Understanding Human Intentions via Hidden Markov Models in Autonomous Mobile Robots

Richard Kelley
Alireza Tavakkoli
Christopher King
Monica Nicolescu
Mircea Nicolescu
George Bebis

Department of Computer Science
University of Nevada, Reno
Reno, NV 89557
{rkelley, tavakkol, cjking,
monica, mircea,
bebis}@cse.unr.edu

ABSTRACT

Understanding intent is an important aspect of communication among people and is an essential component of the human cognitive system. This capability is particularly relevant for situations that involve collaboration among agents or detection of situations that can pose a threat. In this paper, we propose an approach that allows a robot to detect intentions of others based on experience acquired through its own sensory-motor capabilities, then using this experience while taking the perspective of the agent whose intent should be recognized. Our method uses a novel formulation of Hidden Markov Models designed to model a robot's experience and interaction with the world. The robot's capability to observe and analyze the current scene employs a novel vision-based technique for target detection and tracking, using a non-parametric recursive modeling approach. We validate this architecture with a physically embedded robot, detecting the intent of several people performing various activities.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics – *autonomous vehicles, operator interfaces, sensors.*

General Terms

Algorithms, Performance, Design, Experimentation, Human Factors, Theory.

Keywords

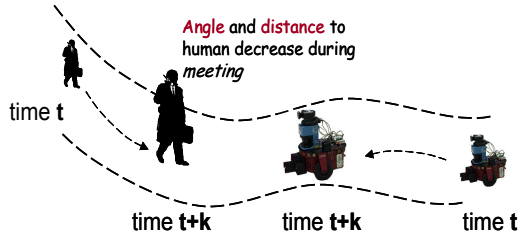
Human-robot interaction, intention modeling, Hidden Markov Models, Theory of Mind, vision-based methods.

1. INTRODUCTION

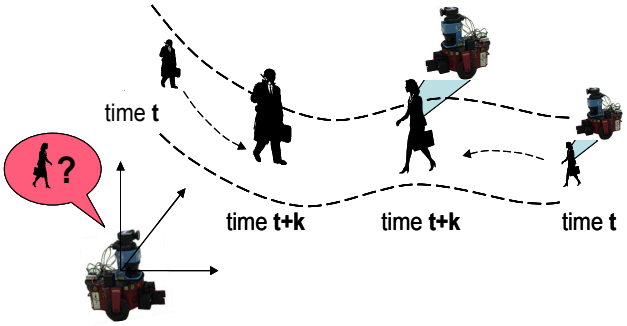
The ability to understand the intent of others is critical for the success of communication and collaboration between people. In our daily interactions we rely heavily on this skill, which allows us to "read" others' minds. Although this is very natural in humans, endowing a robot with similar skills has not been sufficiently addressed in the field. If robots are to become effective collaborators with humans, their cognitive skills must include mechanisms for inferring intent, so they can understand and communicate naturally with people. In this paper, we propose a method that targets the development of such capabilities.

The general principle of understanding intentions that we propose is inspired from psychological evidence of a Theory of Mind [1], which states that people have a mechanism for representing, predicting and interpreting each other's actions. This mechanism, based on taking the perspective of others [2], gives people the ability to infer the intentions and goals that underlie action [3]. We take an approach that uses the observer's own learned experience to detect the intentions of the agent or agents it observes.

Humans are continuously exposed to sensory information that reflects their actions and interactions with the world while performing certain activities. We propose to use this experience to infer the intent of others, by taking their perspective and observing their interactions with the world. When matched with our own past experiences, these sensory observations become



(a) **Activity modeling** from experience (as HMM)



(b) **Intent recognition** from own learned experience and perspective taking

Figure 1. The two stages of the architecture.

HRI'08, March 12–15, 2008, Amsterdam, Netherlands.

indicative of what our intentions would be in the same situation. We propose to model the interactions with the world using a novel formulation of Hidden Markov Models (HMMs), adapted to suit our needs. The distinguishing feature in our HMMs is that they model not only transitions between discrete states, but also the way in which parameters encoding the goals of an activity *change* during its performance. The goals are represented as abstracted environmental states, such as *distance-to-object* or *angle-to-goal*. This novel formulation of the HMM representation allows for recognition of the agents' intent well before the underlying actions are finalized. In our models, the *goals' changes* represent the *visible, observable states*, while the *hidden states* encode the *intentional goals* of the observable agents.

Our approach has two main stages: *activity modeling* and *intent recognition.* During the first stage the robot learns HMMs for each activity it should recognize, from its own experiences of performing these activities. For example (Figure 1(a)), the agent observes that during a *meeting* activity the *distance* and *angle* between its heading and the direction of a person decrease as the two agents are approaching.

During the intent recognition phase (Figure 1(b)), the robot, now an observer, is equipped with the trained HMMs and monitors other agent(s)' actions by evaluating the changes of the same goal parameters, from the perspective of the observed agents.

A significant advantage of our work is that unlike typical approaches to HMMs, which are restricted to be used in a single environment, our models are general and can be transferred to different domains. Even if trained in different environments, our HMMs encode features of the activities that are invariant between domains.

2. RELATED WORK

HMMs are a powerful tool for modeling sequential phenomena, and have been successfully used in applications involving speech and sound. Recently, HMMs have been used for activity understanding, showing a significant potential for their use in activity modeling and inferring intent. In particular, the HMM approach has been used mostly in manipulation tasks, which lend themselves naturally to segmentation into task stages, with clear discrete end-states (e.g., *object-on-table, object-in-hand*, etc.). Representative examples include learning to use a spatula and a pan [4], learning peg-in-the-hole assembly tasks [5], learning trajectory of a 7-DOF robotic arm [6], sequences of trajectories [7], and automated acquisition of behavior models [21]. In such training scenarios, the robot learns the transition probabilities between these states by observing the demonstration of the task performed by a human. The discrete states are linked to robot actions (e.g., *grasp, drop,* etc.), which combined with the learned HMM allow the robot to reproduce the demonstrated task. While some of the existing approaches allude to the potential of using HMMs to learn the user's intentions, these systems fall short of this goal: the approach allows detecting that some goal has been achieved only *after* observing its occurrence. However, for collaborative scenarios or detection of potentially threatening situations, it is of particular importance to detect the intentions *before* the goals of the actions have been achieved. In the context of using HMMs for activity recognition, several approaches have addressed the problem of gesture recognition [8], with the purpose of easily controlling the actions of a mobile robot, and robot behavior recognition [9], with application to the robot soccer domain. However, these systems require that an entire sequence of actions be completed before the activity can be recognized.

An application of HMMs that is closer to our work is that of detecting abnormal activity. The methods used to achieve this goal typically rely on detecting inconsistencies between the observed activity and a set of pre-existing activity models [10]. While this approach is useful in detecting deviations from expected activity patterns, it does not provide information regarding the intent of the observed actions.

Intent recognition has also been addressed from the perspective of intent inference and plan recognition for collaborative dialog [11], but these methods use explicit information such as natural language in order to infer intentional goals. Our robotic domain relies entirely on implicit cues that come from a robot's sensory capabilities, and thus requires different mechanisms for detecting intent.

In robotics, the only existing approach for intent recognition that we are aware of has been proposed by Gray *et. al* [12]. Their solution, which is also based on perspective taking, uses models of a robot's tasks to infer the goals and intentions of human users. The robot monitors the actions performed by the human from his/her perspective and matches them with high-level goals of its own tasks in order to infer what goals the human is trying to achieve. If the human encounters a problem, the robot is able to help the person finish the task. Thus, the method allows for detecting the intentional meanings of a human's high-level task goals (goal sequences or hierarchies). The difference in our work is that we aim at inferring intentions for lower granularity goals, such as the individual goals from [12], before the person finishes the actions meant to achieve them. Our models look at how an activity's goals are changing as the human executes it, rather than modeling a long task activity sequence.

3. GENERAL ARCHITECHTURE FOR INTENT UNDERSTANDING

3.1 Novel HMM Formulation

In our framework, an intention is represented as one of N discrete states $\{s_i\}$. At each time step the system can be in any of these states and can transition to another state with probability $P(s_j(t+1)|s_i(t)) = a_{ij}$. These components of the model describe how an agent's goal-directed mental states change as a scenario unfolds over time. One scenario is modeled using one HMM, and corresponds to one user-defined activity.

We assume that these mental states are not directly observable. Instead, a set of visible variables $\{v_i\}$, dependent upon the mental states, is available to the system. For each state s_j, we have a probability of observing a particular visible state v_k, given by $P(v_k(t)|s_j(t)) = b_{jk}$. In our approach, a model structure is given (i.e., number of hidden and visible states, topology of transitions between hidden states), along with a training set of sequences of the visible symbols. Our untrained models are initialized with an equal probability of observing each visible variable in each state. From these, the transition probabilities a_{ij} and the b_{jk} probabilities are computed. In the trained models, some visible variables will have zero probability of being observed when the system is in some states.

The main contribution of our approach consists in proposing models that focus on the dynamic properties of an agent's interaction with its environment. This new HMM formulation models an agent's interaction with the world while performing an activity, through the way in which parameters that encode the task's goals are changing (e.g., increase, decrease, stay constant, or unknown). This is in contrast with the traditional approaches that solely model static observable parameters, such as position at a single instant in time. With this representation, the *visible states* encode observed changes over time and the *hidden states* represent the underlying intentions that generate the observed behavior.

For the current work's models, we selected visible variables (change in position and angle) whose dynamic properties are easily observable *and* naturally correlate with particular mental states in the scenarios that we developed. For other scenarios, different visible variables may be more appropriate and should be chosen carefully by the modeler.

3.1.1 Activity Modeling

During this stage, the robot uses its experience of performing various activities to train corresponding HMMs, whose structure is currently designed by hand. The robot is equipped with a basis set of behaviors and controllers [21] that allow it to execute these tasks. We use a schema-based representation of behaviors, similar to that described in [13]. Activities that we used in this work include *Following*, *Meeting, Passing By, Picking Up an Object,* and *Dropping Off an Object*. While executing these activities, the robot monitors the changes in the corresponding behaviors' goals. For example, for a *meeting* activity (Figure 2), the *angle* and *distance* to the other person are parameters relevant to the goal, which could be {*angle* = 0 and *distance* = 1m} (i.e., "face the other person directly at 1m away). The robot's observable symbol alphabet models all possible combinations of changes that can occur: increasing (++), decreasing (--), constant (==), or unknown (∗). For example, a visible symbol could be v_k = {*distance:* --, *angle:* ++}. The underlying intent of the actions is encoded in the HMMs' hidden states.

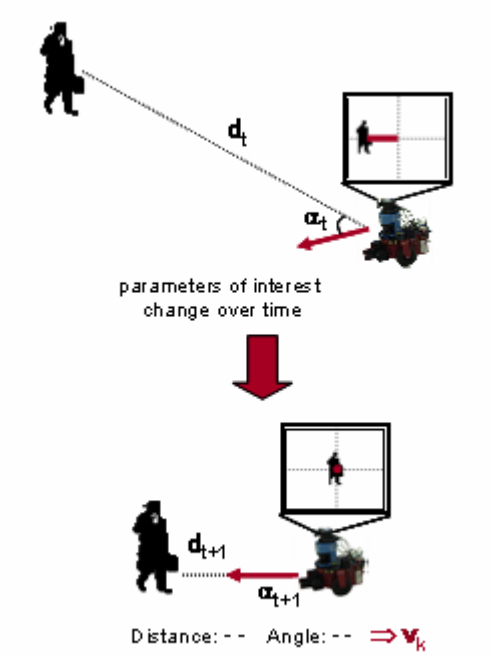


Figure 2. Activity modeling stage: observable symbols are changes in activity goals.

Repeated execution of a given activity provides the data used to estimate the model transition probabilities a_{ij} and b_{jk} using the Baum-Welch algorithm [14]. As a result of training, the robot has a set of HMMs, one for each activity. In our experience, adding an observation sequence to the training set had a large impact on the quality of the models when the training set was small (say, fewer than ten sequences). The impact of additional training sequences on the final models decreased substantially beyond about fifteen to twenty training sequences.

During the training stage, the observed, visible states are computed by the observer from its own perspective. The detection and tracking of relevant targets uses the robot's on-board sensing capabilities such as the camera and the laser rangefinder, as described in Section 4.

3.1.2 Intent Recognition

The recognition problem consists of inferring, for each observed agent, the intent of the actions they most likely perform, from the trained HMMs. Toward this end, the robot observer monitors the behavior of all the agents of interest with respect to other agents. The robot also evaluates the observable symbols for all applicable HMMs. During the recognition phase, the system computes these visible symbols in a different manner than during training. Since the observer is now external to the scene, the features need to be computed from the observed agents' perspective rather than from the observer's own point of view. These observations consist of monitoring the same goal parameters that have been used in training the HMM (e.g., change for distance to target, angle, etc.). For example, in Figure 3, in order to detect the intentions of the woman, the robot takes the following steps: (i) obtains agents' positions with respect to itself (values in black in Figure 3), (ii)

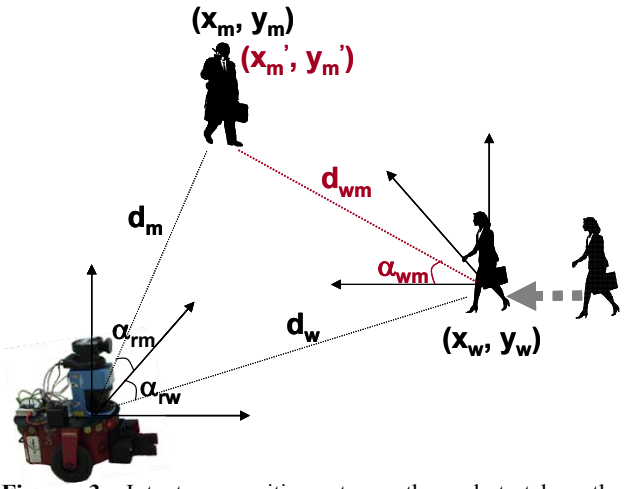
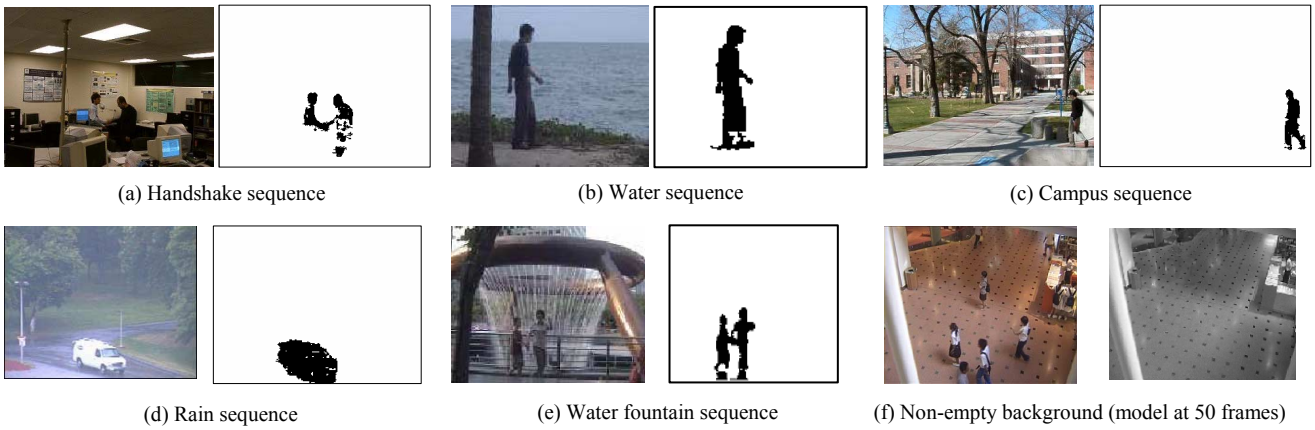


Figure 3. Intent recognition stage: the robot takes the perspective of the monitored agent. $d_{\{m,w,wm\}}$ represent distances, $x_{\{m,w,\ m'\}}$ and $y_{\{m,w,m'\}}$ represent 2D coordinates and $\alpha_{\{rm,rw,wm\}}$ represents the angle displacements w.r.t. the robot and woman.



(a) Handshake sequence (b) Water sequence (c) Campus sequence

(d) Rain sequence (e) Water fountain sequence (f) Non-empty background (model at 50 frames)

Figure 4. Background modeling and foreground detection in the presence of quasi-stationary backgrounds.

transfers the coordinate system to monitored agent (the woman), (iii) computes agents' positions from woman's point of view (values in red in Figure 3), and (iv) computes observable symbols in the woman's coordinate system. The woman's heading is computed by integrating her previous positions, which helps determine the orientation of the coordinate system in step (ii).

For each agent and for all HMMs, the robot computes the likelihood that the sequence of observations has been produced by each model, using the Forward Algorithm [14]. To recognize the intent of an agent we consider the intentional state emitted only by the model with highest probability. For that model, we then use the Viterbi Algorithm [14] to detect the most probable sequence of hidden states.

The standard approach to recognition using an HMM relies on a clear segmentation of the observed activities and on a precise synchronization between observed sequence and the recognizing process. Our system cannot assume that this segmentation is provided, as agents' underlying behaviors are not known, and can start or change at any time. A related challenge is that the observations come as a continuous stream of measurements, rather than as a fixed sequence. In this situation the probability of a particular model decreases to zero as the length of the sequence grows. To address this problem, we chunk the observation sequences to the most recent k observations, similar to [8]. In our work, $k = 30$ has been empirically determined to give good results.

4. VISION-BASED PERCEPTUAL CAPABILITIES

We provide a set of vision-based perceptual capabilities for our robotic system that facilitate the modeling and recognition of actions carried out by other agents. Specifically, we are interested in: *detection and tracking* of relevant entities, and *estimation of 3D positions* for the detected entities, with respect to the observer.

As the appearance of these agents is generally not known a priori, the only visual cue that can be used for attracting the robot's attention toward them is image motion. Although it is possible to perform segmentation from an image sequence that contains general motion (both the camera and the objects in the scene may be moving), such approaches are not very robust and quite time consuming. Therefore, our approach makes significant use of more efficient and reliable techniques traditionally used in real-time surveillance applications, based on background-foreground modeling and segmentation, structured as follows:

- During the *activity modeling stage*, the robot is moving while performing various activities. The appearance models of the other mobile agents, necessary for tracking, are built in a separate, prior process where the static robot observes each agent that will be used for action learning. During this process, the agents are detected through a foreground-background segmentation technique.
- During the *intent recognition stage*, we assume that the camera is static while the robot observes the actions carried out by the other agents. This allows the use of a foreground-background segmentation technique in order to build appearance models on-line, and to improve the speed and robustness of the tracker.

4.1 Detection and Tracking

For tracking we use a standard kernel-based approach [15]. The rest of this section describes our proposed method for background modeling and foreground segmentation.

The detection is achieved by building a representation of the scene background and comparing the new image frames with this representation. We focus on building a statistical representation of the scene background that supports reliable and real-time detection of foreground objects in the scene, while adapting automatically to each scene, and being robust to natural scene variations (quasi-stationary backgrounds).

The background model. In this work, we use a general *non-parametric modeling*, which estimates the density directly from the data, without any assumptions about the underlying distribution. This avoids having to choose a specific model (that may be incorrect or too restricting) and estimating its parameters. It also addresses the problem of background multi-modality, leading to significant robustness in the presence of quasi-stationary backgrounds. At the same time, it allows enough generality for handling a wide variety of scenarios without the need to manually fine-tune various parameters for each scene type, as all thresholds used in detection are estimated during model acquisition.

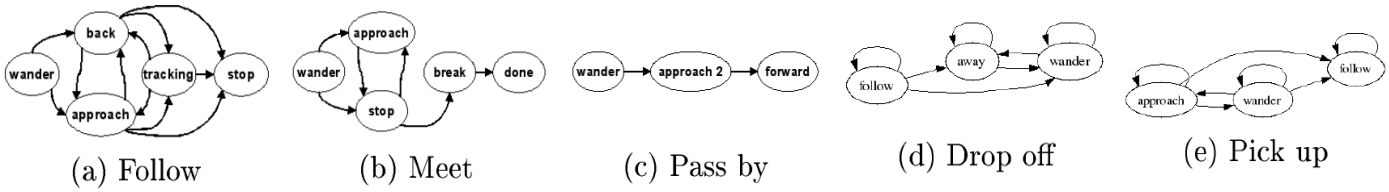


Figure 5. HMM structure for the five activities

However, a non-parametric approach such as [19] is still dependent on the number of image frames used as samples for estimating the background model. Choosing a small number of frames for the model increases speed, but results in a less accurate model. Increasing the number of frames improves the model accuracy but at the cost of higher memory requirements and slower convergence. In general, the non-parametric kernel density estimation tends to be memory and time consuming, as for each pixel in each frame the system has to compute the average of all kernels centered at each training sample.

In order to preserve the benefits of non-parametric modeling while addressing its limitations, we propose a *recursive modeling* scheme. Our approach employs a recursive formulation, where the background model $\theta_t(x)$ is continuously updated according to equation (1):

$$\tilde{\theta}_t(x) = (1-\beta_t)\cdot\theta_{t-1}(x) + \alpha_t \cdot H_\Delta(x - x_t)$$
$$\sum_x \theta_t(x) = 1$$

The model $\theta_t(x)$ corresponds to a probability density function (distinct for each pixel), defined over the range of possible intensity (or color) values x. After being updated, the model is normalized according to equation (2), so that the function takes values in [0,1], representing the probability for a value x at that pixel to be background. This recursive process takes into consideration the model at the previous image frame, and updates it by using a kernel function (e.g., a Gaussian) $H_\Delta(x)$ centered at the new pixel value x_t.

In order to allow for an effective adaptation to changes in the background, we use a *scheduled learning* approach by introducing the learning rate α_t and forgetting rate β_t as weights for the two components in equation (1). The learning and forgetting rates are adjusted online, depending on the variance observed in the past model values. This schedule makes the adaptive learning process converge faster, without compromising the stability and memory requirements of the system, while successfully handling both gradual and sudden changes in the background, independently at each pixel.

Results. Results on several challenging sequences are illustrated in Figure 4, showing that the proposed methodology is robust to noise, gradual illumination changes or natural scene variations, such as local fluctuating intensity values due to monitor flicker (a), waves (b), moving tree branches (c), rain (d) or water motion (e). The ability to correctly model the background even when there are moving objects in every frame is illustrated in Figure 4(f).

Quantitative estimation. The performance of our method is evaluated quantitatively on randomly selected samples from different video sequences, taken from [18]. The metric used is the *similarity measure* between two regions A and B, defined as $S = [A \cap B] / [A \cup B]$, where region A corresponds to the detected foreground, while region B corresponds to the true foreground. This measure is monotonically increasing with the similarity of the two foreground masks, with values between 0 and 1.

Table 1 shows the similarity measure for several video sequences where ground truth was available, as analyzed by our method, the mixture of Gaussians described in [17], and the statistical modeling proposed in [18]. It can be seen that the proposed approach clearly outperforms the others, while also producing more consistent results over a wide range of environments. We also emphasize that in the proposed method the thresholds are estimated automatically (and independently at each pixel), and there is no prior assumption needed on the background model.

Table 1. Quantitative evaluation and comparison to different methods. The video sequences are Meeting Room, Lobby, Campus, Side Walk. Water Surface and Fountain.

Video Sequence	MR	LB	CAM	SW	WS	FT	**Avg**
Proposed approach	0.92	0.87	0.75	0.72	0.89	0.87	**0.84**
Statistic modeling [18]	0.91	0.71	0.69	0.57	0.85	0.67	**0.74**
Mixture of Gaussians [17]	0.44	0.42	0.48	0.36	0.54	0.66	**0.49**

The proposed approach for background-foreground segmentation has the following benefits:

- The recursive formulation allows reliable convergence to the actual background model, without the need to specify a temporal sliding window, while being suitable for slow changes because of its low (and constant) memory and processing time requirements.
- The scheduled learning scheme achieves a high convergence speed, and a fast recovery from expired models, allowing for successful modeling even for non-empty backgrounds (when there are moving objects in every frame); its adaptive localized classification leads to automatic training for different scene types and for different locations within the same scene.

4.2 Estimation of 3D Positions

We employ the robot-mounted laser rangefinder for estimating the 3D positions of detected agents with respect to the observing robot. For each such agent, its position is obtained by examining the distance profile from the rangefinder in the direction where the foreground object has been detected by the camera.

For the intent recognition stage, once the 3D position of each agent is known with respect to the camera, a simple change of coordinates allows the observing robot to take the perspective of any participating agent, in order to map its current observations to those acquired during the action learning stage.

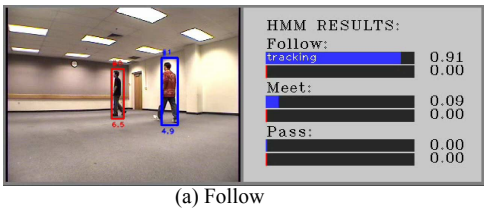
5. EXPERIMENTAL RESULTS

To validate our approach we performed experiments with a Pioneer 2DX mobile robot, with an onboard computer, a laser rangefinder and a PTZ Sony camera. The experiments consisted of two stages: the activity modeling phase and the intent recognition phase. Videos of all the experiments are available on-line at http://www.cse.unr.edu/~mircea/IntentRecognition/.

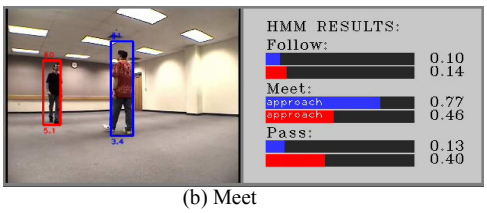
During activity modeling, the robot equipped with controllers for *following*, *meeting*, and *passing by, picking an object up from,* and *dropping an object off for* a person performed several runs of each of the three activities. In the *dropping off* scenario the agent comes with an object (e.g. suitcase, carry-on bags, etc.) drops the object off and leaves it unattended. In the *pick up* scenario, an agent comes to the scene which includes an unattended object, picks up the object, and leaves the scene. The observations gathered from these trials were used to train the HMMs represented in Figure 5, as explained in Section 3.1.1. The goal parameters monitored to compute the observable symbols are the distance and angle to the human, from the robot's perspective.

During intent recognition, the robot acted as an observer of activities performed by two people in several different scenarios, which included, *following, meeting, passing by*, *picking an object up, dropping an object off* and two additional scenarios in which the users switched repeatedly between these three activities. We performed each of the first three scenarios twice, to expose the robot to different viewpoints of the activities and thus to show the robustness of the intent recognition mechanism with varying environmental conditions. The goal of the two complex scenarios is to demonstrate the ability of the system to infer a change in intent as soon as the agents switch from one activity to another.

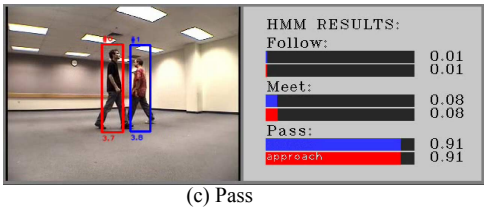
During each scenario, we recorded the probability that the models produced the observations, for each of the three HMMs. Figure 6 shows snapshots of the detection and intent recognition for one of the runs of the *following, meeting,* and *passing by* scenarios.



(a) Follow



(b) Meet



(c) Pass

Figure 6. Intent recognition for different activities.

Under each detection box we show the computed distance from the robot. The blue and red bars correspond to the blue and respectively red-tracked agent. The length of the red and blue bars represents the cumulative likelihood of the models up to that point in time, and the text inside the bars indicates the intentional hidden state of the highest likelihood model.

Figure 7 shows that the robot is able to infer the correct activity and intent for all the scenarios: the probability for the correct model rapidly exceeds the other models, which have very low likelihoods.

In the complex scenarios, the two subjects performed the following sequence of activities (agent 0 is tracked in red, agent 1 is tracked in blue):

- Scenario 1: pass by, meet, red follows blue, blue follows red
- Scenario 2: pass by, pass by, blue follows red, red follows blue

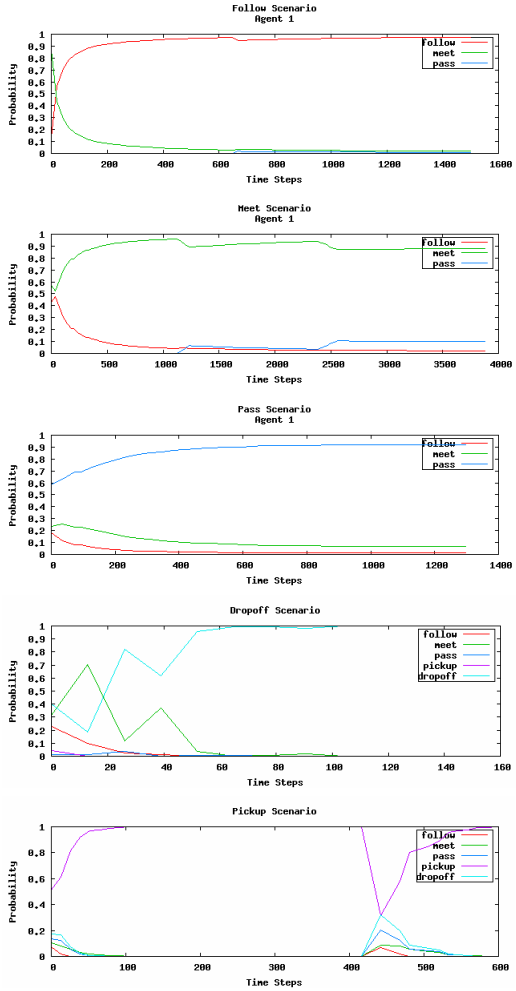
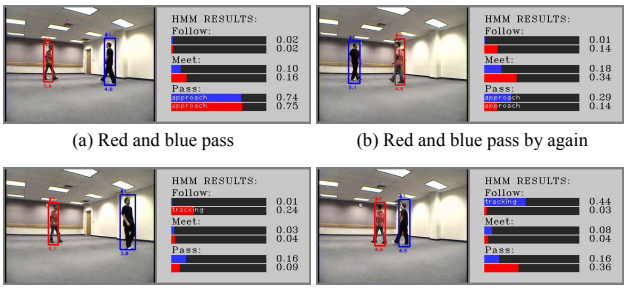


Figure 7. Probabilities for follow, meet, pass (all from agent 1's perspective), pickup and dropoff.



(a) Red and blue pass (b) Red and blue pass by again

(c) Red follows blue (d) Blue follows red

Figure 8. Results from complex

During these runs, the system was capable to quickly adapt to changes in people's activities and detect the correct intentional state of the agents, as shown in Figure 8. Although the activities follow each other continuously, the system does not require an explicit indication of when these start or end. The model with the highest current probability is that for which the graph bar has a label indicating the hidden state (such as tracking or approach).

To provide a quantitative evaluation of our method we employ three measures, typically used in evaluating HMMs [20]:

- *Accuracy rate* = the ratio of the number of observation sequences, of which the winning intentional state or activity matches the ground truth, to the total number of test sequences
- *Early detection* = t^*/T, where T is the length of the observation sequence and
  $t^* = min\{t \mid \Pr(winning\ intentional\ activity)$ is highest from time t to $T\}$
- *Correct duration* = C/T, where C is the total time during which the state with the highest probability matches the ground truth.

For a reliable recognition, the system should have high *accuracy rate*, small value for *early detection* and high *correct duration*. The accuracy rate of our system is 100%: all 12 intent recognition scenarios – 2 for following, 4 for meeting (for both agents), 4 for passing by (for both agents), 1 for drop off and 1 for pick up – have been correctly identified. Table 2 shows the values for early detection and correct duration for these experiments. For all except two cases, the robot inferred the correct intent of actions before less than 10% of the activity had been executed, and in five of the cases the correct intent was detected right from the start (*early detection* = 0). As expected, the *correct duration* for these cases had very high values, with the majority over 90%. The only two cases that produced worse results occurred when inferring the intent of agent 2, during the two meeting scenarios. In the first case, the robot had inferred the correct intent very early on, but had a brief moment when *pass by* seemed more likely at some point during the middle of the run (Figure 9). For most of the scenario, however, the robot correctly inferred that the agent's intent is for meeting (correct duration = 86.09%). In the second case, the robot had mistaken the meeting activity with a pass by, but only from the perspective of the second agent. Toward the end, however, the robot detects the correct intent as *meet* becomes the model with the highest likelihood. From our analysis of the data we observed that this result is due to small variations in computing the observable symbols from agent 2's perspective and due to the high similarity between meeting and passing by.

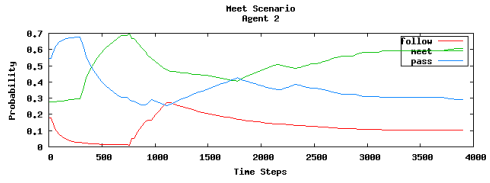


Figure 9. Probabilities for the *meet* scenario from agent two's perspective.

Table 2. Quantitative Evaluation

Scenario	Early Detection[%]	Correct duration[%]
Follow 1	1.23	98.77
Follow 2	3.70	96.30
Meet 1 - Agent 1	0	100.0
Meet 1 - Agent 2	47.25	86.09
Meet 2 - Agent 1	8.24	91.76
Meet 2 - Agent 2	52.45	47.55
Pass by 1 - Agent 1	0	100
Pass by 1 - Agent 2	0	100
Pass by 2 - Agent 1	0	100
Pass by 2 - Agent 2	0	100
Drop off	11.53	90.38
Pick up	0	100

6. CONCLUSION AND FUTURE WORK

In this paper, we proposed an approach for detecting intent with application to human-robot interaction. We use HMMs to encode a robot's interaction with others when performing various actions. These models are used through perspective taking to infer the intent of other agents and can perform this inference well before the agents' actions are finalized. This is in contrast with current activity recognition approaches, which only detect an activity after most of its stages are done. We developed a non-parametric vision-based technique to allow the robot to observe and analyze its environment. We validated this architecture with a physical robot, recognizing people's intent in several scenarios.

We are currently working on expanding the repertoire of activities for the robot to more complex navigation scenarios. We also plan to design collaborative scenarios that take advantage of the capabilities provided by our approach.

7. REFERENCES

[1] D. Premack, G. Woodruff, "Does the Chimpanzee have a Theory of Mind?", Behavioral and Brain Sciences, 1:4, pages 515-526, 1978.

[2] A. Gopnick, A. Moore, "Changing Your Views: How Understanding Visual Perception can Lead to a New Theory of Mind", in Children's Early Understanding of Mind, C. Lewis and P. Mitchell (eds.), Lawrence Erlbaum Press, pages 157-181, 1994.

[3] D. Baldwin, J. Baird, "Discerning Intentions in Dynamic Human Action", Trends in Cognitive Sciences, 5(4), pages 171-178, 2001.

[4] P. Pook, D. Ballard, "Recognizing Teleoperating Manipulations", International Conference on Robotics and Automation, pp. 578-585, 1993.

[5] G. Hovland, P. Sikka, B. McCarragher, “Skill Acquisition from Human Demonstration Using a Hidden Markov Model”, International Conference on Robotics and Automation, pp. 2706-2711, 1996.

[6] J. Yang, Y. Xu, C. Chen, “Hidden Markov Model Approach to Skill Learning and its Application in Telerobotics”, International Conference on Robotics and Automation, pp. 396-402, 1993.

[7] K. Ogawara, J. Takamatsu, H. Kimura, K. Ikeuchi, “Modeling Manipulation Interactions by Hidden Markov Models”, International Conference on Intelligent Robots and Systems, pp. 1096-1101, 2002.

[8] S. Iba, J. Weghe, C. Paredis, P. Khosla, “An Architecture for Gesture-Based Control of Mobile Robots”, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'99), Vol. 2, pp. 851 – 857, 1999.

[9] K. Han, M. Veloso, “Automated Robot Behavior Recognition”, IJCAI-99 Workshop on Team Behaviors and Plan Recognition, 1999.

[10] P. Thompson, “Weak Models for Insider Threat Detection”, Proc. of the Defense and Security Symposium, April, Orlando, Florida, 2004.

[11] B. J. Grosz, C. L. Sidner, “Plans for Discourse”, in Intentions in communication, P.R. Cohen, J. Morgan and M. E. Pollack, editors, Chapter 20, pages 417-444, 1990.

[12] J. Gray, C. Breazeal, M. Berlin, A. Brooks, J. Lieberman, “Action Parsing and Goal Inference using Self as Simulator,” in Proc., the 14th IEEE International Workshop on Robot and Human Interactive Communication (ROMAN), Nashville, Tennessee, 2005.

[13] R. C. Arkin, “Motor schema based navigation for a mobile robot: An approach to programming by behavior”, IEEE Conf. on Robotics and Automation, pages 264-271, 1987.

[14] L. R. Rabiner, “A Tutorial on Hidden-Markov Models and Selected Applications in Speech Recognition”, Proc. of the IEEE, Vol. 77, No. 2, Feb., 1989.

[15] D. Comaniciu, V. Ramesh, P. Meer, “Kernel-based object tracking”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, pp. 564–577, 2003.

[16] C. Wern, A. Azarbayejani, T. Darrel, A. Pentland, “Pfinder: real-time tracking of human body”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, pp. 780-785, 1997.

[17] C. Stauffer, W. Grimson, “Learning Patterns of Activity Using Real-Time Tracking”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 22, pp. 747-757, 2000.

[18] L. Li, W. Huang, I. Gu, Q. Tian, "Statistical Modeling of Complex Backgrounds for Foreground Object Detection", IEEE Transactions on Image Processing, 23, pp. 1459-1472, 2004.

[19] A. Elgammal, R. Duraiswami, D. Harwood, L. Davis, “Background and foreground modeling using nonparametric kernel density estimation for visual surveillance”, Proceedings of the IEEE, vol. 90, pp. 1151-1163, 2002.

[20] N. Nguyen, D. Phung, S. Venkatesh, H. Bui, “Learning and Detecting Activities from Movement Trajectories Using the Hierarchical Hidden Markov Model”, IEEE International Conference on Computer Vision and Pattern Recognition, pp. 955-960, 2005.

[21] M. N. Nicolescu, M. J. Matarić, “A Hierarchical Architecture for Behavior-Based Robots”, Intl. Joint Conf. on Autonomous Agents and Multi-Agent Systems, pp. 227–233, 2002.

[22] A. Olenderski, M. N. Nicolescu, “Robot Learning by Demonstration using Forward Models of Schema-Based Behaviors”, Second International Conference on Informatics in Control, Automation and Robotics, Barcelona, SPAIN, September, 2005.

[23] M. Fox, M. Ghallab, G. Infantes, D. Long, “Robot Introspection through Learned Hidden Markov Models,” Artificial Intelligence 170 (2006), 59-113.