists (Terveen, 1994). Schneider provides a thorough review of the existing research in psychology (Schneider, 2004). Numerous definitions of the term stereotype exist. McCauley et al. state that stereotypes are, "those generalizations about a class of people that distinguish that class from others" (McCauley, Stitt, & Segal, 1980, p. 197). Similarly, Andersen et al. notes that stereotypes are "highly organized social categories that have properties of cognitive schema" (Anderson, Klatzky, & Murray, 1990, p. 192). Finally, Edwards defines a stereotype as "a stimulus which arouses standardized preconceptions which are influential in determining one's response to the stimulus" (Edwards, 1940, pp. 357-358). These definitions highlight several agreed upon characteristics related to stereotypes. Namely, that stereotypes 1) are derived from one's interactions and perceptions with members of a group; 2) generalize individuals into categories; and 3) provide information about new individuals. Smith and Zarate describe three general conceptual classes of stereotypes: attribute-based, schematic-based, and exemplars (Smith & Zarate, 1992). Attribute-based models of stereotyping focus on the attributes of the stimulus person perceived and the ways that these attributes are used to make social judgments. The process of attribute-based stereotyping thus typically involves first searching for and identifying a fixed set of attributes and using these attributes to make predictive decisions about a novel person. Exemplars, on the other hand, utilize specific examples with previous individuals as predictive models of future interaction with perceptually similar individuals. Schematic-based models of stereotyping, in contrast, focus on the creation of a category prototype and then matching of the newly encountered person to the correct category. Knowledge contained within the categories' prototype is used to make social judgments about the person. This article presents algorithms based on a schematic or prototype conceptual model of stereotyping. As such, the result of our algorithm is a prototype that is used by the robot to reason about newly encountered people.

With respect to computer science, the inclusion of techniques for stereotyping is not new. Human Computer Interaction (HCI) researchers have long used categories and stereotypes of users to influence aspects of user interface design (Rich, 1979; Kass & Finin, 1991). The multi-agent systems community has also explored the use of stereotypes. Ballim and Wilks use stereotypes to generate belief models of other agents (Ballim & Wilks, 1991). Denzinger and Hamdan develop a system by which an agent is tentatively stereotyped, then, after interacting with the target for a period of time, stereotype switching may occur (Denzinger & Hamdan, 2006). Their results indicate that stereotype switching improves behavior predictions when the originally selected stereotype is incorrect. Burnett uses stereotypes to gauge an agent's trustworthiness (Burnett, Norman, & Sycara, 2010).

Investigations of stereotyping with respect to robots are comparatively scarce. Fong et al. used predefined categories of users in conjunction with a human-robot collaboration task (Fong, Thorpe, & Baur, 2001). These categories influenced the robot's dialogue, actions, the information presented to the human, and the types of control afforded to the user. Duffy presents a framework for social embodiment in mobile autonomous systems (Duffy, 2004). His framework includes methods for representing and reasoning about stereotypes. He notes that stereotypes serve the purpose of bootstrapping the evaluation of another agent and that the perceptual features of the agent being stereotyped are an important representational consideration. The work presented here differs from this previous work in that we **do not rely on predefined categories** and our work has been developed and tested on a robot which is interacting with a human.

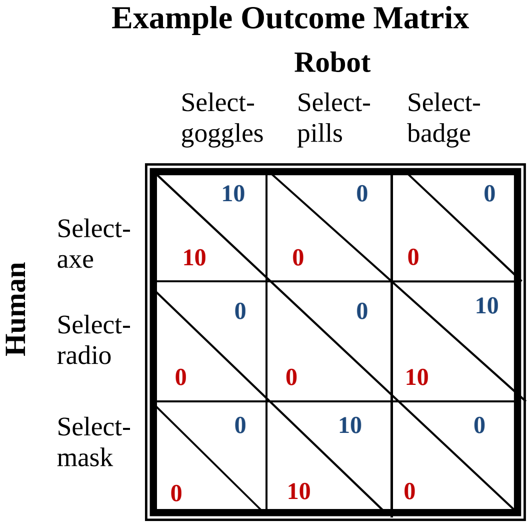
3. Partner Modeling

A stereotype should provide information about newly encountered individuals (Edwards, 1940; Macrae & Bodenhausen, 2000). We use the term *partner model* (denoted m^{-i}) describes a robot's mental model of its interactive human partner. Our work uses the normal-form game or outcome matrix to represent interactions (Figure 1). The outcome matrix is a standard computational representation for interaction and has long been used in social psychology and game theory (Osborne & Rubinstein, 1994; Kelley & Thibaut, 1978). This representation informs us as to what information, at a minimum, must be maintained within the partner model itself. A partner model must contain at least three types of information: 1) a set of partner features $F = \left(f_1^{-i}, \dots, f_n^{-i}\right)$; 2) an action model, A^{-i}; and 3) a utility function u^{-i}. Partner features are perceptual features used for partner recognition. These features allow the robot to recognize the person in subsequent interactions. The action model contains a list of actions available to the partner during the interaction. The utility function includes information related to the partner's change in state (represented by an outcome value) resulting from the selection of a pair of actions by the dyad. Information about the partner's beliefs, knowledge, personality, etc. could also conceivably be included in these models. The superscript $-i$ is used to express individual *i*'s partner (Osborne & Rubinstein, 1994). Thus, for example, A^i denotes the action set of individual *i* and A^{-i} denotes the action set of individual i's partner.

This version of a partner model informs our approach to creating stereotypes. We have argued in prior work that this method of representing a partner is both general and flexible (Wagner A. R., 2009a). Further, our approach to stereotyping can be used by other versions of partner models so long as a distance metric between models is defined and a process for merging models exists.

3.1. A Process for Partner Model Creation

But how does a robot learn a partner model? One simplistic method for learning a model of a partner is to just interact with the person and observe and store their features, action selections, and their change in emotional state after the interaction (represented by outcome). Formally, each interaction provides information in the form $\langle a^i, a^{-i}, o^i, o^{-i} \rangle$. The action selected by the robot, a^i, is known. The action selected by the human, a^{-i}, requires some degree of activity or action recognition on the part of the robot. Activity recognition is an area of active research in which great progress is being made (Aggarwal & Ryoo, 2011). The outcome received by the robot, o^i, relates to the reward and cost associated with the selection of an action pair for the robot. This reward may be externally or internally generated based on the robot's motivations. Finally, the outcome received by the partner, o^{-i}, describes the rewards and costs associated with the selection of the action pair for the partner as witnessed by the robot. In other words this term represents the change in the person's internal (emotional, behavioral or physical) state that results from the interaction. Ongoing research is also exploring methods for assessing the value of this term (Mower, Mataric, & Narayanan, 2011; Fasel & Luettin, 2003). In previous work we have shown that this method could eventually be used to learn a model of a robot's interactive partner assuming that the partner's model was static (Wagner A. R., 2009a). One problem is that this method requires a large number of interactions with each partner and may not be possible if the partner's action model is dynamically changing. Hence, we were motivated to develop an algorithm that would allow a robot to use its previously learned partner models as a means for assessing a new but unknown individual.



Example Outcome Matrix

Robot

Human

	Select-goggles	Select-pills	Select-badge
Select-axe	10 10	0 0	0 0
Select-radio	0 0	0 0	10 10
Select-mask	0 0	10 10	0 0

Figure 1. This outcome matrix represents a coordination game in which the robot and the human only receive a positive outcome if they select complimentary objects.

Yet, if a robot retrieves a previously learned partner model from memory and uses this model to guide its interactions with a new person, undoubtedly the retrieved model, m^{-i}, will differ from the new person's actual model, denoted ${}^{*}m^{-i}$. Still, the robot's interactions with the partner can be used to inform the robot as to the differences. The robot can then adjust its current model of the partner to match its observations. Adjusting the model requires either:

1. Adding unexpected actions to A^{-i};
2. Changing the outcome values resulting from (a^{i}, a^{-i}) ;
3. Removing unused actions from A^{-i} and their related outcomes;

The removal of unused actions is achieved by using the partner's action selection history to calculate the probability of using the action. The probability that the partner will use an action during an interaction is calculated as,

$$p(a^{-i}) = \begin{cases} 1 & if\ a^{-i}\ is\ selected \\ \frac{|A^{-i}|-1}{|A^{-i}|} \cdot p(a^{-i}) & if\ a^{-i}\ is\ not\ selected \end{cases}, \qquad (1)$$

All actions below a predefined threshold are removed. The threshold was 0.49 for all experiments. This threshold was empirically derived

The preceding discussion raises an important question: how can partner models be compared to one another? For example, how close is the partner model, m^{-i}, that the robot learned during several interactions with a human to the actual model, ${}^{*}m^{-i}$, that the person was actually using to select actions? We address this problem by viewing action models and utility functions as discretized sets. The action model is a set of actions and a utility function is a set of triplets, $\langle a^i, a^{-i}, o \in \Re \rangle$, containing the action of each individual and a resulting utility value. If the contents of a partner model are viewed as the elements of a set, then the use of set theoretic measures of distance to compare different partner models is possible. The Jaccard index,

$$J(m^A, m^B) = \frac{|m^A \cup m^B| - |m^A \cap m^B|}{|m^A \cup m^B|}, \qquad (2)$$

is one measure of set distance (Jaccard, 1901). Elements from the utility function were considered to be equal if the actions were the same and the outcome values were within 1 of each other. Because of its simplicity, this measure of distance is an attractive option for measuring distance between partner models. Nevertheless, one of the shortcomings of this measure is the fact that actions and outcomes play an equal role in the determination of distance.

Stereotyping allows a robot to retrieve a partner model that can serve as an initial source of predictive information. The section that follows presents our algorithm for learning and using stereotypes.

4. Stereotyped Partner Models

Definitions of stereotyping indicate that the process generalizes and categorizes individuals based on perceptual observations of the individual (Schneider, 2004; McCauley, Stitt, & Segal, 1980; Anderson, Klatzky, & Murray, 1990). Prototype models of stereotyping, in particular, argue for the creation of a category prototype generated from perceptual observations and used to make social judgments. We therefore contend that any algorithm for prototype-based stereotyping that is derived from the psychological evidence will thus require, at minimum, a step devoted to the creation of these categories and a step devoted to matching a new individual's perceptual features to the categories. Our algorithm has been developed with these steps in mind. This article expands upon research presented in recently published work (Wagner A. R., 2012a; Wagner A. R., 2012b) and represents a more thorough treatment of the original work.

4.1. Creating Stereotypes

The **create stereotypes algorithm** (Figure 2 top) takes as input a new partner model. This input is optional. The algorithm can also be run on the robot's existing history of partner models (termed the model space). As discussed in section 3.1, individual partner models are learned by successively interacting with an individual and updating a model with the results from the interaction.

Initially the robot has no partner models at all in its model space. The robot seeds its model space with a model of itself, its self-model. Conceptually, this act of seeding with the self-model allows the robot to equate its partner's actions and preferences to its own actions and preferences. Our implementation of the algorithm is capable of saving any partner models that the robot has learned as serialized data. Hence, because the robot could always load its previously saved partner models, seeding of the model space was only necessary at the beginning of an experiment.

Once a model of a new partner has been learned, the first step of the **create stereotypes algorithm** adds the new model to the model space. Next, in lines 2 and 3, each model in the space is assigned to a unique cluster. Lines 4 and 5 perform agglomerative clustering, iterating through each cluster and, if the distance between two clusters is less than a predetermined threshold, merging them. Equation (2) is used to determine the distance between two clusters. Convergence is achieved when there are no clusters below the threshold. The cluster centroids that remain after step five are stereotypes, denoted $s_1, \ldots, s_n$. This list of stereotype models is saved by the robot.

In the next phase, a decision tree, denoted ψ, is created mapping from the partner's perceptual features to stereotypes. Lines 6 and 7 from Figure 2 (top) create data for the learning algorithm by pairing each model's perceptual features to its associated stereotype. In the final steps, this data is used to train a classifier mapping partner features to the stereotyped model.

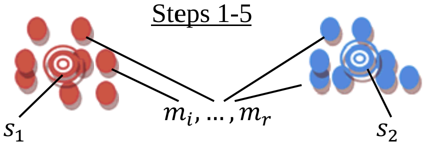
Create Stereotypes Algorithm

```
Input:    Partner model $m^{-i}$.
Output: Classifier $\psi$ mapping $m^{-i}$(features)
 to a stereotype.

Cluster phase
1. Add $m^{-i}$ to partner model space
2. for all models in the model space
3.     make a cluster $c_j$
4. while closest-centroid-distance $(c_j, c_q) < k$
5.     merge-clusters$(c_j, c_q)$

Function learning phase
6. for all models in model space
7.     set data[j]←make-pair($m_j$.(features),  centroid$_j$)
8  $\psi$ ←train-classifier( data  )
9. return $\psi$
```

Steps 1-5



Steps 6-7

Create Data $\begin{bmatrix} m_1: f_1, \dots, f_n & s_1 \\ m_2: f_1, \dots, f_n & s_2 \\ \vdots & \vdots \\ m_r: f_1, \dots, f_n & s_2 \end{bmatrix}$

Steps 8

Learn function $[f_1, \dots, f_n] \xrightarrow{\psi} \{s_1, \dots, s_z\}$

Match to Stereotype Algorithm

```
Input:Partner features $f_1^{-i}, \dots, f_n^{-i}$.
Output: Partner model $m^{-i}$.

1. convert $f_1^{-i}, \dots, f_n^{-i}$ to instance of classifier data
2. result ← $\psi(f_1^{-i}, \dots, f_n^{-i})$
3. $m^{-i}$← StereotypeList( result )
4. return $m^{-i}$
```

Use features as input to function $\psi(f_1, \dots, f_n) \rightarrow s_k$

Figure 2. Algorithms for creating stereotypes and for matching newly perceived individuals to existing stereotypes. The create stereotypes algorithm operates by clustering partner models and then constructing a classifier mapping a partner's perceptual features to a stereotype. The match-to-stereotype algorithm simply uses the classifier to match the partner's perceptual features to the closest stereotype.

The **create stereotypes algorithm** makes two important assumptions. First, it assumes the existence of a distance function capable of measuring the difference between two partner models. Equation (2) was presented as a method for measuring partner model distance. If, however, additional information (such as the partner's beliefs, motivations, goals, etc.) is added to the partner model, then a more elaborate distance function may be necessary. Second, the algorithm assumes that partner models can be merged to create new partner models. In order to merge a partner model one must merge the components of the partner model. For this work, that meant merging the action models and utility functions. Action models were merged by adding an individual action to the combined model only if the action was included in at least half of the data that composed the merged model. For example, if the merged model was created from ten individual partner models and an action existed in four of the models then it was not included in the merged model. If, however, the action existed in five of the models then it was included in the merged model. Similarly, merged utility values were derived from the average utility value of the utility functions being combined.

4.2. Match-To-Stereotypes

When perceiving a new interactive partner, the robot matches the new person's perceptual features to an existing stereotype. This process begins by converting the partner's features into an instance of data for the classifier and then using the classifier to select the correct model (Figure 2 bottom). Line 1 (Figure 2 bottom) uses the partner's features to create an instance for classification. The result is matched to a stereotype (lines 2 and 3).

Prototype creation presupposes that at least some of the features witnessed by the robot actually correlate to an existing category. If the robot is incapable of recognizing any features which correlate to a category then that category may not be recognized, even if it exists. We believe that this limitation is mitigated when the robot can recognize a large number of partner features. Some features may correlate to multiple categories and the presence of a specific pattern of features many indicate that the individual is a member of a category. For the experiments conducted as part of this research three type-specific features and multiple non-type-specific features were included. We included the multiple non-type-specific features because one of our goals was to create a system that is ignorant of the a priori importance of the features. In other words, the robot's developmental history, including the people that it interacts with, determines the relative importance of the features, not the developer.

It should be apparent that these algorithms are composed of several well-known methods from machine learning. The contribution of this work is not so much the specific algorithm but rather the application of these methods to the challenge of learning categorical models of people and the use of this categorical information to guide a robot's social behavior. The section that follows presents a method for testing these algorithms.

5. General Experimental Methodology

Social psychology experimentation regularly involves the use of a controlled environment in which an experimental subject interacts with a confederate of the experimenter (Mitchell & Jolley, 1992). For these types of experiments the environment or the confederate can serve as an independent variable that the researcher manipulates. The behavior of the experimental subject, the robot in this case, is then observed and recorded as the dependent variable. These types of empirical evaluations tend to be high on internal validity, meaning that their results are indicative of a causal relationship. Unfortunately, they also tend to be low on external validity, meaning that the results do not typically generalize well beyond the experimental conditions. Future work in more realistic environments will address this limitation.

For social robotics research, experiments that treat the robot as an experimental subject and the human that the robot interacts with as a confederate of the experimenter offer several advantages. It may allow researchers a controlled method for determining the causal impact of their algorithms on the robot's social behavior. The researcher may thus be able to first verify the internal validity of their techniques before engaging in externally valid experiments involving unstructured interaction.

Like social psychology experiments, the experimental data we collected was obtained from video recording of interactions with the robot. The video was used to record the robot's actions and responses. This video was augmented with data saved from the robot's memory relating to its perception of the partner features, objects available for selection, action selections and outcomes obtained.

5.1. Experimental Setup

A coordination game was used for both the simulation and robot experiments. A coordination game is a game-theoretic social situation in which both individuals receive maximal reward only if they select coordinating actions (Osborne & Rubinstein, 1994). Figure 1 depicts an example of an outcome matrix representing a coordination game. In this example, both individuals receive an outcome of 10 if they select action pairs (*select-goggle*, *select-axe*), (*select-badge*, *select-radio*), or (*select-pills*, *select-mask*) and 0 outcome if any other action pair is selected. Table 1 lists all tools available. A maximum outcome of 10 was obtained whenever the robot and the partner selected tools from the same group.

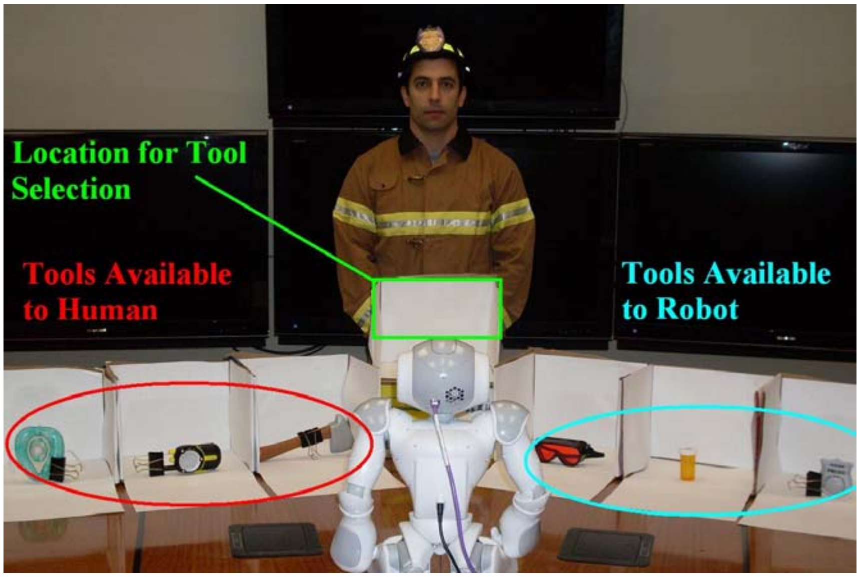


Figure 3 The experimental setup is depicted above. The objects in the blue circle are the tools available to the robot in this interaction. The objects in red circle are the tools that are available to the human in this interaction. The green square depicts the area where the human places tools that he or she has chosen. No tools have been chosen yet during the depicted interaction.

Table 1: Groupings of Tool Types

Type	**Tools**				
1	Extinguisher	Axe	Flashlight	Helmet	Goggles
2	Antiseptic	Mask	Neckbrace	Pills	Bandage
3	Binoculars	Radio	Handcuffs	Badge	Batton

At the beginning of the experiment, the robot searches for a face. Once it finds a face it collects the visual partner features from Table 2. Next, the robot queries the firefighter for the verbal partner features. Any undetermined features are recorded as such. The robot then searches to determine which tools are available to it and to the firefighter. It then uses a stereotyped partner model, if one exists, to predict which tool the person will select. It uses this information to select a tool for itself and verbally states its preference. The firefighter then indicates which tool he or she has chosen by placing the tool in the central box (Figure 3). Finally, after noting which tool the firefighter has selected, the robot waits for the person to either setup the tools for the next interaction or for a new individual to appear.

Notionally the robot is acting as a cooperative assistant to a human by selecting the best tool to assist the human partner. Figure 3 depicts the setup for the robot experiments. The robot selects from among the tools to its right (blue circle Figure 3). The human selects from the three tools to the robot's left (red circle Figure 3). The robot and human receive the maximal outcome if and only if they select a matching pair of tools. Table I lists all of the tools used in these experiments and the groupings of matching tools. In order to receive maximal outcome the robot needed to predict the tool that the person was going to select and to then select the tool that matched. The tools were randomly placed in different bins such that one tool of each type was located in the bins to the left and to the right. Overall, there were 8000 different possible arrangements of tools.

Robot experiments were conducted on an embodied, situated Nao robot by Aldebaran. The Nao robot is a Humanoid platform with 25 degrees of freedom, integrate speech synthesis and recognition capabilities, and two HD 1280x960 cameras.

5.2. Simulation Experiments

Simulation experiments were conducted in addition to robot experiments. These simulations focus on the quantitative results of the algorithms and processes under examination and did not attempt to simulate aspects of the robot, the human, or the environment. These numerical simulation experiments allowed us to test the proposed algorithms with thousands of simulated partners to evaluate the statistical significance of the results. For these types of experiments the human was simulated by providing the robot with a list of perceptual features from Table 2 representing a nominal person. This list of perceptual features was used in place of sensor derived perceptual features in the simulation experiments. The robot used this information in conjunction with the algorithms for creating and matching stereotypes (Figure 2) to obtain a stereotyped partner model of the person. This partner model was then used to predict the tool that would be selected by the simulated person. Using this prediction, the robot made its own tool selection. Finally, the simulated human selected their tool in accordance with the experimental condition and a numerical outcome value was awarded.

Table 2: Partner Features and Possible Values

Feature Name	Values	Verbal or Visual?	Robot Exp	Type Specific or Random	Error Rate in Robot Exp (%)
Badge Present	yes, no	Visual	Yes	Type specific	6.7
Uniform color	brown, green, blue	Visual	Yes	Type specific	0
Head Gear	yes, no	Visual	Yes	Random	6.7
Head Gear Color	black, green, blue	Visual	Yes	Type specific	6.7
Hair Color	black, blonde, red	Visual	Yes	Random	0
Beard Present	yes, no	Visual	Yes	Random	50
Facial Symmetry	highly, symmetric, asymmetric	NA	No	Random	NA
Facial Length	very wide, square, long, very long	NA	No	Random	NA
Skin Color	light, dark	Verbal	Yes	Random	17
Glasses Present	yes, no	Verbal	Yes	Random	3.3
Age	young, old, medium	Verbal	Yes	Random	33.3
Body Type	thin, heavy, medium	Verbal	Yes	Random	20
Height	tall, small, medium	Verbal	Yes	Random	10
Gender	male, female	Verbal	Yes	Random	0

5.3. Human Confederates

Laboratory experiments involving controlled human behavior are standard in many psychology experiments (Sears, Peplau, & Taylor, 1991). These experiments typically require that the experimenter’s confederate look and act in a specific manner. The robot’s interactive partner dressed and, within the limits of the experiment, acted like a firefighter, a police officer, or an EMT. In both the simulation experiments and the robot experiments, the robot was capable of perceiving the features and feature values listed in Table 2. In the simulation experiments the values for the features were given to the simulated robot. In the robot experiment, some of the values for the features were determined by having the robot ask the confederate questions such as “Are you male or female?” Others were captured visually. Table 2 lists those that were spoken and those that were visual. To generate the visual features for the robot experiment, the confederate dressed in Halloween costumes (Figure 3). Rather than seek the assistance of large numbers of confederates, we used false beards, wigs, and differences in attire to create the appearance, based on the visual limitations of the robot, of different individuals. Prior to experimentation, we tested and verified the recognition results on three different individuals. Only one person acted as a confederate for all of the robot experiment. The non-type-specific features for each individual, such as whether or not the person had a beard, were determined at random prior to experimentation. The person acted like a firefighter by selecting the tools denoted as type 1 in Table 1, an EMT by selecting tools of type 2, and a police officer by selecting tools of type 3. The next section uses these experimental techniques to examine the benefits of stereotype learning.

6. The Benefits of Stereotype Learning and Usage

As mentioned in section 1, humans use stereotypes to make social judgments in spite of their obvious and well documented shortcomings (Schneider, 2004). This section explores some of the potential reasons for developing robots that stereotype.

6.1. Bootstrapping the Process of Learning about a new Person

Perhaps the most obvious advantage to using stereotypes is that the stereotype can serve as an initial source of information when the robot meets a new person. This initial partner model is then used to make predictions about the partner's behavior, which in turn, impacts the robot's social behavior. We have shown (Wagner A. R., 2009b), as have others before us (Denzinger & Hamdan, 2006), that stereotypes serve this purpose.

For the sake of brevity, and because this result has been repeated by others, we present only the results from this experiment. The experimental setup essentially follows the description presented in section 5.1, only in simulation and with different features and high-level actions. See (Wagner A. R., 2009b) for a more complete description.

Figure 4 contrasts stereotype learning and use with relearning a new model for each different partner. The *x*-axis depicts the interaction number and partner number (P0-P19). The blue (light gray) line depicts a running average of the control (no-stereotype) condition. In this condition the accuracy of the robot's partner model is consistently poor when interacting with a new partner and results in the regular wave-like pattern. Because the robot does not learn across partners, it must rebuild its partner model with each new partner. Hence, with each newly encountered partner the robot's model is inaccurate until it gradually learns about the partner by interacting with them. In the experimental (stereotyping) condition (bold red/dark gray line), however, learning and using stereotypes eventually aids the robot's performance. Initially (P0-P6) the robot has no stereotype information. Hence its performance is equal to the no stereotype condition during P0, P1, P2, and P4. As models are added to the model space, stereotypes representing each of four different types (EMT, firefighter, citizen, and police officer) of partners are learned. The first several partners (specifically P0, P1, P2, P4, and P6) result in continued refinement of the robot's stereotype models as it constructs clusters that reflect the four different partner types. After the seventh partner the robot has interacted with enough different individuals to have stereotype models for each partner type. In this case, the stereotype model has 80 percent of the same values (actions and utilities) as the partner model. For the remaining partners (P8-P19) the stereotype models only need slight changes (missing action or inaccurate utility value) in order to reflect the partner's actual model. This fact is shown by the high level of accuracy depicted by the bold red (dark gray) line for the later partners in the early interactions. Using 80 percent accuracy as a threshold, the control condition requires an average of 10.2 interactions to reach this threshold. The using stereotypes experimental condition, on the other hand, required only 4.45 interactions on average. This result is significant $(p < 0.01)$.

As demonstrated by our work and the work of others (Denzinger & Hamdan, 2006; Burnett, Norman, & Sycara, 2010), stereotypes can be used to bootstrap the process of learning about a newly encountered person. The experiments that follow examine other ways that stereotype knowledge can be useful to a robot.

6.2. Using Stereotypes for Feature Selection

In addition to bootstrapping the process of learning about new individuals, stereotypes also act as a method for feature selection. Recall from Figure 2 that the stereotype creation algorithm results in a mapping ψ from the set of partner features to stereotype models. The domain of this mapping is subset of the entire set of recognizable features. In fact, several methods exist (decision trees, PCA, etc.) for calculating the relative importance of each feature towards determining the stereotype model. Features below some threshold of importance could thus be ignored allowing the robot to focus on only the most important features. We hypothesized that once a set of features had been selected, the robot could then use only these features to select a stereotype and make a decision, hence reducing perceptual processing. We also conjectured that this feature selection process could be used in situations where some features, perhaps because of occlusion, are not available to the robot.

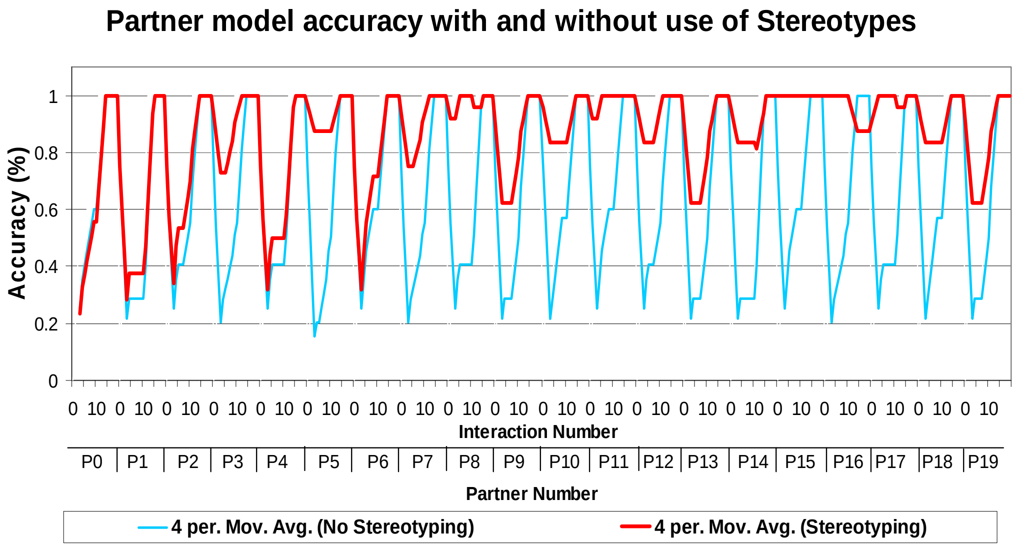


Figure 4. A result from an early stereotyping experiment is depicted above (Wagner A. R., 2009b). The blue (light gray) line is a moving average for the no stereotyping condition. The bold red (dark gray) line indicates is a moving average for the stereotyping condition. Stereotyping requires fewer interacts to obtain an accurate partner model once the stereotypes have been constructed (after P6). Prior to stereotype construction, however, both methods perform approximately the same. Note that the accuracy of the red (dark gray) line does not decrease as much as the blue line for partners P7-P19.

As the number of partners on which the stereotype is based increases, the robot learns which features distinguish the best among the categories. We hypothesized that by using this feature or features, the robot could select the appropriate stereotype and do a better job coordinating. To test this hypothesis we conducted a robot experiment using the general experimental setup described in section 5.1. As discussed above, the robot's task was to select the tool that coordinates with the human's type. To accomplish this task the robot used its model of the person to predict the person's tool preferences. The dependent variable was the percentage of correct coordinations. The number of partners on which the stereotype was based varied as the independent variable. We expected that stereotypes generated from a larger number of individuals would result in better predictions of the partner's tool preference.

Two different conditions were investigated as part of this experiment. In the best feature condition the robot uses the resulting stereotype to determine the best feature for distinguishing among the different stereotypes. Recall that a decision tree, ψ, was created mapping the partner's perceptual features to stereotypes. The top node represents the most important feature in this decision tree. In the random subset of features condition a randomly selected subset of perceptual features was presented to the robot. All other perceptual features related to the partner were unobserved. This condition was meant to simulate a situation in which environmental conditions limit the ability of the robot to perceive the person.

Prior to experimentation, the perceptual features for fifteen individuals were chosen by selecting random values for the random features (see Table 2) and type specific values for the type specific features. The perceptual features for five individuals of each type (firefighter, EMT, and police officer) were determined in this manner. The independent variable indicated how many randomly selected individuals of each type the robot interacted with prior to testing. During testing the robot was presented with an individual that had not been previously encountered. Twenty individuals were presented for each value of the independent variable and each condition. The robot interacted with each individual by playing 4 iterations of the coordination game. The choice of playing 4 iterations of the game was based on our results from section 6.1, the observation that the NAO software often failed after playing 80 games (4x20), and the experimenter's exhaustion. The following experimental procedure was followed:

1. The robot uses the algorithm presented in Figure 2 to create stereotypes for each category. The stereotype created was based on 1, 3, or 5 individuals per type.

2A. The robot uses its current stereotypes to determine which feature is most important. The robot then uses only methods related to this single perceptual feature to observe the new partner. Thus in this condition, the perception of the partner consists of a single feature/value pair.

2B. Robot is given information about 25% of the partner's features. The selection of which features the robot is given information about is random.

3. The robot uses the features to select a stereotype. The stereotype is used to predict the partner's tool preference and to select a tool.

4. The tools selected by both individuals are recorded.

The experiments show that, in the case of using the best features, as additional individuals are modeled and used to create the stereotype, performance increases from 45% to 100%, which is statistically significantly $(p < 0.01)$. After 3 individuals per type, use of the best feature consistently results in accurate prediction of the partner's tool selection. When a random subset of features is provided to the robot, however, performance initially increases by a significant amount $(p = 0.01)$ and then decreases. Overall, this condition depicts a slight upward trend, but, because of the large variance associated with the random selection of features, the results are not significant. Occasionally, this random selection of features produces no predictive features for the task. In this case, the robot's performance is no better than randomly choosing a tool, one out of three.

The results illustrate that stereotyping can be used as a method for feature selection. The stereotyping process generates an ordered list of features that the robot can use to guide its perceptual processing. When all features related to a new individual are available to the robot, restricting perceptual processing to those that provide the most distinction among the different types leads to the selection of the best stereotype for predicting the partner's behavior. Yet, when only a subset of features is available, the presence or absence of type specific features determines performance. Still, even in this case, stereotypes can likely be of assistance by informing the robot that the most important features are not available and, as such, no stereotype should be used.

Intuitively, as the number of partners increases, the stereotype model tends to converge on the features that have the lowest error rate. In the case of selecting the best feature (uniform color) the error rate was zero. This is both a limitation of the experiment and a reflection of the salience of the feature. The robot could always accurately detect the color of the person's uniform because the color of a person's uniform is a large (in terms of pixels), easily located feature and the experiment was conducted in well lit room. Performance in the case of the random subset condition was impacted negatively by the feature error rates.

6.3. Using Observations of Actions to Infer the Partner's Features

The previous sections have examined the potential use of stereotypes for bootstrapping learning about a new person and for feature selection. In this section we investigate if a stereotype can be used in conjunction with observations of one's behavior to infer what the person looks like. We speculate that inferences such as these could be useful when searching for a person in conditions that prevent clear observations of the person's features, such as in dark environments.

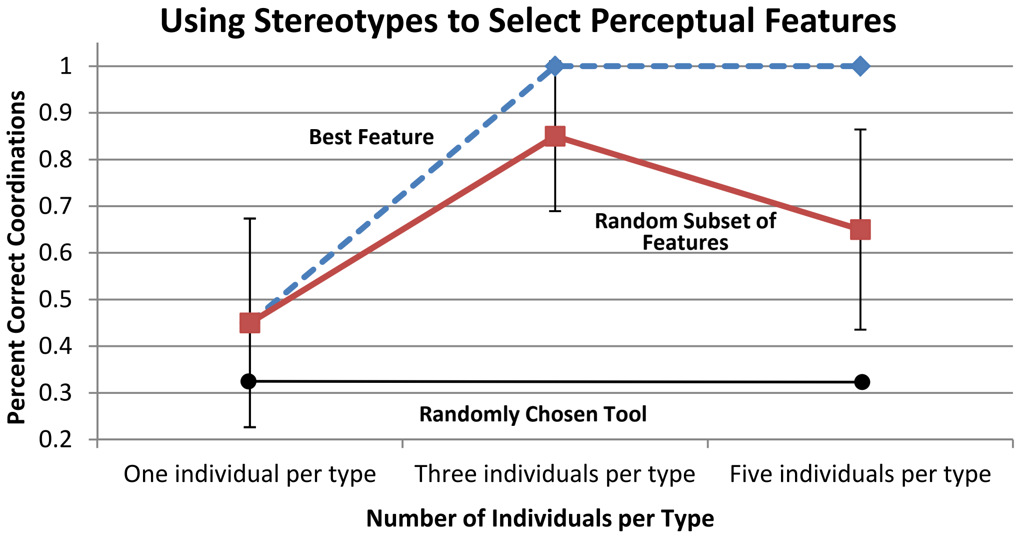


Figure 5. The chart above illustrates the results from an experiment investigating the use of stereotypes as a method for partner feature selection. The blue line (top) depicts a condition in which the stereotype is used to select the best feature for type determination. This condition results in increased performance as the stereotype is learned, ultimately reaching perfect coordination. The red (middle) line depicts a condition in which a random subset of features is selected. This condition does not result in statistically significant improvement, yet an upward trend is witnessed.

We hypothesized that the robot could use a stereotype model to infer a partner's perceptual features given only observations of the person's behavior. To test this hypothesis a robot experiment using the general experimental setup described in section 5.1 was again conducted. In this experiment, however, the robot observed the partner's tool choices rather than their perceptual features. These observations were then compared to the tool preferences for each of three developing stereotypes (firefighter, EMT, and police officer). The best match was used to predict the partner's perceptual features. The dependent variable in this case was the number of correctly predicted features. Here again the number of individual models used to create the stereotype was varied as the independent variable. This independent variable is meant to examine the impact of stereotype maturity. Two conditions were examined. In the first condition the percentage of correct features was determined by using the stereotype model to predict all of the unseen person's features. In the second condition the percentage of correct features is determined only from those features determined to have non-zero importance. Features with non-zero importance were those that had a node in the classifier tree of ψ. These two conditions are called the unweighted and weighted conditions respectively.

Prior to experimentation, the perceptual features for fifteen individuals were chosen by selecting random values for the random features (see Table 2) and type specific values for the type specific features. The perceptual features for five individuals of each type (firefighter, EMT, and police officer) were determined in this manner. The independent variable determined how many randomly selected individuals of each type the robot interacted with prior to testing. During testing the robot was presented with an individual that had not been previously encountered. Twenty individuals were presented for each value of the independent variable and each condition. The robot interacted with each individual by playing 4 iterations of the coordination game. The following experimental procedure was followed:

1. The robot uses the algorithm presented in Figure 2 to create stereotypes for each category. The stereotype created was based on 1, 3, or 5 individuals per type.
2. The robot observes the tool selection by new partner but does not observe their partner's features. The robot determines which stereotype's action model best matches the partner's action selection. The robot uses the selected stereotype to predict the partner's features.
3. The robot states the partner's features verbally and in a data file. A feature importance list is also recorded.
4. The predicted features are compared to the person's actual features.

5A. During analysis, the percentage of correctly predicted features is recorded for each independent measure.

5B. During analysis, the correctly predicted features are weighted by importance. The result is recorded for each independent measure.

The results indicate (Figure 6 blue middle line) that for the unweighted features the robot is able to use stereotypes in conjunction with the observed action to correctly predict between 45-49 percent of features. For comparison, a random selection of features would be expected to result in approximately 42% of correct features selected. The red (top) line depicts the results when the features are weighted by importance. The percentage of correct features selected in this condition ranges between 59-64 percent. The weighted feature condition significantly outperformed the unweighted condition when 3 or 5 individuals were used to create the stereotype. For both conditions, a lack of increased performance with respect to the independent variable indicates that the accuracy of the stereotype does not significantly impact feature prediction.

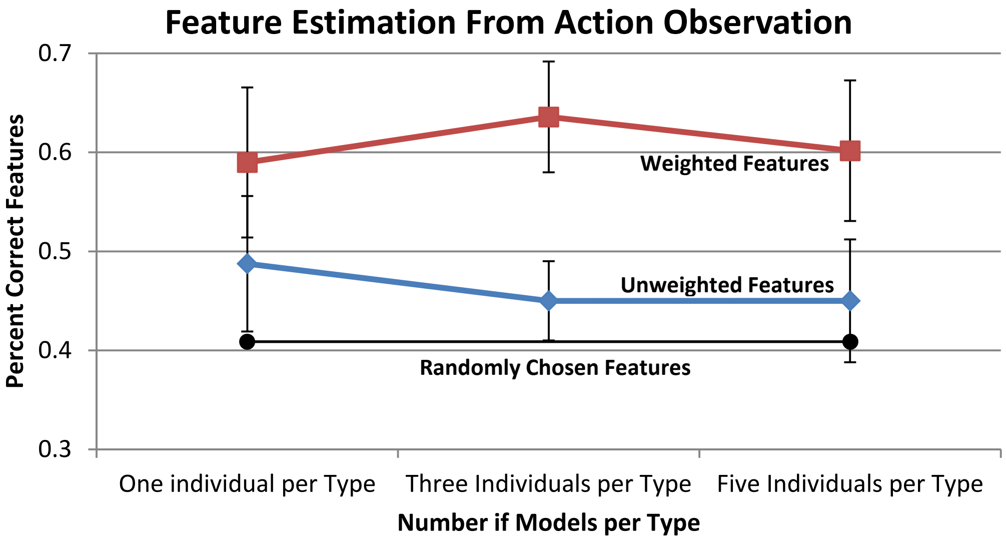


Figure 6. The chart above depicts the results from an experiment exploring the use of observations of actions and stereotype knowledge to infer a partner's features. The blue line (middle) depicts a condition in which the robot uses the stereotype to predict all of the partner's features. This condition does not result in statistically significant improvement. The red (top) line depicts a condition in which the predicted features are included only if the stereotype predicts that the feature has non-zero importance. This condition results in significantly increased performance over the unweighted condition, but an insignificant increase in performance with respect to the accuracy of the stereotype.

Overall, the results show that when the partner's features are weighted in terms of importance, stereotypes can be used to predict approximately two-thirds of these importance adjusted features. This is roughly 20% better than randomly selecting features. Several factors appear to have impacted performance. First, only three features are actually type specific (Table 2). Hence, in the unweighted case, the stereotype is being used to predict the values for features that are not related to the stereotype. This leads to higher variance in all conditions. Each feature also has a perceptual error rate associated with the robot's ability to perceive the feature (Table 2). This perceptual error rate reduces performance by impacting the accuracy of the stereotype model. As will be shown in the next section, error such as this degrades a stereotype model's ability to make predictions about an interactive partner. Finally, the classifier tree ψ learns that the type specific features are important after only interacting with one individual per type. Intuitively, additional individuals act as additional instances of data. Given sufficient individuals we expected that all of the extraneous features would have zero importance. This does appear to be the case but requires more than five instances of data to result in the expected effect. Our unfortunate decision to limit the independent variable to a maximum of 5 individuals per type was based on prior simulation experiments which did not include the same level of error. Moreover, because actions were performed by placing objects into bins, the error associated with action recognition was low (ranged from 5%-10%). This limits the generality of this experiment. We expect that higher action recognition error rates would result in moderate decrease in partner feature prediction rates.

6.4. Discussion

The experiments presented in this section illustrate several potential benefits for robots using stereotypes. The experiments support the contention that stereotypes can be used to 1) bootstrap the process of learning about a new person; 2) used as a method for perceptual feature selection; and 3) be used in conjunction with observations of a person's behavior to predict what they look like. Overall, these results indicate that, within the limitations of these experiments, stereotypes can serve as an important source of information for predicting, understanding, and responding to human interaction. In spite of the simplicity of the stereotype creation algorithm, the experiments have demonstrated on a real robotic system that the algorithm can be used to make inferences about a new person. Furthermore, these stereotypes were learned without prior knowledge by the robot as to the number of categories, any category characteristics, or of any specific people. Using only its model of itself and experiential information, the robot was able to learn the features and the behavior associated with three types of occupations. The use of stereotypes may also allow for a top-down method by which higher level deliberative processes can control and possibly enhance perception. Given hundreds or thousands of features it could be valuable for a robot to know which features are most important for a particular social judgment.

Nevertheless, some limitations are worth noting. The experiments were designed to be proof of concepts, and thus, no validation in terms of scalability or applicability in a more naturalistic setting is provided. Our goal in this article was only to show that this approach has merit. More realistic settings would likely impact the error rates associated with feature and action detection. The next section therefore explores how the use of stereotypes can act as a source of misinformation. For a variety of reasons, stereotypes can be misinformed representations on which the robot bases its predictions of the person's behavior. Understanding how and why these errors occur will be important if we are to employ these methods for in a more naturalistic setting.

7. The Challenges Associated with Stereotype Learning and Usage

The use of stereotypes presents several unique challenges to a social robot. Primary among these is the possibility that the stereotype itself could become a source of interactive misjudgments. In the sections below we examine the different ways that error can impact a robot's stereotype model.

7.1. Stereotyping Errors

Various types of errors can cause the stereotype model to contain incorrect content or to be incorrectly applied to a situation. As a result, the robot's model of its interactive partner will be incorrect which can, in turn, lead to improper social action selection. Consider, for example, a robot tasked with learning a stereotype of an EMT. In this case, the partner features might include perceptual information related to the person's uniform, appearance, height, weight, and age. The learned partner model would contain information related to the person's likely actions, such as performing CPR, bandaging victims, and stopping blood loss. A stereotyping error could cause a robot to not identify a person as an EMT, possibly leading the robot to provide incorrect information or selecting an action that impedes the EMT's work. It is thus valuable to understand how incorrectly learned and applied categorical models can impact the human-robot dyad. In the subsections below we examine the different types of error with the goal of understanding their potential impact on the robot's social behavior.

Perception is one source of error. The robot may misperceive the features that describe the partner's appearance or, alternatively, it may misperceive the information contained within the partner model, such as the action being performed. For example, if the robot incorrectly perceived a firefighter's brown uniform as blue, this would be a partner feature error. Partner feature errors could potentially cause the individual to be incorrectly categorized. On the other hand, if the robot incorrectly perceives the action of putting out a fire as the action of making an arrest, this is a partner action error. In this case, the robot's action model for that particular firefighter would indicate that the firefighter makes arrests. Partner action errors can cause the robot's stereotype to be inaccurate with respect to the true category. Both partner feature errors and partner action errors constitute perception errors.

Stereotyping error can also result from statistical anomalies. Ideally, the stereotype model is created from individuals that are representative of an underlying category. It is possible, however, that the individuals from which the stereotype is created are actually outliers with respect to the overall category. If this is the case, then the stereotype created will not be representative of the category. Consider, for example, a robot that creates a stereotype based on models it has learned from interactions with firefighters. Rather than put out fires and rescue victims, these particular firefighters knock on doors and ask for candy. Because the robot's stereotype has been created from outliers of the overall category, predictions based on the stereotype will be incorrect when the robot interacts with a non-outlier member of the category. Alternatively, assume that the stereotype is created from individuals that are representative of a category. Yet, when using the stereotype to predict the behavior of a newly encountered individual the robot then encounters a person that acts as an outlier with respect to the category. Because the person is an outlier with respect to the category, the stereotype model will not accurately reflect the new individual's behavior. For example, this error would occur if the robot were to create a stereotyped model based on actions and utilities of a typical firefighter and were to then encounter an individual that is dressed like a firefighter at a costume party.

Table 3: Types of Stereotyping Errors

Type	Name	Error Description	Exp.
1	Perception error	Misrecognition of partner features or of information in partner model	Yes
2	Un/ Representative stereotype	Individuals used to create stereotype are outliers or individual encountered is an outlier	Yes
3	Generalization error	Unfounded generalization	No

One final type of error that can occur is an error in generalization. Several authors note that prototypes, because they assume a simple, constrained representation of a category, are strongly impacted by the bias associated with generalization (Mitchell T. M., 1980; Briscoe & Feldman, 2011). Our algorithm generalizes by averaging outcome values and by adding actions to the stereotype model if the action occurs in more than half of the individual models that compose the stereotype. Thus, an action may be generalized to all of the members of a category even if the action is only valid for just over half of the individuals. Table 3 lists each type of potential stereotyping error and its characteristics.

Each of these different types of error can potentially result in incorrect action prediction. The extent to which each type of error poses a serious limitation for robotic stereotyping applications is not known. To better understand how these different types of errors impact action prediction and limits the usefulness of this approach we conducted an observational study which introduced different types of error into the stereotyping process and recorded the impact this error had on action prediction. The focus of this study was on perception errors and un/representative stereotype errors (types 1&2 from Table 3). We do not examine the impact of generalization errors in this paper. See Briscoe and Feldman for a full treatment of this type of error (Briscoe & Feldman, 2011).

Simulation and robot experiments were conducted to examine perception errors (type 1 from Table 3) and un/representative stereotype errors (type 2 from Table 3). Our experiments employed the tool selection game described in section 5.1. Confederates of the experimenter were simulated by providing the robot with a list of perceptual features from Table 2 representing a nominal person. The robot used this information in conjunction with the algorithms for creating and matching stereotypes (Figure 2) to obtain a stereotyped partner model of the person. This partner model was then used to predict the tool that would be selected by the simulated person. Based on this prediction the robot made its own tool selection. Finally, the human selected their tool in accordance with the experimental condition and a numerical outcome value was awarded.

For the simulated experiments the dependent variable was the percentage of correct coordinations performed by the simulated human and the robot. Correct coordinations represent a measure of task success. Four different conditions were examined. In the first, baseline condition, no error was introduced. In the second condition perception action error was added at a rate of 50%. Perception action error was introduced by giving the robot a 50% chance of incorrectly perceiving the human's action selection. In the third condition, representative stereotype error was added at a rate of 20%. Outliers, although perceptually similar to a firefighter, consistently selected tools not associated with firefighting. In the final condition, both types of errors were introduced at their respective rates. The rates for these different types of errors were selected based on preliminary experiments and were meant to reflect high error conditions. Thirty trials of the experiment were run in order to obtain statistical significance. A single trial consisted of 4 interactions in the game with 15 different individuals.

The results show (Figure 7) that the introduction of error in all three of the error conditions significantly $(p < 0.01)$ impacts the robot's performance when compared to the no error condition for all partners after the second partner. Yet for the outlier and action error conditions, performance significantly improves over the course of interacting with 15 partners ($p < 0.01$ for both). Most of this improvement occurs in the first two partners (+21% outlier, +19% action error). Further, both the outlier and action error condition significantly outperformed random tool selection. Finally, if we define reasonable performance as 75% percent correct coordinations, then results indicate that both the perception action error and outlier conditions obtained a reasonable level of performance over the course of the experiment.

To further explore these simulation results, we conducted a similar experiment on the NAO robot. In this experiment the robot interacted with 15 different notional firefighters in four different coordination games each. Here again the robot's ability to make correct coordinations was measured as a function of the interaction number. Because this experiment was conducted on a robot, the rate of partner feature errors and partner action errors could not be controlled. Over the course of the experiment 19.4% of the partner features were not recognized and another 8.5% were incorrectly recognized. The partner action error rate was 4.8%. Prior to the experiment outliers were selected at random with each partner having a 20% chance of being an outlier. Partners 6, 11, and 14 were selected as outliers. One condition contained outliers, the other did not.

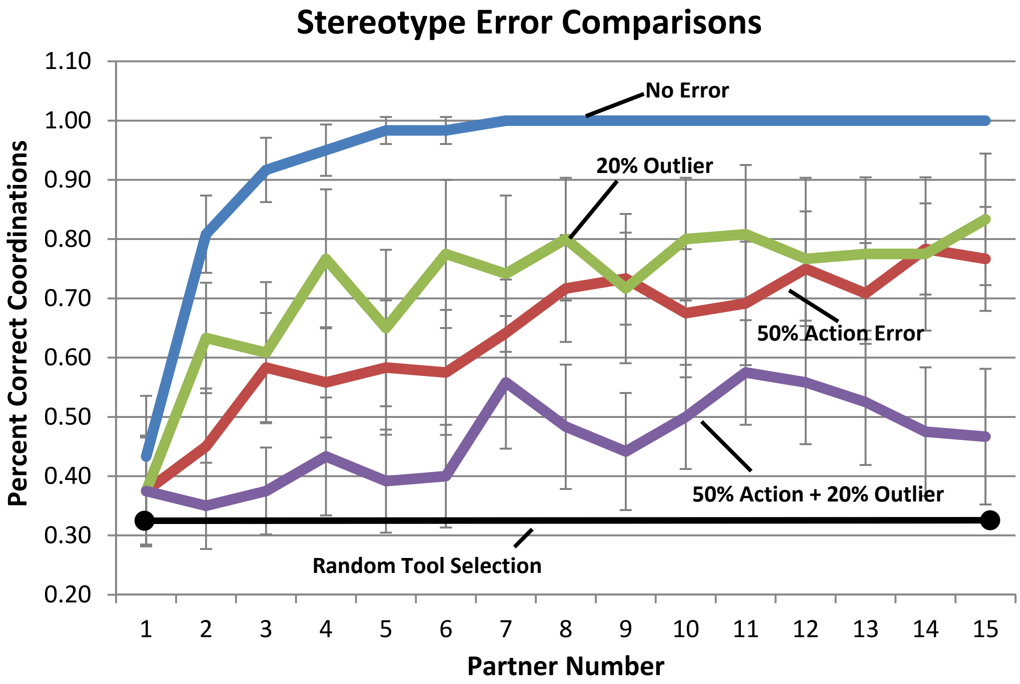


Figure 7. The chart above depicts the results from a simulation experiment examining different types of error. The blue line (top) depicts the no error condition, the green line (second) depicts the outlier condition, the red line (third) the perception action error condition, and the purple (bottom) line the action error and outlier error condition. Overall, outlier error and action error both result in a significant reduction of task performance when compared with the no error condition.

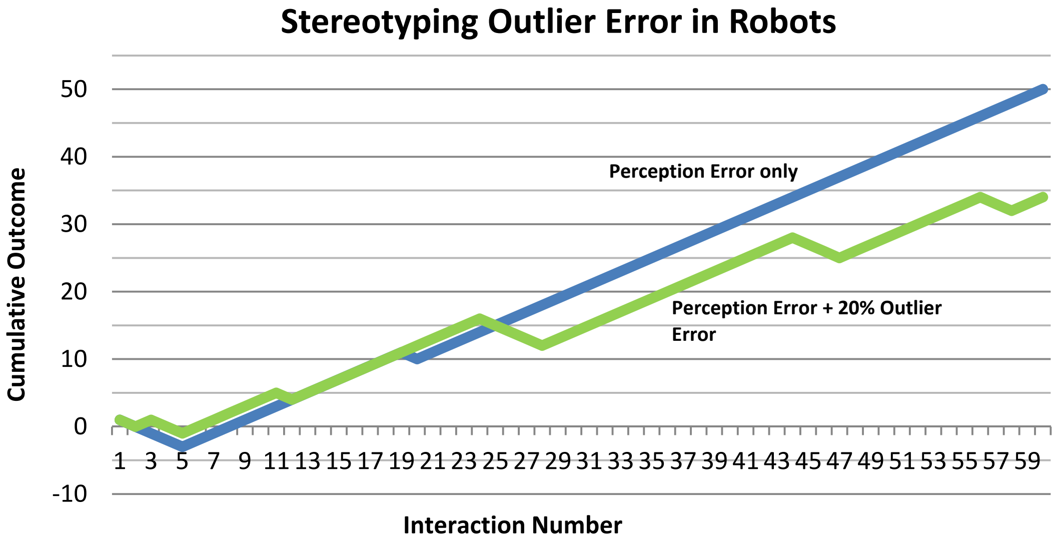


Figure 8. The chart above depicts the results from the robot experiment. The blue line (top) depicts the perception error only condition; the green line (bottom) depicts the outlier and perception error condition. Outliers impact performance locally only when the robot is interacting with the outlier.

Figure 8 depicts the results from the real robot experiment. The x-axis in Figure 8 indicates the interaction number. Hence every 4 interactions represents a new partner. There were 60 interactions during the entire experimental condition (15 individuals times 4 interactions per individual). The y-axis indicates the cumulative outcome received assuming +1 outcome for a correct coordination and -1 outcome for an incorrect coordination. For each condition, all 60 interactions were run in sequence. Hence, in both cases the robot iteratively learned and refined its stereotype model. Because only a single sequence was run per condition, determination of statistical significance was not possible.

As with the simulation experiment, the robot experiment indicates that the robot performs poorly (zero cumulative outcome during the first 6-9 interactions) in both conditions as it learns the stereotypes. Nevertheless, once the robot has created the stereotype, in the perception error only condition, it consistently picks the correct tool. In the combined perception error and outlier condition, the robot's performance matches the perception error only condition whenever it is not interacting with an outlier. The robot encounters its first outlier at interaction 24. In this case, for each of the 4 interactions, the robot incorrectly predicts the human's action selection, resulting in a cumulative outcome for this individual of -4. When the robot encounters the second outlier (at interaction number 44) it receives an outcome of -3. The final outlier (interaction 56) results in an overall outcome of -2.

Overall the experiments demonstrate the impact of two different potentially important types of error. Perception errors tend to cause interaction-specific but not individual-specific errors in coordination. This type of error could be characterized as a type of background noise. Outliers, on the other hand, cause individual-specific errors. A close look at the combined perception error and outlier error condition in Figure 8 reveals that the negative impact of an outlier decreases with each outlier encountered. As will be seen in the following experiments, this trend suggests that the timing of outlier error has an impact on performance.

7.2. The Timing of Errors

The previous experiment hinted that early outliers may have a longer impact on performance than later outliers. We can then ask if the process of learning a stereotype more sensitive to errors early in the category learning process, late in the category learning process, or does timing not matter at all? If the stereotype learning process is more sensitive to errors that occur early then applications that employ stereotyping should, to the extent possible, take care to observe examples that are representative of the category first.

Various social psychologists have demonstrated the importance of early interactions in the creation of stereotypes (Cordua, McGraw, & Drabman, 1979; Yamagishi, 2001). Influenced by work from social psychologists, we wondered what would happen if one of the initial models added to the robot's model space was an outlier. In other words, if one of the robot's earliest models does not reflect the true type, how does this impact performance with later individuals. We hypothesized that encountering an outlier early in the robot's social development would impact the robot's performance longer and to a greater extent than encountering an outlier later in development. Because these models act as the foundation for a new category, we believed that early errors would more strongly influence the development of the category. On the other hand, we further hypothesized that the timing of perceptual errors would not impact performance.

The dependent variable in this experiment was again the percentage of correct coordinations performed by the simulated human and the robot. The independent variable was again the partner number. Four different conditions were examined. In the perception action error conditions, error was added at a rate of 50 percent. Perception action error was introduced by giving the robot a 50 percent chance of incorrectly perceiving the human's action selection. In the outlier error conditions two partners were assigned the role of outlier. Again outliers, although perceptually similar to a firefighter, consistently selected tools not associated with firefighting. Early errors occurred while interacting with the 2nd and 3rd partners. Late errors occurred while interacting with the 12th and 13th partners.

Thirty trials of the experiment were run in order to obtain statistical significance. A single trial consisted of 15 interactions in the game with 15 different individuals. The create stereotypes algorithm from (Figure 2) was used to create new stereotypes after interacting with each partner.

The stereotypes that resulted were then used to predict the partner's action selection and which in turn influenced the action selection of the robot.

The results of this experiment are presented in Figure 9. When an early outlier is introduced (the red line) the robot's performance in the coordination task is impacted for the remainder of the experiment. Although the robot's performance gradually improves going from 28 percent to 64 percent over the course of partners 4 thru 15, in this condition the performance remains significantly ($p < 0.01$) below the performance obtained before the introduction of the error (98 percent). In contrast, when the robot encounters the same type of error later in the experiment (partners 12 and 13), its performance rebounds after only two partners to 76 percent. Put another way, when the outlier error occurs late it only takes one partner for the performance to rebound from 14 to 76 percent. When the same error occurs early, after 11 additional partners the performance still has not fully recovered.

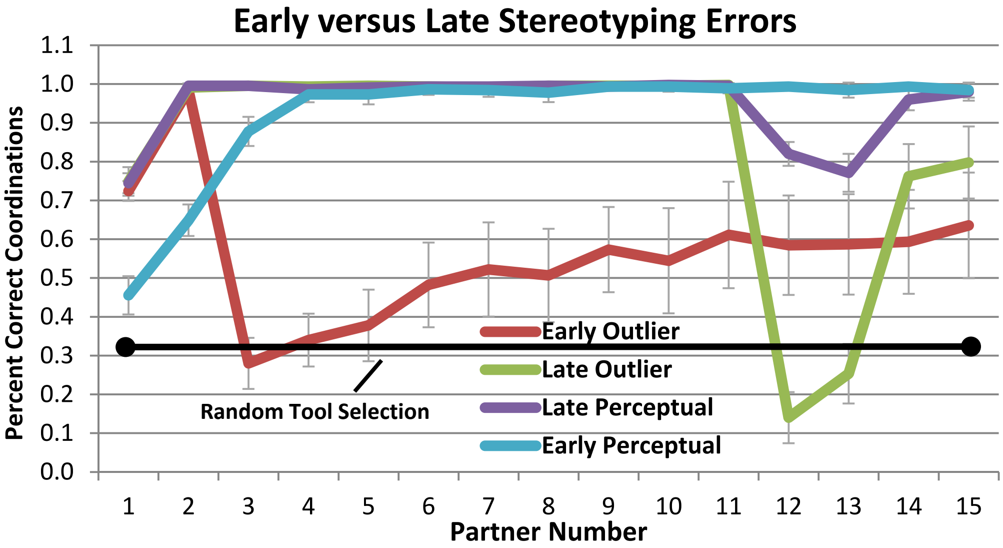


Figure 9. The chart above depicts the results from a simulation experiment examining the timing of different types of error. The blue line depicts a condition in which perception action errors occur early in the learning process. The purple line depicts the occurrence of late perception action errors. The green line illustrates late outlier errors. The red line illustrates early outlier errors. Overall, early outlier errors impact performance to a greater extent and longer duration then other types of error or when outliers occur later in the learning process.

The results for the perception action errors condition do not depict the same trend. Modeling error (purple and blue lines) only appears to impact performance while these errors are occurring. Moreover, the sum of the number of correct coordinations over all 15 partners is 207.13, 213.20, 125.20, and 186.70 for the early perception error, late perception error, early outlier error, and late outlier error respectively. Thus earlier outliers impact performance to a greater extent than any other type of error we examined.

Hence, we can conclude that the experiments support our hypothesis that earlier outliers impact a robot's performance longer and to a greater extent than late outliers. In addition, we have shown that outlier errors but not perception action errors, affect performance in this manner. These results are important for several reasons. First, they indicate not just that different types of errors impact social category learning in different ways, but also that the timing of these errors is critical. Hence the creation of a robot that learns and uses stereotypes would be aided by ensuring that the robot's first interactive partners are not outliers.

7.3. The Influence of Error on Close versus Transient Models

A person's model of another individual may vary with respect to the model's predictive ability (Funder & Colvin, 1988). In close relationships arising from long and varied interactive experiences, these models are rich and predictive over a large number of different circumstances. We term this type of model a close model. A close model is constructed by interacting with the person over a long period of time and over a larger variety of situations. One's model of a parent, child, or other close relative would be an example of a close model. Close models have significant predictive ability over a wide number of social situations. We therefore expect that the information from a close model could be used to accurately predict that person's responses and mental states in a greater number of situations. In fact, some evidence indicates that knowledge of a person is more important than perceptual indicators such as facial appearance (Todorov, Gobbini, Evans, & Haxby, 2007).

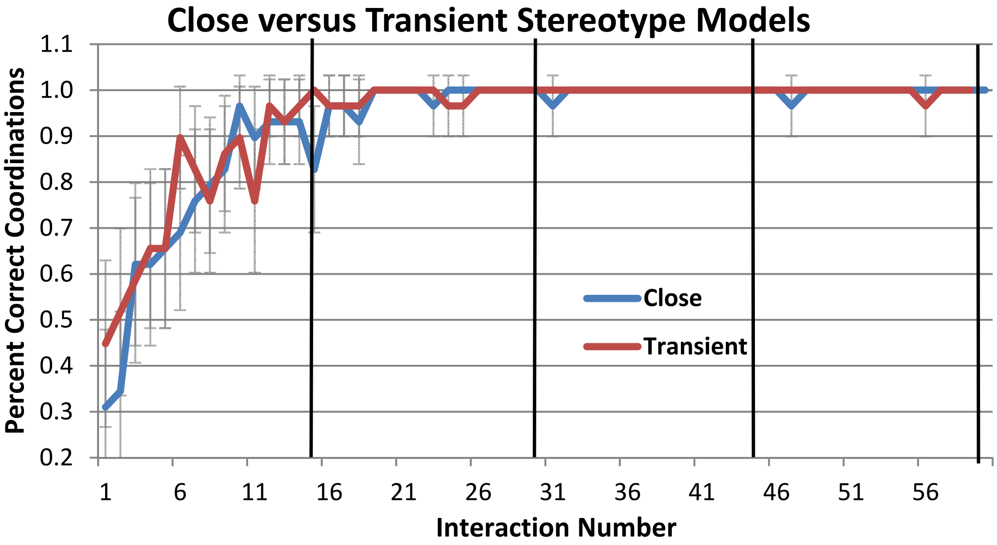


Figure 10. The graph above depicts the results from a simulation experiment comparing the impact of close versus transient models on coordination performance. In the close condition (blue) the robot interacted with 4 individuals for 15 interactions. In the transient condition (red) the robot interacted with 15 individuals for 4 interactions. The graph shows that close versus transient models do not impact performance. For this task, only the number of interactions influences performance.

In contrast to close models, a transient model is constructed over comparatively fewer interactions and contains less information. The transient relationships that commonly occur throughout the course of one's daily social behavior generate transient models. These types of models tend to be less predictive over a large number of situations and tend to contain less information. Transient models may make the bulk of a person's model space.

An understanding of the tradeoff between close and transient models could be important for a robot deciding whether or not additional interactions with a new individual is worth the time or energy investment. Some tasks, such as those involving trust, may demand a close model. Other tasks, such as those involving simple, transient interactions, may only require the information contained within a transient model.

We hypothesized that the creation of a close model would allow the robot to make more accurate predictions of its partner's behavior. Simulation experiments were conducted to test this hypothesis. These simulations used the approach described in section 5. The dependent variable was again the percentage of correct coordinations and the independent variable was the interaction number. In the close model condition the robot interacted with 4 partners for 15 interactions. In the transient model condition the robot interacted 15 partners for 4 interactions. We hypothesized that the creation of a close model would have greater predictive power during interactions with later partners than the transient models.

The results are depicted in Figure 10. The dark vertical lines delineate the interactions with each of the four different individuals in the close model condition. Notice that the lines representing the two conditions are not significantly different for interactions with any of the four individuals, refuting our hypothesis. In an effort to better understand these results additional simulation experiments were conducted. We hypothesized that the lack of difference might be a reflection of the task complexity. We varied the number of tools available to the robot expecting that a larger number of tools would favor close models. Although the results indicated a trend towards better performance of close models when the task complexity was increased, the difference was not statistically significant.

In general, our results indicate that the number of interactions, not the type of model, has the strongest influence on task performance. As illustrated in Figure 10, task performance initially increases with the number of interactions in both conditions, eventually reaching near perfect performance. The lack of difference between close and transient models may be a reflection of the task used in the experiments. For the tool coordination task, each interaction provides independent information reflecting the person's type. Individualized responses were not possible. Hence, this type of task naturally minimizes the difference between close and transient models. Close models result from a unique pattern of interactions over many different interactions. Put another way, close models are individualistic and not stereotypical.

7.4. Discussion

The experiments presented in this section have examined some of the challenges associated the use of stereotypes. Our results show that although errors impact performance, even in the presence of 50% perception action error or 20% outlier error the robot can still use stereotypes to perform 75 percent of coordinations correctly. The experiments also show that perception action error impacts task performance in a manner that is similar to background noise. Outlier errors, on the other hand, impact task performance locally only when the robot is interacting with the person that is the outlier. Hence these two types of error have qualitatively different effects on performance.

The experiments also show that the timing of specific errors has an important impact on performance. The impact of early outliers reflects the scarcity of early data distinguishing the true type from the outlier. This lack of data results in a mapping ψ which has little early basis for selecting a firefighter stereotype with the correct tool preferences over one with incorrect tool preferences. As additional data in the form new partners is added to the robot's model space earlier outlier performance eventually becomes statistically indistinguishable from late outlier performance. Hence the creation of a robot that learns and uses stereotypes would be aided by ensuring that the robot's first interactive partners are not outliers. To take this point a bit further, because the data highlights the importance of a robot's first interactive partners, it could be argued that the work foreshadows the potential importance of a type of robot parentage. This parentage would operate by providing correct, stable initial models from which the robot could contrast the behavior of later individuals that it meets.

A final experiment examined whether it was better for a robot to have more interactions with fewer partners or fewer interactions with more partners. The results of this experiment were inconclusive. Additional experiments which attempted to increase the complexity of the task also failed to demonstrate a difference when using close or transient models. We speculate that the experimental task, tool selection, is too limited to discern the value of learning close models. We expect that experiments that explore a larger, richer, and a more diverse set of social situations would observe situations in which close models performed better at predicting the person's behavior than transient models and vice versa.

8. Conclusions

This article has presented an algorithm for stereotype creation and usage by a robot. The algorithm clusters models of individuals that the robot has interacted with producing a centroid representing a generalized model of the partner, a stereotype. Next a function mapping the partner's perceptual features to cluster centroid is generated. The stereotype is used when the robot encounters a new, unknown person. The novel person's perceptual features serve as input to the function resulting in a centroid representing a stereotype, if one exists. A novel paradigm in which the robot acts as a test subject in social psychological style experiment was used to test this algorithm. The situation and behavior of the human was controlled by making the human a confederate of the experimenter. The robot's behavior was then observed as the dependent measure. Our experiments examined several benefits and challenges associated with stereotype learning and usage. The results indicate that the use of stereotypes allows the robot to bootstrap the process of modeling newly encountered individuals, that stereotypes can be used for feature selection, and that stereotypes can be used to infer a person's appearance after only observing the person's behavior. An exploration of the challenges of stereotyping revealed that perception action and outlier errors impact tool coordination performance in qualitatively different ways and that timing of outlier errors is an important consideration. Robots that are subjected to earlier outlier errors take longer to recover. A final experiment failed to show whether or not a close partner model confers an advantage with respect to predicting a partner's behavior.

The approach we present makes several assumptions. Foremost among these is the robot's ability to perceive. It is assumed that the robot can observe a number of features related to the person, the person's internal state or outcome, and recognize the action performed by the person. We have argued that many of these topics are actively being explored by other researchers. Still, the techniques presented here strongly argue for the further development of perceptual methods that will allow for the detection of high-level partner features such as hair style, clothing style, beardedness, etc. A recently funded project has allowed us to begin to develop some of these methods. It has also been assumed that, within the scope of the coordination task, the person being modeled is static. It is not clear if, and to what extent, the results apply if the person's behavior and internal state are rapidly changing.

The methods are not without limitations. Scalability with respect to the number of partner features, partner models, or underlying types has yet to be shown. Moreover, the methods do not currently included context information, which is an important factor when selecting predictive models of person. Techniques for managing situations in which one's initial stereotype is incorrect could be valuable for real world deployment. Such techniques have been explored in the past (Denzinger & Hamdan, 2006). It is worth reiterating that the primary contribution of this article is a preliminary method for learning and using stereotypes. Additional future work will be necessary in order to ensure that the use of these techniques in realistic environments is warranted and advisable.

8.1. Ethical Ramifications

A robot that stereotypes has important ethical ramifications. The methods presented here afford a means for social decision-making based on what a person looks like. In some cases this approach seems warranted. For instance, the detection of bloodstains as an indicator that a person needs to be rescued seems entirely appropriate. On the other hand, the use of race as an indicator of whom to rescue in a search and rescue situation is clearly not appropriate. But is it ethical for a robot to use perceptual cues related to age or gender to determine whom to rescue from a disaster? History presents numerous examples in which gender and age were the determining factors for deciding whom to rescue. Given that a social robot will learn its social rules from the people around it should it learn chivalry or egalitarianism? This article does not offer an answer to these questions, but notes that the creation of social robots will demand an answer.

Also of concern are the categories that a robot might learn, irrespective of their validity. In other words, should we as researchers allow robots to learn stereotypes even if the stereotype learned are justified? To this we respond by noting that the only feasible alternative to stereotyping that we can think of is to learn a new model for each individual the robot encounters. This approach might be appropriate for applications in which the robot only interacts with a limited number of people over its operational life. The goal of our research, however, is to develop robots that will interact with many people in a variety of different social situations. In this case, we see no alternative to the creation of categorical models of people.

A revealing anecdote occurred during one set of experiments: the robot's stereotype model indicated that a beard was the most important feature for determining if a person was a firefighter because, by chance, all of the firefighters the robot had interacted with had beards. These types of errors are common when humans stereotype (Schneider, 2004). In our experience, interactive diversity eradicates these types of unsupported associations.

8.2. Potential Applications and Future Work

This research represents an active and ongoing effort. Our work is directed at both the theoretical underpinnings of stereotype creation and usage and developing applications. This ongoing theoretical work explores alternative methods of creating stereotypes, including methods for visualizing unknown people. We are also in the process of developing a system which recognizes different types of rescuers in a search and rescue situations and provides type specific information to the individual. A system that recognizes different categories of disabilities or injuries in an assisted living setting and uses knowledge about a person's limitations to tailor the robot's behavior with respect to the person may also be useful. Further, categories related to a person's behavior, mannerisms, and/or context might also be possible.

As social robotics researchers come to recognize the importance of modeling a robot's interactive partner, methods that create categories of people will be necessary to allow the robot organize and bootstrap the process of learning about someone new. Categories related to disabilities could be of service in home healthcare and education settings, categories related to occupation and expertise could be used in search and rescue and military settings. Although the learning and use of stereotypes presents challenges and serious ethical considerations, it may also afford opportunities in terms of the development of applications devoted to aiding people in need.

Acknowledgements

We would like to thank Zsolt Kira and the reviewers for their helpful comments related to this article. Portions of this work were funded by contract N00164111-2008 from the Naval Surface Warfare Center.

References

Aggarwal, J. K., & Ryoo, M. S. (2011). Human activity analysis: A review. *ACM Computing Survey, 43*(3), 1-43. doi:10.1145/1922649.1922653

Anderson, S. M., Klatzky, R. L., & Murray, J. (1990). Traits and social stereotypes: Efficiency differences in social information processing. *Journal of Personality and Social Psychology, 59*(2), 192-201.

Ballim, A., & Wilks, Y. (1991). Beliefs, stereotypes, and dynamic agent modeling. *User Modeling and User-adapted Interaction, 1*(1), 33-65.

Bargh, J. A., Chen, M., & Burrows, L. (1996). Automaticity of social behavior: direct effects of trait construct and stereotype activation on action. *Journal of Personality and Social Psychology, 71*, 230-44.

Briggs, G., & Scheutz, M. (2011). Facilitating Mental Modeling in Collaborative Human-Robot Interaction through Adverbial Cues. *12th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, (pp. 239–247). Portland, Oregon. doi:http://www.aclweb.org/anthology/W11-2026

Briscoe, E., & Feldman, J. (2011). Conceptual complexity and the bias/variance tradeoff. *Cognition*, 2-16. doi:10.1016/j.cognition.2010.10.004

Burnett, C., Norman, T. J., & Sycara, K. (2010). Bootstrapping trust evaluations through stereotypes. *International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, (pp. 241-248). Toronto, CA. doi:10.1016/j.dss.2005.05.019

Cordua, G. D., McGraw, K. O., & Drabman, R. S. (1979). Doctor or Nurse: Children's Perception of Sex Typed Occupations. *Child Development*, 590-593.

Denzinger, J., & Hamdan, J. (2006). Improving observation-based modeling of other agents using tentative stereotyping and compactification through kd- tree structuring. *Web Intelligence and Agent Systems: An International Journal, 4*(3), pp. 255-270. doi:10.1109/IAT.2004.1342931

Duffy, B. (2004). Robots Social Embodiment in Autonomous Mobile Robotics. *International Journal of Advanced Robotic Systems, 1*(3), 155-170.

Edwards, A. (1940). Studies of Stereotypes: I. The directionality and uniformity of responses to stereotypes. *Journal of Social Psychology, 12*, 357-366.

Fasel, B., & Luettin, J. (2003). Automatic facial expression analysis: a survey. *Pattern Recognition, 36*(1), 259–275.

Fong, T., Thorpe, C., & Baur, C. (2001). Collaboration, dialogue, and human–robot interaction. *Proceedings of the International Symposium on Robotics Research*, (pp. 255-266).

Funder, D., & Colvin, C. (1988). Friends and strangers: acquaintanceship, agreement, and the accuracy of personality judgment. *Journal of Personality and Social Psychology, 55*, 149-158.

Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin de la Société Vaudoise des Sciences Naturelles, 37*, 547–579.

Kass, R., & Finin, T. (1991). General User Modeling: A Facility to Support Intelligent Interaction. In J. Sullivan, & S. Tyler, *Intelligent User Interfaces* (pp. 111-128). ACM Press.

Kelley, H. H., & Thibaut, J. W. (1978). *Interpersonal Relations: A Theory of Interdependence.* New York, NY: John Wiley & Sons.

Macrae, C., & Bodenhausen, G. (2000). Social Cognition: Thinking Categorically about Others. *Annual Review of Psychology, 51*, 93-120.

McCauley, C., Stitt, C. L., & Segal, M. (1980). Stereotyping: From prejudice to prediction. *Psychological Bulletin, 87*(1), 195-208.

Mitchell, M., & Jolley, J. (1992). *Research Design Explained* (2nd Edition ed.). Orlando, FL: Harcourt Brace Jovanovich.

Mitchell, T. M. (1980). *The need for biases in learning generalizations.* CBM-TR 5-110, Rutgers University, New Brunswick, NJ.

Mower, E., Mataric, M. J., & Narayanan, S. (2011). A Framework for Automatic Human Emotion Classification Using Emotion Profiles. *IEEE Transactions on Audio, Speech, and Language Processing, 19*(5), 1057-1070. doi:10.1109/TASL.2010.2076804

Osborne, M. J., & Rubinstein, A. (1994). *A Course in Game Theory.* Cambridge, MA: MIT Press.

Rich, E. (1979). User Modeling via Stereotypes. *Cognitive Science, 3*(197), 329-354.

Rilling, J. K., Gutman, D. A., Zeh, T. R., Pagnoni, G., Berns, G. S., & Kilts, C. D. (2002, July). A Neural Basis for Social Cooperation. *Neuron, 35*, 395-405. doi:10.1016/S0896-6273(02)00755-9

Schneider, D. J. (2004). *The Psychology of Stereotyping.* New York, New York: The Guilford Press.

Sears, D. O., Peplau, L. A., & Taylor, S. E. (1991). *Social Psychology*. Englewood Cliffs, New Jersey: Prentice Hall.

Smith, E. R., & Zarate, M. A. (1992). Exemplar-Based Model of Social Judgement. *Psychological Review*, 3-21.

Terveen, L. (1994). An overview of human–computer collaboration. *Knowledge-Based Systems, 8*(2–3), 67–81.

Todorov, A., Gobbini, M., Evans, K., & Haxby, J. (2007). Spontaneous retrieval of affective person knowledge in face perception. *Neuropsychologia, 45,* 163-173. doi:10.1016/j.neuropsychologia.2006.04.018

Trafton, J., Cassimatis, N., Bugajska, M., Brock, D., Mintz, F., & Schultz, A. (2005). Enabling effective human–robot interaction using perspective-taking in robots. *IEEE Transactions Systems, Man, Cybernetics A, 35*(4), 460–470. doi:10.1109/TSMCA.2005.850592

Wagner, A. R. (2009a). Creating and Using Matrix Representations of Social Interaction. *Human-Robot Interaction (HRI)*, (pp. 125-132). San Diego, CA. doi:10.1145/1514095.1514119

Wagner, A. R. (2009b). *The Role of Trust and Relationships in Human-Robot Social Interaction.* Ph.D. diss., School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA.

Wagner, A. R. (2012a). Using Cluster-based Stereotyping to Foster Human-Robot Cooperation. *Proceedings of IEEE International Conference on Intelligent Robots and Systems (IROS 2012*, (pp. 1615-1622). Villamura, Portugal. doi:10.1109/IROS.2012.6385704

Wagner, A. R. (2012b). The Impact of Stereotyping Errors on a Robot's Social Development. *Proceedings of IEEE International Conference on Development and Learning (ICDL-EpiRob 2012),* (pp. 261-270). San Diego, CA. doi:10.1109/DevLrn.2012.6400834

Yamagishi, T. (2001). Trust as a Form of Social Intelligence. In K. S. Cook, *Trust in Society* (pp. 107-131). New York, NY: Russell Sage Foundation.

Alan R. Wagner, Georgia Tech Research Institute, Atlanta, GA, United States. Email: alan.wagner@gtri,gatech.edu