Where is ...? Learning and Utilizing Motion Patterns of Persons with Mobile Robots

Grzegorz Cielniak[†‡] **Maren Bennewitz**[†] **Wolfram Burgard**[†]
[†]Department of Computer Science, University of Freiburg, 79110 Freiburg, Germany[*]
[‡]Department of Technology, Örebro University, 70182 Örebro, Sweden

Abstract

Whenever people move through their environments they do not move randomly. Instead, they usually follow specific trajectories or motion patterns corresponding to their intentions. Knowledge about such patterns may enable a mobile robot to robustly keep track of persons in its environment or to improve its obstacle avoidance behavior. This paper proposes a technique for learning collections of trajectories that characterize typical motion patterns of persons. Data recorded with laser-range finders is clustered using the expectation maximization algorithm. Based on the result of the clustering process we derive a Hidden Markov Model (HMM). This HMM is able to estimate the current and future positions of multiple persons given knowledge about their intentions. Experimental results obtained with a mobile robot using laser and vision data collected in a typical office building with several persons illustrate the reliability and robustness of the approach. We also demonstrate that our model provides better estimates than an HMM directly learned from the data.

1 Introduction

Whenever mobile robots are designed to operate in populated environments, they need to be able to perceive the people in their neighborhood and to adapt their behavior according to the activities of the people. Knowledge about typical motion behaviors of persons can be used in several ways to improve the behavior of a robot since it may provide better estimates about current positions of persons as well as allow better prediction of future locations.

In this paper we present an approach for learning probabilistic motion patterns of persons. We use the EM-algorithm [10] to simultaneously cluster trajectories belonging to the same motion behavior and to learn the characteristic motions of this behavior. We apply our technique to data recorded with laser-range finders. Furthermore, we demonstrate how the learned models can be used to predict positions of persons.

Recently, a variety of service robots has been developed that have been designed to operate in populated environments. These robots for example, have been deployed in hospitals [6], museums [4], office buildings [1], and department stores [5] where they perform various services e.g., deliver, educate, entertain [14] or assist people [13, 8, 12]. A variety of techniques has been developed that allows a robot to track people in its vicinity [15, 7]. Additionally, several authors have used models of people's motions to adapt the behavior of a mobile platform according to predicted movements [18, 17, 9]. These approaches, however, assume that a motion model is given. They provide no means to learn the parameters of the motion behavior of persons. Bui et al. proposed an Abstract Hidden Markov Model (AHMM) to predict people's motion [3]. They assume that the goals and subgoals a person might have (i.e. locations the person aims to approach) are given. Our approach, in contrast, is able to learn the intentions and to automatically derive the parameters of the underlying HMM. The technique described in this paper is an extension of the approach recently proposed by Bennewitz et al. [2]. We describe how to learn the intentions of persons and how to derive an HMM from the corresponding motion patterns. This Hidden Markov Model allows the robot to maintain a belief about the current location of a person.

The paper is organized as follows. The next section introduces our approach to learn motion patterns from observed trajectories and describes how we generate Hidden Markov Models to predict motions of persons. In Section 3 we present several experiments illustrating the robustness of our approach for estimating the positions of single and multiple persons using laser and vision data with a mobile robot. We also give results indicating that our models provide better estimates than Hidden Markov Models directly learned from the observations.

2 Learning Motion Patterns

When people perform their everyday activities in their environment they do not move permanently. They usually stop at several locations and stay there for a certain period of time, depending on what activity they are currently carrying out. Accordingly, we assume that the input to our algorithm is a

[*]This work has partly been supported by the German Science Foundation (DFG) under contract number SFB/TR8-03 and by the EC under contract numbers IST-2000-29456 and HPMT-CT-00251.

collection of trajectories $d = \{d_1, \ldots, d_N\}$ between resting places. The output is a number of different types of motion patterns $\theta = \{\theta_1, \ldots, \theta_M\}$ a person might exhibit in its natural environment. Each trajectory d_i consists of a sequence $d_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,T_i}\}$ of positions $x_{i,t}$. Accordingly, $x_{i,1}$ is the first position of the person when it starts leaving a resting area and x_{i,T_i} is the destination. The task of the algorithm described in this section is to cluster these trajectories into different motion behaviors and finally to derive an HMM from the resulting clusters.

Throughout this paper we assume that all trajectories have the same length T, where T is chosen as the maximum length of all trajectories. A trajectory d_i of length $T_i < T$ is extended by linear interpolation. The learning algorithm described below operates solely on this information and therefore does not take into account the velocities of the persons during the learning phase. In our experiments, we never found evidence that the walking speed of a person depends on its intention. However, one way to incorporate velocities is to introduce further dimensions to the state variables. The necessary changes to our clustering algorithm are then straightforward.

2.1 Motion Patterns

We begin with the description of our model of motion patterns, which is subsequently estimated from data using EM. Within this paper we assume that a person engages in M different types of motion patterns. A motion pattern denoted as θ_m where $1 \leq m \leq M$ is represented by K probability distributions $p(x \mid \theta_{m,k})$. For each $\theta_{m,k}$ the probability distribution $p(x \mid \theta_{m,k})$ is computed based on $\beta = \lceil T/K \rceil$ subsequent positions on the trajectories. Accordingly, $p(x \mid \theta_{m,k})$ specifies the probability that the person is at location x after $[(k-1)\cdot\beta + 1; k\cdot\beta]$ steps given that it is engaged in motion pattern m. Thus, we calculate the likelihood of a trajectory d_i under the m-th motion pattern θ_m as

$$p(d_i \mid \theta_m) \quad = \quad \prod_{t=1}^{T} p(x_{i,t} \mid \theta_{m,\lceil t/\beta \rceil}). \tag{1}$$

2.2 Expectation Maximization

In essence, our approach seeks to identify a model θ that maximizes the likelihood of the data. To define the likelihood of the data under the model θ, it will be useful to introduce a set of correspondence variables denoted as c_{im}. Here i is the index of the trajectory d_i, and m is the index of the motion pattern θ_m. Each correspondence c_{im} is a binary variable. It is 1 if and only if the i-th trajectory corresponds to the m-th motion pattern. If we think of a motion pattern as a specific motion activity a person might be engaged in, then c_{im} is 1 if person was engaged in motion activity m in trajectory i.

In the sequel, we will denote the set of all correspondence variables for the i-th data item by c_i, that is, $c_i = \{c_{i1}, \ldots, c_{iM}\}$. For any data item i the fact that exactly one of its correspondence variable is 1 leads to $\sum_{m=1}^{M} c_{im} = 1$.

Throughout this paper we assume that each motion pattern is represented by K Gaussian distributions with a fixed standard deviation σ. The goal is to find the set of motion patterns which has the highest data likelihood. EM is an algorithm that iteratively maximizes expected data likelihood by optimizing a sequence of lower bounds. In particular it generates a sequence of models denoted as $\theta^{[1]}, \theta^{[2]}, \ldots$ of increasing data likelihood. The standard method is to use a so-called Q-function which depends on two models, θ and θ'. In accordance with [2] this Q-function is factored as follows:

$$Q(\theta' \mid \theta) = \sum_{i=1}^{N} \Bigg(T \cdot M \cdot \ln \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \cdot \sum_{t=1}^{T} \sum_{m=1}^{M} E[c_{im} \mid \theta, d] \| x_{i,t} - \mu'_{m,\lceil t/\beta \rceil} \|^2 \Bigg). \tag{2}$$

The sequence of models is then given by calculating

$$\theta^{[j+1]} \quad = \quad \operatorname*{argmax}_{\theta'} Q(\theta' \mid \theta^{[j]}) \tag{3}$$

starting with some initial model $\theta^{[0]}$. Whenever the Q-function is continuous as in our case, the EM algorithm converges at least to a local maximum.

In particular, the optimization involves two steps: calculating the expectations $E[c_{im} \mid \theta^{[j]}, d]$ given the current model $\theta^{[j]}$, and finding the new model $\theta^{[j+1]}$ that has the maximum expected data log likelihood under these expectations. The first of these two steps is typically referred to as the E-step (short for: expectation step), and the latter as the M-step (short for: maximization step).

To calculate the expectations $E[c_{im} \mid \theta^{[j]}, d]$ we apply Bayes' rule, obeying independence assumptions between different data trajectories:

$$\begin{aligned} E[c_{im} \mid \theta^{[j]}, d] &= p(c_{im} \mid \theta^{[j]}, d) = p(c_{im} \mid \theta^{[j]}, d_i) \\ &= \eta p(d_i \mid c_{im}, \theta^{[j]}) p(c_{im} \mid \theta^{[j]}) \\ &= \eta' p(d_i \mid \theta_m^{[j]}), \end{aligned} \tag{4}$$

where the normalization constants η and η' ensure that the expectations sum up to 1 over all m. If we combine (1) and (4) utilizing the fact that the distributions are represented by Gaussians we obtain:

$$E[c_{im} \mid \theta^{[j]}, d_i] = \eta' \prod_{t=1}^{T} e^{-\frac{1}{2\sigma^2} \| x_{i,t} - \mu_{m,\lceil t/\beta \rceil}^{[j]} \|^2}. \tag{5}$$

Finally, the M-step calculates a new model $\theta^{[j+1]}$ by maximizing the expected likelihood. Technically, this is done by computing for every motion pattern m and for each probability distribution $p(x \mid \theta_{m,k})$ a new mean $\mu_{m,k}^{[j+1]}$. We thereby consider the expectations $E[c_{im} \mid \theta^{[j]}, d]$ computed in the E-step:

$$\mu_{m,k}^{[j+1]} = \frac{1}{\beta} \cdot \sum_{t=(k-1)\cdot\beta+1}^{k\cdot\beta} \frac{\sum_{i=1}^{N} E[c_{im} \mid \theta^{[j]}, d] x_{i,t}}{\sum_{i=1}^{N} E[c_{im} \mid \theta^{[j]}, d]}. \tag{6}$$

2.3 Estimating the Number of Model Components

Since in general the correct number of motion patterns is not known in advance, we need to determine this quantity during the learning phase. If the number of motion patterns is wrong, we can distinguish two different situations. First, if there are too few motion patterns there must be trajectories, that are not explained well by any of the current motion patterns. On the other hand, if there are too many motion patterns then there must be trajectories that are explained well by different model components. Thus, whenever the EM algorithm has converged, we check whether the overall data likelihood can be improved by increasing or decreasing the number of model components. To limit the model complexity, during the evaluation we use a penalty term that depends on the number of model components (see [2]). This avoids that our algorithm learns a model that overfits the data, which in the worst case is a model with one motion pattern for every single trajectory. If the maximum number of iterations is reached or if the overall evaluation cannot be improved after increasing and decreasing the model complexity our algorithms stops and returns the model with the best value found so far. In most of the experiments carried out with different data sets our approach correctly clustered the trajectories into the corresponding categories.

2.4 Laser-based Data Acquisition

The EM-based learning procedure has been implemented for data acquired with laser-range finders. To acquire the data we used several laser-range scanners which were installed in the environment so that the relevant parts of the environment were covered. First, to identify persons in the laser data our system extracts features which are local minima in the range scans that come from the legs of persons. Additionally, it considers changes in consecutive scans to more reliably identify the moving people. To keep track of a person, we use a Kalman filter. The state x_r of a person at time step r is represented by a vector $[x, y, \delta x, \delta y]'$. Whereas x and y represent the position of the person, the terms δx and δy represent the velocity of the person in x- and y-direction. Accordingly, the prediction is carried out by the equation:

$$x^-_{r+1} \quad = \quad \begin{vmatrix} 1 & 0 & t_r & 0 \\ 0 & 1 & 0 & t_r \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} x_r \tag{7}$$

where t_r is the time elapsed between the measurement z_{r+1} and z_r. Usually, sensors only give the position of an object. Since the laser range sensor does not provide the velocities δx and δy, which are also part of our state space, the measurement matrix projects onto the first two components of the state space. Accordingly, the predicted measurement at step $r + 1$ is:

$$z^-_{r+1} \quad = \quad \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{vmatrix} x^-_{r+1}. \tag{8}$$

In a second step we identify the resting places and perform a segmentation of the data into different slices in which the person moves. Finally, we compute the trajectories, i.e. the sequence of positions covered by the person during that motion. When computing these trajectories, we ignore positions which lie closer than 15cm to each other.

2.5 Deriving Hidden Markov Models from Learned Intentions

Once the intentions of the persons have been learned, we can easily derive Hidden Markov Models to estimate their positions. To achieve this, we distinguish two types of nodes. The first class are the initial and final nodes that correspond to the resting places. To connect these nodes we introduce so-called intermediate nodes which lie on the learned motion patterns. In our current system we use a sequence of L_m intermediate nodes $\nu^1_m, \ldots, \nu^{L_m}_m$ for each intention θ_m. The intermediate nodes are distributed over θ_m such that the distance between two consecutive nodes is $\Delta_\nu = 50\text{cm}$. Given this equidistant distribution of the sub-nodes and assuming a constant speed v with standard deviation σ_v of the person, the transition probabilities of this HMM depend on the length Δ_t of the time interval between consecutive updates of the HMM as well as on v and σ_v. In our current system, this value is set to $\Delta_t = 0.5$secs. Accordingly, we compute the probability that the person will be in node ν'_m given it is currently in ν_m and given that the time Δ_t has elapsed as:

$$p(\nu'_m \mid \nu_m, \Delta_t) = \int_{\nu'_m - \frac{\Delta_\nu}{2}}^{\nu'_m + \frac{\Delta_\nu}{2}} \mathcal{N}(\nu_m + v \cdot \Delta_t, \sigma_v, x)\, dx. \tag{9}$$

Here $\mathcal{N}(\nu_m + v \cdot \Delta_t, \sigma_v, x)$ is the value of the Gaussian with mean $\nu_m + v \cdot \Delta_t$ and standard deviation σ_v at position x. The transition probabilities for the resting places are computed based on a statistics about the average time period which elapses before the person starts this particular motion behavior after staying in the corresponding resting area.

Please note that the resulting model can be regarded as a two-level Abstract Hidden Markov Model [3]. Whereas the higher-level goals of this AHMM correspond to the resting places of the person, the lower-level goals are the nodes along the paths to the high-level goals.

2.6 An Application Example

To see how our EM-based learning procedure works in practice please consider Figure 1. In this example, a model for nine trajectories with three different intentions has to be learned. The leftmost image shows the initial model (the means of the three model components are indicated by circles). In the next two images one can see the evolution of the model components. The fourth image shows the model components after convergence of the EM algorithm. As can be seen, the trajectories are approximated quite well by the corresponding motion patterns. Finally, the rightmost picture shows the HMM derived from these motion patterns. The different resting places are indicated by rectangles and numbers.

3 Experimental Results

The technique described above has been implemented and evaluated using data acquired in an unmodified office environment. The experiments described in this section are de-

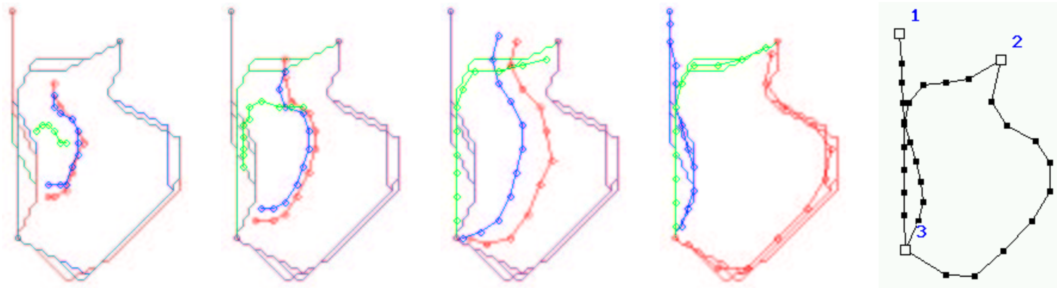

Figure 1: Trajectories of three different intentions and evolution of the motion patterns during the EM algorithm (images 1-4) and resulting HMM (rightmost image).

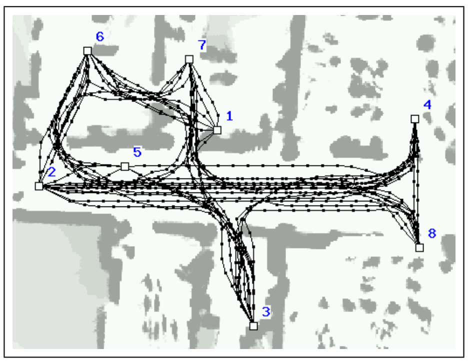

Figure 2: Hidden Markov Model derived from learned intentions.

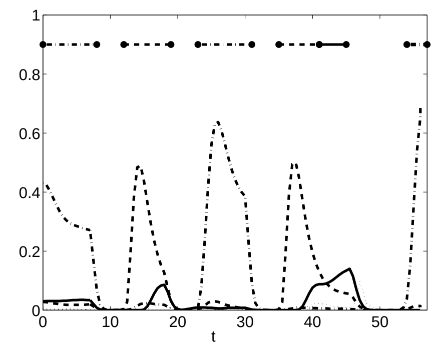

Figure 3: Evolution of the probability for a person of being in different resting areas over the time.

signed to illustrate that the approach can learn complex motion behaviors in a typical office environment. We furthermore demonstrate that the resulting models can be used to robustly estimate the positions of persons. Additionally, we compare the performance of the models learned by our algorithm to that of a standard HMM. Finally, we present an extension which allows the system to deal with multiple persons.

3.1 Learning Intentions in an Office Environment

To evaluate our approach, we applied it to data recorded over two hours in our office environment. During the acquisition phase the average speed of the person was v=107 cm/sec with a standard deviation σ_v=25 cm/sec. From the resulting data our system extracted 129 trajectories which were successfully clustered into 49 different intentions. The resulting Hidden Markov Model is shown in Figure 2.

3.2 Tracking a Single Person

To analyze the applicability of the learned HMM for the prediction of the locations of a person, we used our mobile robot Albert, which is a B21r platform equipped with a laser range scanner. While the robot was moving along the corridor of our department with speed up to 40cm/sec, its task was to maintain a belief about the position of the person.

To incorporate observations into the HMM we apply the recursive Bayesian update scheme:

$$p(\nu \mid z_1, \ldots, z_R) = \alpha \cdot p(z_R \mid \nu) \cdot p(\nu \mid z_1, \ldots, z_{R-1}) \quad (10)$$

Thereby the likelihood $p(z_r \mid \nu)$ of an observation z_r given the state ν is computed using a Gaussian distribution which depends on both, the variance in the current estimate of the tracking system and the variance σ used during the learning of the intentions.

Figure 3 plots for different resting areas the probability that the person stays in this particular place. Whereas the x-axis represents the individual time steps, the y-axis indicates the probability. The graph also includes the ground truth, which is indicated by the corresponding horizontal line-pattern at the .9 level. As can be seen from the figure, the system can reliably determine the current position of the person. During this experiment it predicted the correct place of the person in 93% of the time.

3.3 A Comparison to Standard HMMs

The second experiment is designed to demonstrate that an HMM that takes into account the people's intentions allows a better prediction than a standard HMM that is directly generated from the observed trajectories of the persons and that does not take into a account the clustered trajectories. To evaluate the performance of the two different approaches we chose two motion patterns from those depicted in Figure 2. The first pattern is the one leading from resting place 7 via the office containing resting place 6 to the staying area 2. The second one is the motion pattern between the places 6 and 5. We defined a standard HMM over the possible states of the person in the $\langle x, y, dx, dy \rangle$ space where x and y were discretized in 15cm patches; dx and dy encode 9 possible incremental moves per cell. The transition probabilities were learned from the trajectories corresponding to both motion patterns by counting. We randomly chose a position along the trajectories of both patterns as the observed position of the person. The states of the HMM were initialized according to the observation model (see Section 3.2). After convergence of the HMM we measured the likelihood of the final destination. We compared this value to those obtained by the HMM generated by our algorithm for the trajectories corresponding to these two intentions. We repeated this experiment for different locations along the trajectories of both patterns and determined the average probability of the true goal location. Whereas we obtained an average of .74 with our model, the corresponding value of the standard HMM is .56. This illustrates that our model leads to better results and that the standard independence assumption of HMMs is generally not justified in this application domain. Please note that similar observations have been reported by Murphy [11]. In contrast to a standard HMM our model automatically chooses the transitions that correspond to the actual intention of the person.

3.4 Estimating the Locations of Multiple Persons

The final experiment described in this section is designed to illustrate that our models can also be used to maintain beliefs about multiple persons. The major difference to the situation with a single person is that we have to be able to represent the beliefs for the individual persons and to detect multiple persons in the observations. In our current system we learn an individual HMM for every person.

To track multiple persons in the range scans, we apply independent Kalman filters, one for each feature. To solve the data association problem, we apply a nearest neighbor approach, i.e. we update a filter using the observation z_{r+1} that is closest to z^-_{r+1}. New filters are introduced for observations from which all predictions are too far away. Furthermore, filters are removed if no corresponding feature can be found for one second.

We also need to be able to identify a person in order to appropriately update the belief about the location of that person. To achieve this we additionally employ the vision system of our robot. To identify a person, we proceed as follows: Every time the laser-based people tracker detects a person in the field of view of the camera, an image is collected and following three steps are applied:

1. *Segmentation:* The size of a rectangular area of the image containing the person is determined.
2. *Feature extraction:* We compute a color histogram for the area selected in the previous step.
3. *Database matching:* To determine the likelihood of a particular person, we compare the histogram computed in step 2 to all prototypes existing in the database.

To determine the area in the image corresponding to a feature detected by the laser tracking system, we rely on an accu-

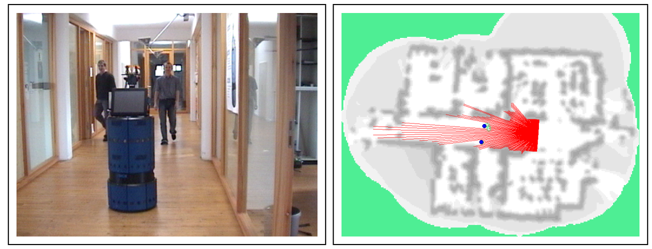

Figure 4: Typical scene with two persons walking along the corridor (left image) and corresponding estimate of the laser-based people tracking system (right image).

rate calibration between the camera and the laser and we use a perspective projection to map the 3D position of the person in world coordinates to 2D image coordinates. Whereas color histograms are robust with respect to translation, rotation, scale and to any kind of geometric distortions they are sensitive to varying lighting conditions. To handle this problem we consider the HSV (Hue-Saturation-Value) color space. In this color model the intensity factor can be separated so that its influence is reduced. In our current system we simply ignore this factor. Throughout all our experiments we could not find any evidence that this factor negatively affected the performance of the system. The image database is created beforehand. For each person it contains one histogram which is built from 20 images.

To compare a given query histogram I with a prototype M in the database we use normalized intersection norm $H(I, M)$ [16]. This quantity can be computed as:

$$H(I, M) \quad = \quad \frac{\sum_{j=1}^{n} \min(I_j, M_j)}{\sum_{j=1}^{n} M_j}, \tag{11}$$

where I and M are color histograms both having n bins. One advantage of this norm is that it also allows to compare partial views, i.e. when the person is close to the camera and only a part of it is visible.

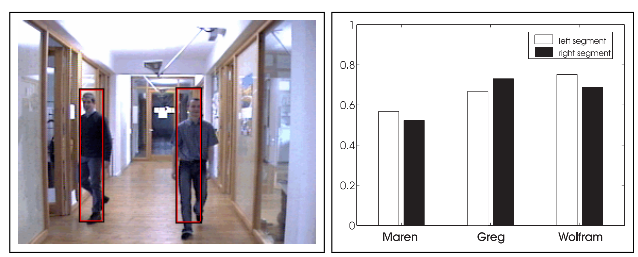

Figure 5: Segmentation of the two persons from the image grabbed with the camera of the robot (left image) and similarity of these segments to the data base prototypes (right image).

To incorporate the similarity measure provided by the vision system into the HMM of the person π, we simply multiply the likelihoods provided by the laser tracking system with the similarity measure $H(I_s, M_\pi)$ of the query histogram I_s for the segment s and the data base prototype M_π for person

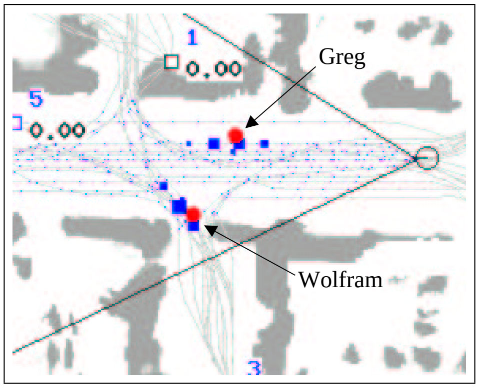

Figure 6: Resulting posterior after incorporating the two segments shown in Figure 5 into the belief over Wolfram's position.

π. If the current estimate of the laser tracker is not in the field of view of the camera we simply update the HMM for all persons as we do in the case in which we track a single person only.

As an application example consider the situation depicted in the left image of Figure 4. In this particular situation two persons (Wolfram and Greg) are walking along the corridor within the perceptual field of the robot. The right image of Figure 4 shows the estimate of the laser-based people tracking system at the same point in time. The corresponding image obtained with the robot's camera is shown in the left image of Figure 5. Also shown there are the two segments of the image that correspond to the two persons detected with the laser. The right image of this figure plots the similarities of the two segments to the individual prototypes stored in the data base. Finally, Figure 6 depicts the HMM for Wolfram (who is the left person in Figure 5). As can be seen, the probabilities indicated by the size of the rectangles are slightly higher for the states that correspond to Wolfram's true location. Throughout this experiment the robot was able to predict the correct location of the persons in 79% of all cases.

4 Conclusions

In this paper we have presented a method for learning and utilizing motion behaviors of persons. Our approach applies the EM-algorithm to cluster trajectories recorded with laser range sensors into a collection of motion patterns, each corresponding to a possible intention of a person. From these motion patterns we automatically derive an HMM that can be used to predict the positions of persons in their environment.

Our approach has been implemented and applied successfully to trajectories recorded in a typical office environment. Practical experiments demonstrate that our method is able to learn typical motion behaviors of persons and to reliably use them for state estimation. The experiments have been carried out using a mobile robot equipped with a laser-range sensor and a vision system. We have furthermore presented experiments indicating that standard HMMs directly learned from the same input data are less predictive than our models.

References

[1] H. Asoh, S. Hayamizu, I. Hara, Y. Motomura, S. Akaho, and T. Matsui. Socially embedded learning of office-conversant robot Jijo-2. In *Proc. of the Int. Joint Conference on Artificial Intelligence (IJCAI)*, 1997.

[2] M. Bennewitz, W. Burgard, and S. Thrun. Using EM to learn motion behaviors of persons with mobile robots. In *Proc. of the Int. Conference on Intelligent Robots and Systems (IROS)*, 2002.

[3] H. Bui, S. Venkatesh, and G. West. Tracking and surveillance in wide-area spatial environments using the Abstract Hidden Markov Model. *Intl. J. of Pattern Rec. and AI*, 2001.

[4] W. Burgard, A.B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1-2), 1999.

[5] H. Endres, W. Feiten, and G. Lawitzky. Field test of a navigation system: Autonomous cleaning in supermarkets. In *Proc. of the Int. Conference on Robotics & Automation (ICRA)*, 1998.

[6] S. King and C. Weiman. Helpmate autonomous mobile robot navigation system. In *Proc. of the SPIE Conference on Mobile Robots*, pages 190–198, Boston, MA, November 1990. Volume 2352.

[7] B. Kluge, C. Köhler, and E. Prassler. Fast and robust tracking of multiple moving objects with a laser range finder. In *Proc. of the IEEE Int. Conference on Robotics & Automation (ICRA)*, 2001.

[8] G. Lacey and K. Dawson-Howe. The application of robotics to a mobility aid for the elderly blind. *Journal of Robotics and Autonomous Systems (RAS)*, 23:245–252, 1998.

[9] S. M. Lavalle, H. H. Gonzalez-Banos, G. Becker, and J.-C. Latombe. Motion strategies for maintaining visibility of a moving target. In *Proc. of the IEEE Int. Conference on Robotics & Automation (ICRA)*, 1997.

[10] G.J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley Series in Probability and Statistics, 1997.

[11] K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference, and Learning*. PhD thesis, University of California, Berkely, 2002.

[12] N. Roy, G. Baltus, D. Fox, F. Gemperle, J. Goetz, T. Hirsch, D. Magaritis, M. Montemerlo, J. Pineau, Schulte J., and S. Thrun. Towards personal service robots for the elderly. In *Proc. of the Workshop on Interactive Robotics and Entertainment*, 2000.

[13] C. Schaeffer and T. May. Care-o-bot - a system for assisting elderly or disabled persons in home environments. In *Assistive technology on the threshold of the new millenium*. IOS Press, Amsterdam, 1999.

[14] R.D. Schraft and G. Schmierer. *Service Robots*. Springer Verlag, 1998.

[15] D. Schulz, W. Burgard, D. Fox, and A.B. Cremers. Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *Proc. of the Int. Conference on Robotics & Automation (ICRA)*, 2001.

[16] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1), 1991.

[17] S. Tadokoro, M. Hayashi, Y. Manabe, Y. Nakami, and T. Takamori. On motion planning of mobile robots which coexist and cooperate with human. In *Proc. of the Int. Conference on Intelligent Robots and Systems (IROS)*, 1995.

[18] Q. Zhu. Hidden Markov model for dynamic obstacle avoidance of mobile robot navigation. *IEEE Transactions on Robotics and Automation*, 7(3), 1991.