People-Aware Navigation for Goal-Oriented Behavior Involving a Human Partner

David Feil-Seifer
Interaction Laboratory
Viterbi School of Engineering
University of Southern California
University Park, Los Angeles, CA 90089-0781
email: dfseifer@usc.edu

Maja Matarić
Interaction Laboratory
Viterbi School of Engineering
University of Southern California
University Park, Los Angeles, CA 90089
email: mataric@usc.edu

***Abstract*—In order to facilitate effective autonomous behavior for human-robot interaction, the robot should be able to execute goal-oriented behavior while reacting to sensor feedback related to the people with whom it is interacting. Our prior work has demonstrated that autonomously sensed distance-based features can be used to detect user state. In this paper we demonstrate that such models can also be used as input for action selection. We consider the problem of a robot moving to a goal along with a partner, demonstrating that a learned model can be used to weight trajectories of a robot's navigation system for autonomous movement. The paper presents a realization of a person-aware navigation system that requires no *ad hoc* parameter tuning, and no input other than a small set of training examples. The system is validated using an in-lab demonstration of people-aware navigation.**

I. Introduction

A long-term goal of the USC Interaction Lab is to enable the development of autonomous socially assistive robotic (SAR) systems [6] for domains such as elder care, post-stroke rehabilitation, and therapeutic intervention for children with Autism Spectrum Disorders (ASD). Major challenges from a system development standpoint include sensing human behavior and executing appropriate actions in real time. As part of our work on developing autonomous robots for children with ASD, we have focused on spatial interactions, particularly those that are social in nature and can be observed and influenced using interpersonal distance [7, 8]. Our prior work has shown that such spatial information could be used by an automated system to determine if a child is interacting with the robot, and to allow the robot to navigate in socially appropriate ways.

Feasibility studies have provided some important lessons concerning how intended robot actions can have unintended effects on people interacting with the robot. For example, in one of our studies involving children with ASD, we intended for our robot to face the child in order to maintain social contact. We observed that, when children with ASD try to avoid the robot by moving away from it, they become irritated when the robot orients to try to continue to face them. The children, in turn, attempt to move away from the robot, resulting in the robot in effect chasing them [8]. We also observed that, depending on the context, a robot moving toward a child could be viewed either as threatening or inviting. Thus, what is intended as a friendly social action for a robot to take might result in unintended negative consequences.

In general, a system designer cannot assume that, when a robot initiates a social interaction with a person, the a person will necessarily participate in that interaction in the intended or anticipated manner. Pre-programming social behavior might rely on undue assumptions or on over-simplifications of the sensing/actuation problem in such a way that intended social behavior might be inappropriate or unintended emergent behavior might interfere with the intended social act. During feasibility studies, we also observed that, when the robot moved away from the child for the purpose of maintaining appropriate social distance, the child sometimes thought that the robot was no longer interested in interacting.

Our preliminary work on using socially assistive robots with children with ASD has produced various insights, including that when the child is socially interacting with the robot, if the robot moves away from the child and toward some other goal, the child will typically not follow. From an external observer's perspective, this appears as if the robot is ignoring the child or vice versa. There is clearly a need for a more obvious/explicit way for the robot to reflect its intention/invitation for the child to follow. Our prior work has demonstrated that spatial movement behavior can be described using a Gaussian Mixture Model (GMM) and that the model can then be used to detect the social behavior of a person interacting with the robot [8]. If such a model can be used to detect social behavior taking place between a person and a robot, then it could also be used for action selection.

The focus of this work is to modify a trajectory planner that can exhibit goal-oriented behavior to include people-related sensing as part of the action selection process. We have chosen to explore the task of moving to a goal while being accompanied by a partner. In that task, both the robot and social partner need to arrive at the goal; their mutual proximity is a consideration but the relationship between the distance to the goal and to the partner is not explicit. Given that, what is the appropriate social distance between the robot and the social partner (child) and how does that distance change with progress toward the goal? Given a set of example movements, where a social partner is following a robot toward a goal, we aim to model example following behavior and use the resulting data-driven model for on-line people-aware trajectory planning.

Fig. 1. A view from the overhead camera with a pose overlay of the robot and axes showing the detected position of the social partner.

In this paper, we present a learned model for moving to a destination while accompanied by a person in addition to a method of weighting potential trajectories using that model. For validation, the model is trained and the system validated with typically developing adults. Our goal is to next train and validate the model with children with ASD.

The rest of this paper is organized as follows. In the next section, background and related work are presented. They are followed by a description of the modeling framework and the system implementation. Finally, we present an in-lab validation of the approach on an implemented system.

II. Prior and Related Work

Robins et al. [16] and Kozima et al. [13] used table-top robots to interact socially with children with ASD. Instead of interacting through robot base movement, these robots interacted through drumming and dance, respectively. The systems were either reactive or interacted explicitly with turn-taking behavior. They did not demonstrate deliberative planning behavior coupled with feedback from the user, as we intent to do. Salter et al. [17] used proprioceptive sensing, such as accelerometer information and orientation information, to determine how a ball-shaped robot was being played with and to select actions. For example, if the robot was being pushed, the robot moved quickly, while if the robot was being spun in a circle, it could play a noise. Such information is useful for reactive social interaction, however it does not involve specific planning behavior integrating an activity model.

Several groups have used a function of tracked positions over time for activity modeling for action feedback [12, 15, 19, 20]. Most of these tracked positions assume that events are related to absolute locations in the environment and utilize some function of the image coordinates as features in the model [19, 20]. Such approaches apply to conditions involving a fixed camera observing a mostly static scene; however, when modeling interpersonal interaction, proxemic information (the interpersonal distance and/or orientation between two mobile agents) is necessary as a supplement or replacement for absolute features [12].

Another prominent use of activity modeling is for detection of irregular behavior [2]. Still another is classification of human gestures [4, 15]. Finally, some activity modeling is targeted at safety, such as a drowning detector [5]. In a departure from purely data-driven approaches, some approaches use semantic labels of actions described in advance [3, 14]. Such work is usually an explicit alternative to attempting to model infrequent events, or to assign a text description to modeled actions.

The navigation approach in our prior experiments involving robots and children with ASD used interpersonal distance and orientation to decide action selection [7]. For the most part, the robot oriented toward the participant during the experiment. If the participant moved too far from the robot, the robot moved closer; if the participant moved too close to the robot, the robot slowly backed up. The system was completely reactive, requiring no planning on the part of the robot, and relying only on immediate feedback. This paper describes a new version of the system, featuring people-aware trajectory planning.

A popular approach to the type of navigation task we have described is the vector field approach [1]. In such a method, the goal and child emit an attractive field, while any obstacles emit a repulsive field. For the robot's pose in the environment, the sum of the fields at the robot's pose produces its desired trajectory. The interaction of such fields produces emergent behavior exhibiting the desired movement. Significant parameter tuning is typically required in such approaches, and would be especially challenging given differences in proxemic behavior in general and in particular among children with ASD.

III. Method

We present a modification to a trajectory planner that accomplishes the goal of navigation with a partner. We use a model trained over data of spatial features recorded human-robot interactions to rate candidate trajectories. In this section, we show how a widely-used trajectory planner can utilize this model for action selection.

A. The System Implementation

We implemented our system using the *nav_core* stack of the Robot Operating System (ROS), which is designed to be general-purpose, to work with a broad range of sensor inputs, and with holonomic or differential drives, in order to plan and execute trajectories to a target, (x,y,θ). The stack has already been validated on a variety of existing robot systems, making it an ideal starting place for adding person-aware properties to the navigation system. When a goal is set, the *nav_core* stack plans a path and determines control trajectories for movement toward the goal while avoiding obstacles:

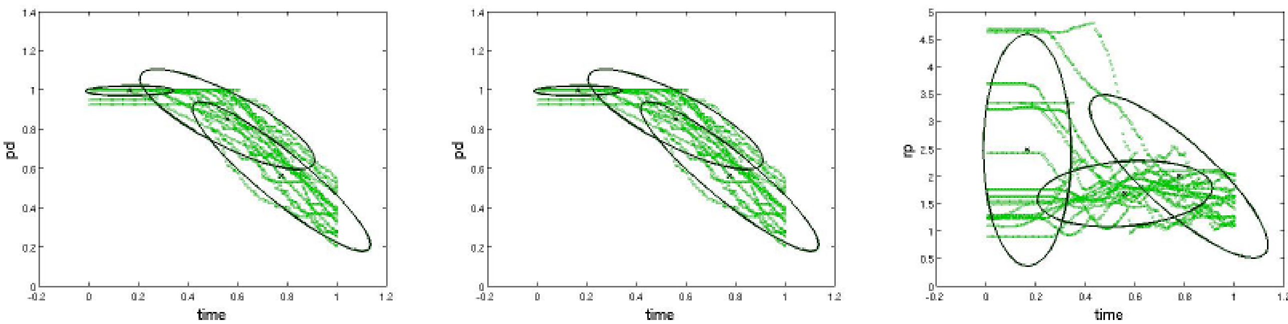

Fig. 2. The training data used for the model, and components of the trained GMM. Left: Δ_g^r by t. Center: Δ_g^p by t. Right: Δ_p^r by t.

1) Determine a global path to the goal. This path will avoid any known obstacles (either from a map or from sensor readings). This is implemented using the *navfn* package, applying Dijkstra's algorithm.
2) The local path planner will follow the global path, while avoiding any new obstacles encountered while traversing that path.
3) If the robot has diverged significantly from the global path, then the global path planner will re-plan.

For local planning the robot uses the base_local_planner package in nav_core. This system can employ either the Trajectory Rollout System [11] or the Dynamic Windowing Approach [10] to trajectory planning. Both systems create a series of candidate trajectories (v_x, v_y, v_θ), where v_x and v_y are translational velocities along the robot's x and y axes respectively (non-holonomic robots have a v_y of zero), and v_θ represents the rotational velocity. Trajectories are simulated for a finite period of time in the future (1.7 seconds in the current implementation), and are eliminated from consideration if their paths would result in contact with an obstacle. The remaining candidate trajectories are then scored, as follows:

$$cost(v_x, v_y, v_\theta) = \alpha(\Delta_{path}) + \beta(\Delta_{goal}) \tag{1}$$

where Δ_{path} is the distance of the simulated trajectory from the nearest point on the planned global path, and Δ_{goal} represents the distance from the goal, α and β are constants used to balance the drives for path fidelity and goal directedness, respectively. Thus, the robot can deviate from the path in order to avoid obstacles, but will also be weighted to move toward the goal. The trajectory with lowest cost is the control sent to the drive system of the robot. Such deliberative behavior applies to autonomous robot actions for undirected play scenarios typical of therapeutic intervention for children with ASD.

B. Detection of Robot/Person Position

We equipped the experiment space with an overhead camera to detect the positions of the robot, partner, and other obstacles in the room. Given *a priori* information about the layout of the room, this sensor alone was enough for the robot to maneuver autonomously in the experimental space and to use the detected positions over time to sense movement-based actions of the participants to trigger robot behavior.

We equipped the robot with two brightly and differently colored targets easily visible from the overhead camera, used to observe and track the position and orientation of the robot. We had the partner in the experiment wear a brightly-colored shirt so that his position, but not orientation, could be detected from the camera. Given that the camera was mounted to the ceiling and the experiment room is windowless, the background of the image (the floor) was mostly static; a small amount of image static occurred as a result of normal camera operations, but could easily be filtered out. Therefore, obstacles were detected as foreground blobs in the image.

Each video frame, collected at 15 frames per second, served as a single sample for detecting the locations of the robot and partner and deciding on an appropriate action. Each frame required less than 40% of the frame interval to process leaving additional time for behavior classification, supporting the real-time needs of the system. After each frame, this system was able to determine the position and orientation of the robot, the position of the partner, and any other obstacles that may be in the room. These positions made up the data used as features for the model used in this experiment.

C. Model Formulation

The formulation of the model requires selecting a candidate feature set. One approach to modeling mobility actions used the image coordinates of detected and tracked agents as part of a feature vector. However, such features assume that the relevant information can be expressed as a Cartesian position. That assumption holds for fixed-camera and fixed-goal activity-modeling systems [19]. However, for social interaction activity modeling, fixed positions are not nearly as useful as the distances between the social actors in the scene. Recent work has demonstrated the use of relative features for modeling human behavior [8, 12].

For this task, we use a 4-dimensional feature vector: $w = \langle t, \Delta_g^r, \Delta_g^p, \Delta_p^r \rangle$ where t is the normalized time spent on task ($t = 0$ is when the goal was set, $t = 1$ when the goal has been reached), Δ_g^r and Δ_g^p are the normalized distance between the robot or partner and goal (1 at the starting

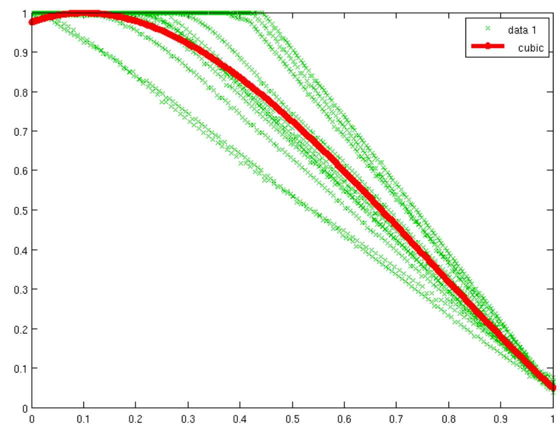

Fig. 3. Δ_g^r by t, with a cubic regression shown in red.

distance between robot or partner and goal, 0 at the goal itself). Δ_p^r is the distance between the robot and partner, in meters. Since the actions have a definitive start (when the robot sets a position goal) and end (when the robot reaches the goal), we can normalize these features by completion progress. These features can easily be extracted by the overhead system described in Section III-B.

We recorded complete examples of the robot moving toward a goal, with a person following. These examples varied in how far the robot was from the goal and how far the partner was from the robot or the goal. We trained the model using 14 example movements, for a total of 323 seconds comprising 3232 data points resulting in the model $\phi = \langle \mu, \Sigma \rangle$. The model was created by fitting a GMM using expectation-maximization (EM) to the training data. We used OpenCV for the GMM and EM implementations.

D. Trajectory Planning

As described in Section II, each candidate trajectory is evaluated for its cost to progress toward the goal. To accomplish the desired task of moving to the goal while being followed by a person, we evaluate trajectory fitness with the trained GMM as well. Each candidate trajectory results in an endpoint in Cartesian space. With this potential position for the robot along with the known positions of the partner and goal, the values of Δ_g^r, Δ_g^p, and Δ_p^r are known, only t is not known. Figure 3 shows that a cubic regression can accurately model the relationship between the robot's distance to the goal and the time spent moving to the goal. The candidate point w is the candidate feature vector given a trajectory, (v_x, v_y, v_θ).

In order to weight these trajectories using the GMM model, we modified the cost function as follows:

$$cost(v_x, v_y, v_\theta) = \Big(1 - p'(w|\phi)\Big)\Big(\alpha(\Delta_{path}) + \beta(\Delta goal)\Big) \tag{2}$$

Where $p'(w|\phi)$ is the probability that a candidate point w conforms to the given model ϕ. This preserves the existing method for planning a path to a goal, without requiring re-tuning of the bias parameters to accommodate the addition of the model. Using the Mahalnobis distance, the standardized distance of the feature vector w from a given component k of the model can be determined [9]:

$$\delta_M(w, k|\phi) = \frac{\sqrt{(w - \mu_{\phi(k)})^T \Sigma_{\phi(k)}^{-1} (w - \mu_{\phi(k)})}}{2} \tag{3}$$

This gives the distance from an individual component of the GMM, taking into account the variance of that component. This value can then be used to calculate the probability that w is part of this model [18]:

$$p(w, k|\phi) = \frac{1}{\sqrt{2\pi n}|\Sigma_{\phi(k)}|^{-1}} exp\Big(-\frac{\delta_M(w, k|\phi)}{2}\Big) \tag{4}$$

The probability that w conforms to a given model ϕ is the sum of the probabilities that it conforms to each of the k components of that model:

$$p'(w|\phi) = \Sigma_k p(w, k|\phi) \tag{5}$$

In summary, the modified cost function, Equation 2, should have a reduced cost for trajectories that conform to the model, and an increased cost for trajectories that do not conform to the model.

In the next section, we describe a validation of the described system.

IV. Validation

We evaluate the modified trajectory planner based on its effectiveness in reaching the goal while remaining close to the social partner, i.e., managing to bring the social partner along. In our training examples, the robot was about $1 - 1.5m$ away from the social partner when it reached the goal. Our validation compares the effectiveness of a robot using the standard trajectory planner provided with the ROS nav_stack to the robot using the modified navigation planner described above, applied to the described task performed with a human partner. To test this system, the partner was instructed to exhibit one of these three behaviors:

1) Follow as closely as possible, keeping up with the robot;
2) Wait a few seconds before following the robot;
3) Follow the robot, but at a very slow rate, slower than the robot's normal speed.

The above conditions are designed to demonstrate the effectiveness of the planner. Each participant was asked to perform each instruction 10 times, half of which were with the standard planner, half the modified planner. The conditions were presented in a randomized order. After the robot reaches the goal (defined as being within $0.3m$ radius of the goal), the distance of the partner with respect to the robot, and with respect to the goal, are recorded. The performance of

partner behavior	normal planner (m)	modified planner (m)
no waiting	1.41	1.25*
waiting	2.47	1.61*
slow	2.29	1.82*

TABLE I
AVERAGE DISTANCE BETWEEN PARTNER AND GOAL WHEN ROBOT ARRIVES AT GOAL, LOWER IS BETTER. ($*, p < 0.001$)

partner behavior	normal planner (m)	modified planner (m)
no waiting	1.16	0.99*
waiting	2.21	1.35*
slow	2.00	1.57*

TABLE II
AVERAGE DISTANCE BETWEEN PARTNER AND ROBOT WHEN ROBOT ARRIVES AT GOAL, LOWER IS BETTER. ($*, p < 0.001$)

the standard and modified planner are compared based on those values. Smaller partner-goal distance and partner-robot distances suggest better planner performance for the person-aware following task.

We hypothesized that, for partner behavior 1, there would be little or no difference between the traditional and modified planners, but that for behaviors 2 and 3, the modified planner would perform better than the standard planner, demonstrating that the robot's trajectory is chosen so that the partner can better follow the robot. Participants were recruited from the USC student community, ranging from those not at all experienced with robotics to those very experienced with robotics. In all, 8 participants were recruited (7 male / 1 female) with an average age of 20.8 years.

Table I shows how close the partner was to the goal when the robot reached its goal. Even for the first partner condition (no waiting, keeping up with the robot), the robot and partner reached the goal more closely together with the modified planner, with the partner just under $1.5m$ away. However, for the other two conditions, waiting and slow following, the partner was significantly further away from the goal when the robot arrived in the standard planner condition compared to the modified planner condition. This is a consequence of the robot slowing down and stopping when the partner was not moving toward the goal. The robot using the modified planner took longer to arrive at its goal, but the partner was much closer when it did.

Predictably, the results for robot-partner distance were consistent (see Table II), again as a consequence of the robot slowing down when the partner was not keeping up it. Overall, the validation shows that the robot, using the modified planner, is able to plan trajectories that allow a social partner to follow it, even with delays due to the partner waiting or moving very slowly.

V. DISCUSSION AND FUTURE WORK

The goal of this work was to execute goal-oriented behavior while incorporating people-aware sensing into the trajectory planning process. We developed a modification for the ROS trajectory planner to weight trajectories based on the fitness with a trained model of human following behavior. The modified planner was able to slow down and stop in order to wait for a follower to catch up. The validation showed that while both planners were able to guide the robot to the goal, the modified planner was able to arrive at the goal with the partner much closer to both the robot and goal.

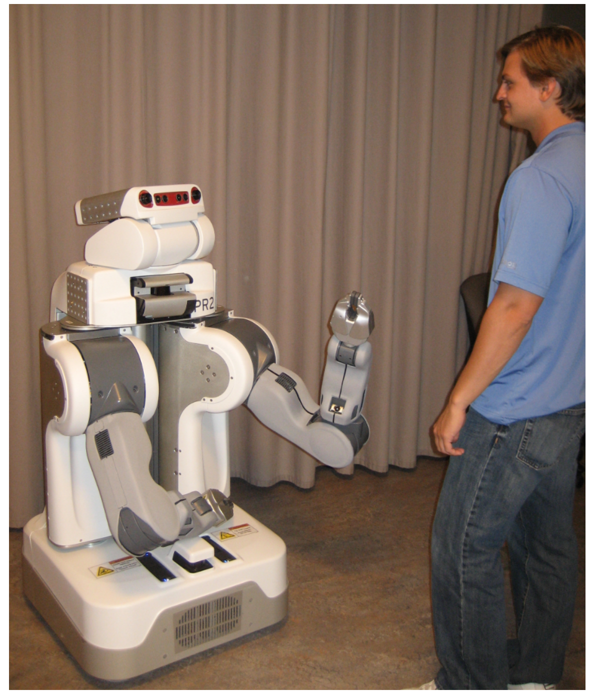

Fig. 4. The PR2 robot interacting with a social partner.

Our stated goals for real-time performance and minimal parameter tuning were also met by the described approach and system implementation. The system was able to recognize the features in real time and to evaluate the fitness of candidate trajectories with minimal modification of the existing trajectory planning systems. No tuning of the planner bias parameters was involved in order to make the modified planner function as intended, suggesting that this approach could be used for other data-driven models that could have impact on trajectory planning.

Social distance between agents can be determined not only from the distance to the goal but also from the free space available for navigation. The approach described in this paper is limited in that it only considers distance to the goal while ignoring the available free space. This model could be augmented to consider free-space features, such as free space in front of each social agent, distances to walls, and distances to other obstacles, in order to be more informed. However, since the motivating example for this work concerns open-space environments, the limited approach outlined in this paper is sufficient for the target social skill.

Continuing work on this project will test this trajectory planning approach with children with ASD. The goal is to have the person-aware navigation system enable free-form social interaction between a child and a socially assistive robot. In particular, we wish to use such people-aware navigation to encourage and train children with ASD to move close to other people, thereby facilitating social interaction. It would also be beneficial to use this model for detecting if a partner is not following the robot at all, in order to prompt some type of social intervention. The evaluation in this paper was only concerned with the distance between the robot and partner. However, when evaluating this approach in our target domain, we will explore the use of more subjective measures as well, such as how well/willingly/consistently the child follows the robot, to determine the effectiveness of this approach. We also wish to explore how such a modified trajectory planner can be used for other people-aware spatial tasks, such as walking with other people, game-playing scenarios, and turn-taking behaviors.

Acknowledgments

This work is supported by the National Science Foundation (CNS-0709296, IIS-0803565), the Dan Marino Foundation, and an ONR MURI N00014-09-1-1031. The PR2 robot was provided by Willow Garage.

References

[1] R. Arkin. *Behavior-Based Robotics*. MIT Press, 1998.

[2] O. Boiman and M. Irani. Detecting irregularities in images and in video. *International Journal of Computer Vision*, 74(1):17–31, 2007.

[3] M. Chan, A. Hoogs, J. Schmiederer, and M. Petersen. Detecting rare events in video using semantic primitives with HMM. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, volume 4, pages 150–154, Cambridge, England, UK, 2004.

[4] A. Efros, A. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *Proceedings of the International Conference on Computer Vision*, pages 726–733, Nice, France, 2003.

[5] H. Eng, K. Toh, A. Kam, J. Wang, and W. Yau. An automatic drowning detection surveillance system for challenging outdoor pool environments. In *Proceedings of the International Conference on Computer Vision*, pages 532–539, Nice, France, 2003.

[6] D. Feil-Seifer and M. Matarić. Defining socially assistive robotics. In *Proceedings of the International Conference on Rehabilitation Robotics*, pages 465–468, Chicago, Il, Jul 2005.

[7] D. J. Feil-Seifer and M. J. Matarić. Toward socially assistive robotics for augmenting interventions for children with autism spectrum disorders. In *11th International Symposium on Experimental Robotics 2008*, volume 54, pages 201–210, Athens, Greece, Jul 2008. Springer.

[8] D. J. Feil-Seifer and M. J. Matarić. Automated detection and classification of positive vs. negative robot interactions with children with autism using distance-based features. In *Proceedings of the International Conference on Human-Robot Interaction*, Lausanne, Switzerland, Mar 2011.

[9] D. Forsyth and J. Ponce. *Computer Vision, A Modern Approach*. Prentice Hall, 2003.

[10] D. Fox, W. Burgard, and S. Thrun. The dynamic window approach to collision avoidance. *Robotics & Automation Magazine, IEEE*, 4(1):23–33, 1997. ISSN 1070-9932.

[11] B. Gerkey and K. Konolige. Planning and control in unstructured terrain. In *Workshop on Path Planning on Costmaps, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Pasadena, California, May 2008.

[12] R. Kelley, A. Tavakkoli, C. King, M. Nicolescu, M. Nicolescu, and G. Bebis. Understanding human intentions via hidden markov models in autonomous mobile robots. In *Proceedings of the Conference on Human Robot Interaction*, pages 367–374, Amsterdam, Netherlands, March 2008.

[13] H. Kozima, C. Nakagawa, and Y. Yasuda. Interactive robots for communication-care: a case-study in autism therapy. In *IEEE International Workshop on Robot and Human Interactive Communication (ROMAN)*, pages 341–346, Nashville, TN, Aug 2005.

[14] S. Park and J. Aggarwal. Semantic-level understanding of human actions and interactions using event hierarchy. In *Conference on Computer Vision and Pattern Recognition*, pages 12–12, 2004.

[15] D. Ramanan and D. Forsyth. Automatic annotation of everyday movements. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, Cambridge, MA, 2004. MIT Press.

[16] B. Robins, K. Dautenhahn, and P. Dickerson. From isolation to communication: a case study evaluation of robot assisted play for children with autism with a minimally expressive humanoid robot. In *International Conference on Advances in Computer-Human Interaction*, pages 205–211, Cancun, Mexico, Feb 2009. IEEE.

[17] T. Salter, F. Michaud, D. Létourneau, D. Lee, and I. Werry. Using proprioceptive sensors for categorizing interactions. In *Proc. Human-Robot Interaction*, pages 105–112, Washington, D.C., March 2007.

[18] A. Watt and F. Policarpo. *The Computer Image*. ACM Press, 1998.

[19] T. Xiang and S. Gong. Beyond tracking: Modelling activity and understanding behaviour. *International Journal of Computer Vision*, 67(1):21–51, 2006.

[20] H. Zhong, J. Shi, and M. Visontai. Detecting Unusual Activity in Video. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, volume 2, pages 819–826, Washington, DC, USA, 2004. IEEE Computer Society.