Utilizing Learned Motion Patterns to Robustly Track Persons

Maren Bennewitz[†] Wolfram Burgard[†] Grzegorz Cielniak[‡]

[†]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 the position of the persons in its environment or to improve its 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 typical motion patterns. 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

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 [8], museums [2], office buildings [1], and department stores [5] where they perform various services e.g., deliver, educate, entertain [17] or assist people [11].

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 by deriving an HMM [15] from the learned motion patterns.

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 collection of trajectories $s = \{s_1, \ldots, s_N\}$ 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 s_i consists of a sequence $s_i = \{s_{i,1}, s_{i,2}, \ldots, s_{i,T_i}\}$ of positions $s_{i,t}$. Accordingly, $s_{i,1}$ is the resting place the person leaves and s_{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.

2.1. Motion Patterns

We begin with the description of our model of motion patterns, which is subsequently estimated from data using EM. A motion pattern denoted as θ_m with $1 \leq m \leq M$ is represented by K probability distributions $p(x \mid \theta_{m,k})$ where M is the number of different types of motion patterns a person is engaged in.

Throughout this paper we assume that the input to our algorithm consists of trajectories which have the same number of observed positions, i.e., that $T_i = T$ for all i. To achieve this, we transform the trajectories in s into a set d of N trajectories such that each $d_i = \{d_{i,1}, d_{i,2}, \ldots, d_{i,T}\}$ has a fixed length T and is obtained from s_i by a linear interpolation. The length T of these trajectories corresponds to the maximum length of the input trajectories in s. The learning algorithm described below operates solely on $d_1, \ldots, d_N$ and does not take into account the velocities of the persons during the learning phase. In our experiments, we never found evidence that the linear interpolation led to wrong results or that the walking speed of a person depends on its intention. Note that one can easily extend our algorithm to also incorporate the velocities by introducing further dimensions to the state variables.

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. Accordingly, the application of EM leads to an extension of the fuzzy k-Means Algorithm (see e.g. [4]) to trajectories. 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 our case this Q-function is factored as follows:

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

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:

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

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:

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

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 $\mu^{[j+1]}_{m,k}$. We thereby consider the expectations $E[c_{im} \mid \theta^{[j]}, d]$ computed in the E-step:

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

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. 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 [20].

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 Motion Patterns

Once the motion patterns 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 $\nu_m^1, \ldots, \nu_m^{L_m}$ for each motion pattern θ_m. The intermediate nodes are distributed over θ_m such that the distance between two consecutive nodes is $\Delta_\nu = 50$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:

$$p(\nu'_m \mid \nu_m, \Delta_t) = \int_{\nu'_m - \frac{\Delta_\nu}{2}}^{\nu'_m + \frac{\Delta_\nu}{2}} \mathcal{N}(\nu_m + v \cdot \Delta_t, \sigma_v, x)\, dx. \quad (7)$$

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. We currently define the same transition probabilities for all intermediate nodes and assume that the persons are moving at constant speed. The transition probabilities for the resting places are computed based on two statistics. The first one is a statistics about the average time period which elapses before the person starts to move after staying at the corresponding resting place. Furthermore, we count how often a person starts to move on a particular trajectory after staying at the resting places and this way determine the transition probabilities for the nodes corresponding to the resting places.

To update such an HMM based on sensory input we distinguish two different situations, namely when we are tracking a single person and when we are tracking multiple persons.

2.5.1 Keeping Track of a Single Person

Let us first consider the case that we are tracking a single person. We apply the well-known recursive Bayesian update scheme:

$$p(\nu \mid z_1, \ldots, z_R) = \alpha \cdot p(z_R \mid \nu) \cdot p(\nu \mid z_1, \ldots, z_{R-1}). \quad (8)$$

Here α is a normalizer and the observation z_r is either the position of a person provided by the laser-based people tracking system or the information that no person was detected. To compute the likelihood of an observation z given the state ν we have to distinguish four different cases. When the observation is the position z_{xy} of a person and the robot is at position r_{xy} we get:

$$p(z = z_{xy} \mid \nu, r_{xy}) = \begin{cases} \mathcal{N}(0, \sigma_z, \|z_{xy} - \nu\|) & \text{if } \nu \text{ is visible from } r_{xy} \\ c_1 & \text{otherwise.} \end{cases} \quad (9)$$

Here σ_z is the variance in observations of persons and c_1 is a constant that is determined by counting how often a person detection around z_{xy} is reported even if actually no person is in the vicinity of ν.

For the case that no person is detected we define the likelihood of this observation as:

$$p(z = \text{no Person} \mid \nu, r_{xy}) = \begin{cases} c_2 & \text{if } \nu \text{ is visible from } r_{xy} \\ c_3 & \text{otherwise.} \end{cases} \quad (10)$$

Here the constant c_2 stands for the cases in which the people tracker fails to detect a person even if there is one in the vicinity of ν. c_3 is proportional to the probability that the people tracker gives the correct information that no person is in the sensor range. Again the values for c_2 and c_3 can be determined by counting. The constants are needed in order to prevent that all probabilities vanish from the corresponding nodes in the case that the people tracker fails to track a person or, in the other case, a false positive detection occurs.

2.5.2 Keeping Track of Multiple Persons

To keep track of multiple persons in an environment one in principle would have to maintain a belief over the joint state space of all persons. This approach, however, is usually not feasible since the complexity of the state estimation problem grows exponentially in the number of persons or dimensions of the state space. Additionally, learning the joint transition probability distribution would require a huge amount of training data. We therefore approximate the posterior by factorizing the belief over the joint state space and by considering independent beliefs over the states of all persons. With our current system we first compute an individual HMM for every person. To maintain the individual beliefs, however, we need to be able to update the HMMs for the persons based on observations made by the robot, which requires the ability to reliably keep track of persons and to identify them. To achieve this, our current systems combines laser and vision information.

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 and learn an image database beforehand. For each person this database contains one histogram which is built from 20 images. 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 the following three steps are applied:

1. *Segmentation:* The size of a rectangular area of the image containing the person is determined. To determine the area in the image corresponding to a feature detected by the laser tracking system, we rely on an accurate calibration between the camera and the laser. We use a perspective projection to map the 3D position of the person in world coordinates to 2D image coordinates.

2. *Feature extraction:* We compute a color histogram for the area selected in the previous step. 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.

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 compare a given query histogram I with a prototype M in the database we use the normalized intersection norm $H(I, M)$ [19]. This quantity can be computed as:

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

   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.

To incorporate the vision information into the update procedure of the HMM, we apply the following formula:

$$p(\nu \mid z_1, \ldots, z_R) = \alpha \cdot \begin{cases} p(z_R \mid \nu) \cdot p(\nu \mid z_1, \ldots, z_{R-1}) \\ \quad \text{if } z_r \text{ is only a range measurement} \\ p(z_R \mid \nu) \cdot H(I, M) \cdot p(\nu \mid z_1, \ldots, z_{R-1}) \\ \quad \text{if } z_r \text{ is a range and vision measurement} \end{cases} \quad (12)$$

Whenever a person is not in the field of view of the camera but the robot perceives laser range information about the position of a person we update all HMMs using the range data as we do when we track a single person only (Equations 8 to 10). If, however, the measurement includes

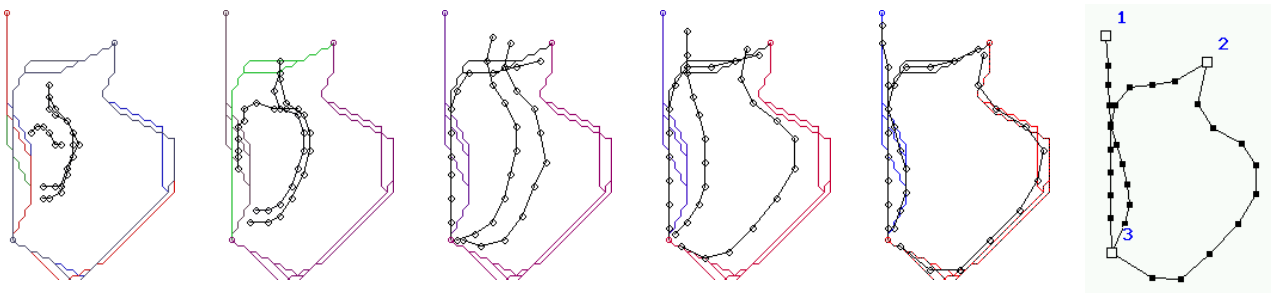

Figure 1: Trajectories of three different motion patterns and evolution of the motion patterns during the EM algorithm (images 1-4) and resulting HMM (rightmost image).

range and vision information, we update each HMM using the range information but also proportional to the likelihood that this measurement was reflected by the person corresponding to the particular HMM. Thus, to integrate 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 $H(I_j, M_\pi)$ of the query histogram I_j for the segment j and the data base prototype M_π for person π.

3. Experimental Results

The experiments described in this section are designed to illustrate that our algorithm can learn complex motion behaviors of persons in different types of environments. We also demonstrate that the HMMs derived from the learned motion patterns allow a robust estimation of the positions of persons. Finally, we present an experiment illustrating that our approach yields better estimates than a standard Hidden Markov Model directly learned from the input data.

3.1 Learning Example

To see how our EM-based learning procedure works in practice please consider Figure 1. In this example, a model for nine trajectories of three different motion patterns 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.

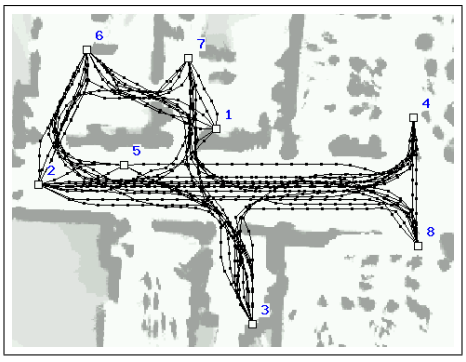

Figure 2: Hidden Markov Model derived from learned motion patterns.

3.2. Learning Motion Patterns 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 motion patterns. The resulting HMM as well as identified resting places are shown in Figure 2.

3.3. Tracking a Single Person

The first experiment is designed to illustrate that our approach is able to reliably estimate the position of a person in its environment. In this experiment, a single person was moving in our department and the task of the robot, which itself did not move, was to estimate the positions of this person. Especially, we were interested in the probability that the person stayed at the correct resting place.

Figure 3 shows a scene overview (left hand side) for a part of an experiment in which a single person was moving through the environment. The robot could only cover a part

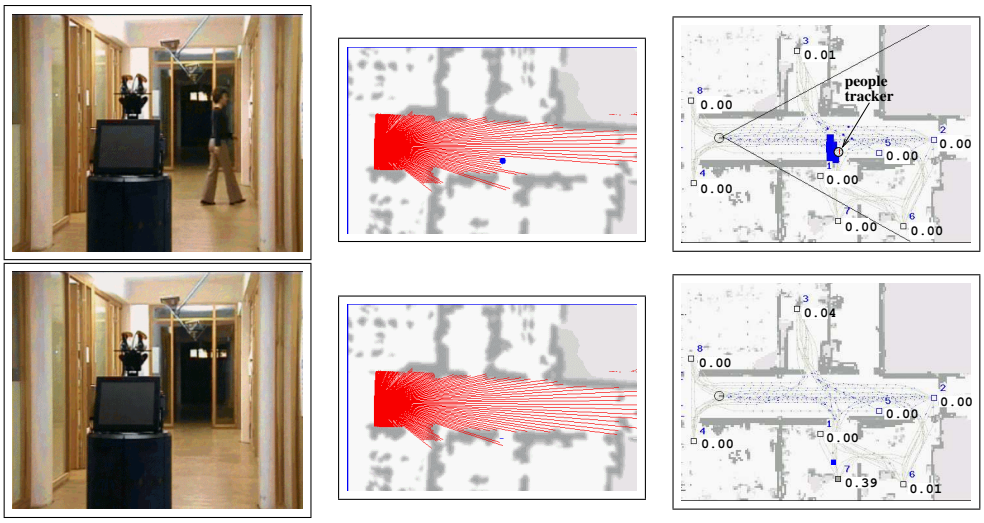

Figure 3: Albert tracking a person while it is moving through the environment. The center images depict the results of the laser-based tracking system. The images on the right show the evolution of the belief over the position of the person.

of the environment with its sensors (which was mainly the corridor as indicated) but even though it was able to maintain and update the belief about the position of the person. The center images of the figure depict the results from the laser-based people tracking system and the images on the right hand side show the evolution of the HMM. In this case we did not use vision information because we assumed only one person was moving in the environment. In the HMM the grey dot corresponds to the position of the person provided by the laser tracking system. The size of the squares of the states of the HMM represents the probability that the person is currently in the corresponding state, similarly the resting places are labeled with the probability that the person stays currently at this particular place. In the images depicted in the first row of this figure the robot observed the person walking through the corridor. Then the person entered a room and walked outside the field of view of the robot. According to the transition probabilities of the HMM, which models the typical behavior of the person, most of the probabilities "wander" to resting place 7 (second row of Figure 3).

Figure 4 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.

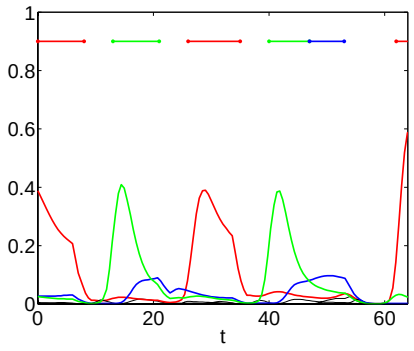

Figure 4: Evolution of the probability of the person to be at the different resting places over the time. The ground truth is indicated by the horizontal line-pattern at the .9 level.

3.4. Estimating the Locations of Multiple Persons

As an application example consider the situation depicted in the left image of Figure 5. 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 5 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 6. 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 7 depicts the

HMM for Wolfram (who is the left person in Figure 6). 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.

3.5. A Comparison to Standard HMMs

The final experiment is designed to demonstrate that an HMM that takes into account the motion behaviors of persons allows a better prediction than a standard HMM that is directly generated from the observed trajectories of the persons. 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 15 cm 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 a person. The states of the HMM were initialized according to the observation model. After convergence of the HMM we measured the likelihood of the true destination. We compared the results to those obtained with the HMM for the two corresponding motion patterns as they are generated by our algorithm. 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 because in contrast to a standard HMM our model is able to differentiate between various motion behaviors and automatically chooses the correct transitions. Note that this experiment was carried out using only the range information so that both HMMs used exactly the same input.

4. Related Work

A variety of laser-based techniques has been developed for tracking people [18, 12]. These approaches assume that the models of the motion behavior of the objects to be tracked are given. Our approach, in contrast, is able to learn such models and to use the learned models for the long-term prediction of motions of persons. Kruse and Wahl [9] use a camera system mounted at the ceiling to track persons in the environment and to learn where the people usually walk in their workspace. Johnson and Hogg [7] learn probability density functions (pdfs) of typical object trajectories to

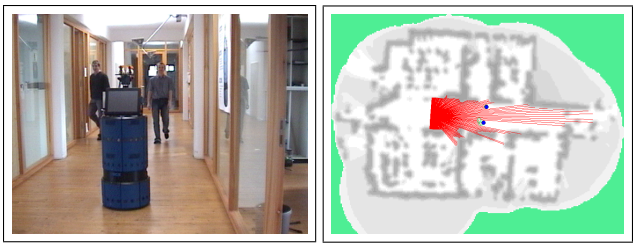

Figure 5: Typical scene with two persons walking along the corridor (left image) and corresponding estimate of the laser-based people tracking system (right image).

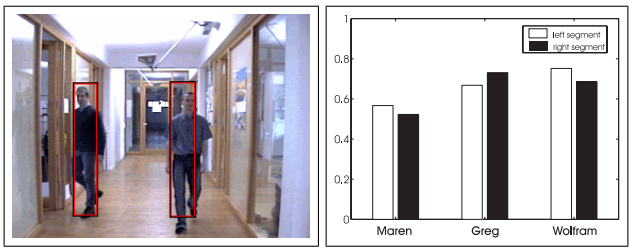

Figure 6: Segmentation of the two persons from the image grabbed with the camera of the robot (left image) and similarity of these segments to the data base prototypes (right image).

detect atypical behaviors. Compared to the work presented here, their approach lacks a technique to estimate the number of different behaviors. The goal of the work by Stauffer and Grimson [3] is also to detect unusual events. They learn codebooks of a given number of prototypes. Rosales and Sclaroff [16] analyze 3D trajectories to learn typical classes of actions like walking, running, and biking. Oliver et al. [14] use data obtained from various sensors as input to an Layered HMM and infer the state of a user's activity. Galata et al. [6] use Variable Length Markov Models (VLMMs) to model structured behaviors. One problem to be solved in the context of VLMMs is the estimation of the optimal size of the time window in order to correctly predict the next states. In our approach the relevant steps in the past are generated automatically by the clustering procedure. Nguyen et al. [13] recently proposed to use an Abstract Hidden Markov mEmory Model (AHMEM) to infer intentions of persons. The idea of an AHMEM is to model higher level behaviors by a stochastic sequence of more simple behaviors at the lower levels. The authors apply an EM-based learning method for labeled trajectories to determine the transition probabilities for the states at the lowest level (grid cells) and assume that the landmarks the persons want to approach are given. Our approach in contrast applies an unsupervised clustering method to the observed

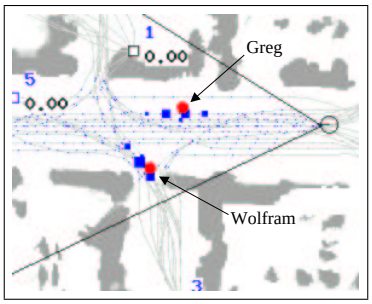

Figure 7: Posterior after incorporating the two segments shown in Figure 6 into the belief over Wolfram's position.

trajectories and is also able to automatically infer resting places which correspond to the landmarks in the AHMEM.

5. Conclusions

In this paper we 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 motion pattern of a person. From these learned motion patterns we automatically derive an HMM that can be used to predict the positions of persons in their environments. We presented techniques to update the resulting HMMs using laser range and vision information.

Our approach has been implemented and applied successfully to data recorded in a typical office environment. In practical experiments we demonstrated that our method is able to use learned motion models to reliably predict states of multiple persons. The experiments have been carried out using a mobile robot equipped with a laser-range sensor and a vision system. We 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 International Joint Conference on Artificial Intelligence (IJCAI)*, 1997.

[2] 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.

[3] Stauffer. C. and W.E.L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–757, 2000.

[4] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley-Interscience, 2001.

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

[6] N. Galata, A.and Johnson and D. Hogg. Learning variable length markov models of behaviour. *Computer Vision and Image Understanding (CVIU) Journal*, 81(3), 2001.

[7] N. Johnson and D. Hogg. Learning the distribution of object trajectories for event recognition. In *British Machine Vision Conference*, 1995.

[8] 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.

[9] F. Kruse, E. und Wahl. Camera-based monitoring system for mobile robot guidance. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1248–1253, 1998.

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

[11] M. Montemerlo, J. Pineau, N. Roy, S. Thrun, and V. Verma. Experiences with a mobile robotic guide for the elderly. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 2002.

[12] M. Montemerlo, S. Thrun, and W. Whittaker. Conditional particle filters for simultaneous mobile robot localization and people-tracking. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 2002.

[13] N.T. Nguyen, H.H. Bui, S. Venkatesh, and G. West. Recognising and monitoring high-level behaviours in complex spatial environments. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003.

[14] N. Oliver, E. Horvitz, and A. Garg. Layered representations for learning and inferring office activity from multiple sensory channels. In *Proc. of the International Conference on Multimodal Interfaces (ICMI)*, 2002.

[15] L.R. Rabiner and B.H. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(1):4–16, 1986.

[16] R. Rosales and S. Sclaroff. A framework for heading-guided recognition of human activity. *Computer Vision and Image Understanding (CVIU) Journal*, 2003. to appear.

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

[18] 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 IEEE International Conference on Robotics & Automation (ICRA)*, 2001.

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

[20] G. Welch and G. Bishop. An introduction to the Kalman Filter. Technical report, University of North Carolina at Chapel Hill, 1997.