Robotics and Autonomous Systems 62 (2014) 591–602

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Learning intentions for improved human motion prediction

Jos Elfring *, René van de Molengraft, Maarten Steinbuch

Department of Mechanical Engineering, Eindhoven University of Technology, PO Box 513, 5600 MB Eindhoven, The Netherlands

HIGHLIGHTS

- Learning human's goal position from the data using growing hidden Markov models.
- Human motion prediction using social forces-based motion models exploiting estimated goal.
- Extended experimental analysis of both strengths and weaknesses using the existing data set.
- Comparison with benchmark strategy shows significant performance gain.

ARTICLE INFO

Article history:
Received 15 July 2013
Received in revised form
18 December 2013
Accepted 10 January 2014
Available online 22 January 2014

Keywords:
Human motion prediction
Growing hidden Markov models
Intention estimation

ABSTRACT

For many tasks robots need to operate in human populated environments. Human motion prediction is gaining importance since this helps minimizing the hinder robots cause during the execution of these tasks. The concept of social forces defines virtual repelling and attracting forces from and to obstacles and points of interest. These social forces can be used to model typical human movements given an environment and a person's intention. This work shows how such models can exploit typical motion patterns summarized by growing hidden Markov models (GHMMs) that can be learned from data online and without human intervention. An extensive series of experiments shows that exploiting a person's intended position estimated using a GHMM within a social forces based motion model yields a significant performance gain in comparison with the standard constant velocity-based models.

1. Introduction

As service robots are expected to perform tasks in human populated environments, notion of how people move is gaining importance. Occlusions by other humans or objects and movements outside the robot's field of view cause intervals without detections for time scales up to minutes. During such 'blind' periods, a robot must rely on human motion models. In addition, minimizing the hinder for humans during, *e.g.*, navigation or delivery tasks is only possible if robots have information about how people move in the nearby future. Parameterizing pre-defined models is difficult due to the dependency on the environment and the person's intentions and a simple motion model assuming a constant velocity or acceleration is typically not valid over longer periods of time.

To overcome these problems, part of the relevant literature assumes that people, or moving objects in general, tend to follow a limited set of motion patterns given an environment [1,2]. These patterns are assumed to exist among a limited set of positions, *e.g.*, doors, tables, chairs. By learning these patterns, motions can be predicted during periods without detections. Another solution models the interactions between objects and their environment based on the concept of social forces [3]. By estimating attractive and repulsive forces to and from target locations and obstacles, predictions respecting the physical constraints of the environment can be made.

The questions investigated in this work are: (i) how can typical motion patterns be learned and exploited for human motion prediction, and (ii), how can the knowledge about the possibly dynamic environment a robot is operating in be exploited during human motion prediction? Based on these questions, four requirements are formulated.

(R1) Goal prediction must exploit the experience summarized by a collection of observation sequences. Simple linear extrapolation is not sufficient for predictions over time intervals in the order of seconds.
(R2) The number of possible destinations in a given environment usually is limited, *e.g.*, dinner table, office A, desk B, however, the world in between these destinations should not be discretized since infinitely many trajectories are possible dependent on the presence of, *e.g.*, dynamic obstacles or doors that are open or closed.
(R3) The probability of a person moving towards a specific target location can vary over time. For this reason it must be possible to update a human motion model given new observation sequences.

* Corresponding author. Tel.: +31 402473580.
E-mail address: J.Elfring@tue.nl (J. Elfring).

0921-8890/$ – see front matter
http://dx.doi.org/10.1016/j.robot.2014.01.003

Table 1
Summary of the requirements satisfied by related work. Requirements are either fulfilled (✓), not fulfilled (×) or not applicable (□).

Reference	R1	R2	R3	R4
Bennewitz et al. [1]	✓	×	×	✓
Vasquez et al. [2]	✓	×	✓	✓
Luber et al. [3]	×	✓	×	✓
O'Callaghan et al. [4]	×	□	×	✓
Chung et al. [5]	✓	×	×	✓
Tseng et al. [6]	×	✓	×	✓
Yamaguchi et al. [7]	✓	✓	×	×
Rohrmüller et al. [8]	×	✓	×	✓
Kanda et al. [9]	✓	×	×	✓
Foka et al. [10]	✓	✓	×	×
Kitani et al. [11]	✓	×	×	✓
Chen et al. [12]	✓	×	×	✓
Pellegrini et al. [13,14]	×	✓	×	✓
Chung et al. [15]	×	✓	×	✓
Luber et al. [16]	×	✓	×	✓
Ziebart et al. [17]	✓	×	□	□
Required	✓	✓	✓	✓

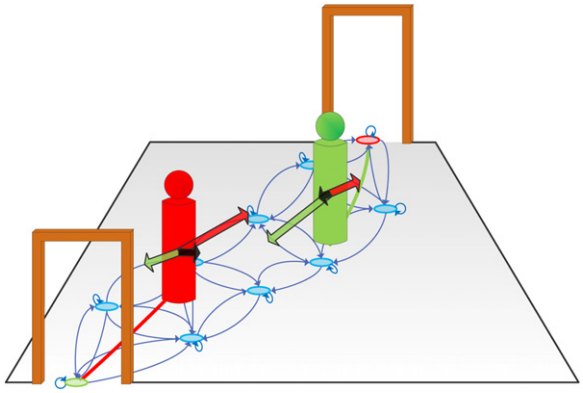

Fig. 1. Simplified visualization of the approach. A person's trajectory can be predicted by incorporating an intended position (colored ellipses) estimated using a hidden Markov model (blue ellipses and arrows) into a motion model based on the concept of social forces (large colored arrows). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

(R4) Updating some human motion model as required in the third requirement must happen autonomously, *i.e.*, without human intervention of any form (see Fig. 1).

2. Related work

Various authors define a finite number of discrete states and estimate a path, using observed trajectories, by a state sequence. In [1] motion patterns are represented by hidden Markov models (HMMs) learned from data clustered using expectation maximization. Although being a good starting point, the discretization of the paths in a finite number of states and the disability to update the model online conflict with requirements R2 and R3. In [2] growing HMMs (GHMMs) are learned. Contrary to [1], both the HMM structure and parameters can be learned online. However, predicting human motions on a discretized world grid contradicts the continuous world requirement R2, as shown in Table 1. In [9] the goal is to let a robot actively approach humans in a shopping mall based on their intention. Human motion primitives such as walking or running are used to predict the human behavior, *e.g.*, stopping at a shop. In [12] trajectories are clustered and summarized by a mean and a left and right boundary. Based on the match between a sequence of observations and each cluster, a credibility level is calculated which is used during prediction. Both works fulfill the requirement of estimating a person's goal, however, during short deviations from the learned trajectories, *e.g.*, because of a person crossing the intended path, the correspondence between the learned and observed trajectory drops and the accuracy of the prediction deteriorates. In addition, the models are not updated online, hence both of the aforementioned methods fail to fulfill the continuous world requirement R2 and lack the ability to update models online as formulated in requirement R3. In [10], linear extrapolation is used for the goal prediction. Potential goals are given by a user and the mismatch between predicted heading direction and goal location is translated into a probability. The prediction method using linear extrapolation is considered overly simplistic and both the need for human input and the disability of online updating the model contradict our requirements, as shown in Table 1. The work of [11] researches activity forecasting which contains 'destination forecasting' as a subproblem. Their approach combines semantic scene understanding with ideas from the optimal control theory. Main drawbacks are the discretized world and the focus on the static part of the environment only.

A second group of work uses the concept of so-called social forces. A person is attracted by its goal, whereas it is repelled by other objects. This adds robustness and enables estimating trajectories in a continuous domain. Important work on this topic is performed by Helbing in, *e.g.*, [18,19]. In [14,13] the forces are based on energy potentials, in [3] these forces are based on existing models for crowd simulation [18,19]. In either of the cases, a constant velocity motion model is used to incorporate the interactions into the motion model of an object by introducing an acceleration term. The authors in [3] estimate a person's goal based on constant velocity extrapolation and show that the model leads to improved tracking performance and up to two times less data association errors. The work of [13] does not deal with intention estimation since defining either the left or the right hand edge of a camera image suffices in their experiments, so both works do not meet the goal prediction requirement R1. Online updating the human motion model is not dealt with in both [3] and [13]. The work of [6] is based on similar ideas, however, stream field based human motion models are used. The forces are calculated at a grid map for efficiency purposes and the persons' goal is estimated using a constant acceleration motion model. This goal estimation is considered too simplistic and online updating is impossible. In [7] an energy function incorporating social effects is used for prediction. The intended position of a person is estimated using a support vector machine trained off-line using past trajectories and possible destinations given by a human user, hence both requirements R3 and R4 are violated. The work of [15] uses a cost function incorporating various social factors. Prediction is performed using a grid map in which transition probabilities are learned from the data. The grid map contradicts the continuous world requirement R1 and the transition probabilities are not learned online. Finally, [16] learn a set of dynamic motion prototypes from the data. Estimating a human's intended position is excluded since the focus is on estimating a person's path in the presence of another person given the starting and end positions of the path.

Interesting work in which prediction is a means for improved robot path planning is [4,8,17]. The authors in [8] use Markov chains for pre-computing transition probabilities in a grid. The probabilities are used to predict people movements, however, the intended goal is not estimated. In [4] a navigational map is learned for each possible location. During an offline learning phase a deviation function describing how humans deviate from the map is learned. The main application is planning paths to given locations hence the intention estimation problem is not dealt with in [4]. In [17] goal-directed trajectories are estimated using a strategy the authors refer to as maximum entropy inverse optimal control.

Strongest feature of this approach in the context of our requirements is the robustness against changes in the environment, main drawbacks are the discretization of the environment into a fixed grid and the lack of modeling the interactions between humans.

None of the related works fulfill all of our requirements. In the next section, the new contribution of this paper will be presented.

3. Contributions

The research presented in this paper uses the concept of social forces, as in [3] as a starting point. This way, there is no need for predicting human trajectories on a discretization of the world. Contrary to the number of paths towards a goal, the number of possible goals given an environment is considered to be limited. For that reason, we tailored the advanced goal estimation method presented in [2] and incorporated it in the human motion models used by [3], which is our main contribution. With this human motion prediction framework, many experiments using a large existing data set have been performed during which both the strengths and weaknesses of the proposed algorithm are analyzed.

The proposed approach contains two steps. First, motion patterns are learned for the purpose of human goal estimation. Section 4 explains how this is done and how the work in [2] was modified and applied. Then in step two, the learned patterns are combined with the social forces based approach. Section 5 explains how this combination is merged into a framework that can be used for human motion prediction. After that, Section 6 presents the experimental results and in Section 7 conclusions are drawn.

4. Learning of statistical motion patterns

The first step of the proposed algorithm is estimating a person's goal using a growing hidden Markov model (GHMM). Section 4.1 explains how the world is discretized for the purpose of goal estimation. Sections 4.2 and 4.3 transform this discretized world into a GHMM. Section 4.4 explains how the parameters of the GHMM are updated and Section 4.5 explains how the intended position of a person is estimated using the GHMM.

4.1. Updating the topological map

This section explains how a set of observed person trajectories is summarized by a topological map. This is a data reduction step that is considered reasonable since the map, and later the GHMM, will be used to estimate the person's goal and not the path towards this goal.

A topological map contains nodes connected by undirected edges. Each node represents a region in the environment and the edges represent the connectivity among nodes, *i.e.*, a node represents a Voronoi region, an edge a Delaunay edge. In [2] the Instantaneous Topological Map (ITM) algorithm [20] is used. Here a modified version is used instead. The algorithm has linear time and memory complexity with respect to the number of nodes in the map.

An observation of a person's position obtained at time t is denoted by the vector $\boldsymbol{O}_t$, an observation sequence up to time t is denoted by the set $\mathcal{O}_{1:t} = \{\boldsymbol{O}_1, \ldots, \boldsymbol{O}_t\}$. Each observation sequence is assumed to originate from one person, hence data association is excluded from this work. For each observation in the sequence $\mathcal{O}_{1:t}$, the topological map is updated in four steps. Fig. 2 visualizes the various steps.

1. For the current observation represented by the cross in Fig. 2(a), find the nodes n and s with the smallest and second smallest Euclidean distance to this observation. The dashed lines indicate the associated Voronoi region for each node, the black solid lines the edges between the nodes. If nodes n and s are not connected, a link is added (red line).
2. Check for each neighbor m of n if node s lies in the Thales sphere[1] through n and m. If yes, as is the case in Fig. 2(b), remove the edge connecting nodes n and m. If not, as is the case in Fig. 2(c), keep all edges. After removing an edge to node m, remove node m if it has no neighbors left.
3. If, as in Fig. 2(d), an observation lies both outside the (dashed) Thales sphere through nodes n and s and outside a (dotted) sphere with radius $e_{\max}$ around node n, a new node is created at the location of the measurement. If, as is the case in Fig. 2(e), nodes n and s are closer than $0.5 \cdot e_{\max}$ (dash-dotted sphere) [20] removes node s, however, here nodes n and s are merged (red node).
4. If no new node is created in one of the previous steps, associate observation $\boldsymbol{O}_t$ with node n. In this work, the $N_{p,\max}$ most recent observations are stored per node. In [20,2] this step is absent hence no measurements are stored. However, the associated observations can be used to estimate the observation probabilities needed in the GHMM later. The updated node position equals the average of all associated points. In the example in Fig. 2(f), the observations associated with the nodes are represented by the crosses, the dashed lines again indicate the Voronoi regions associated with each topological map node and the solid lines represent the edges.

The first two steps are according to the original work in [20], the third and fourth step are modified respectively added since this allows for an improved calculation of the observation probabilities later. The only parameters that have to be set in this algorithm are the radius $e_{\max}$, that can be interpreted as the resolution of the map, and $N_{p,\max}$, which is the size of the sliding window, a measure of the time horizon that is taken into account during the calculations explained in Section 4.2.

4.2. Formulation of the GHMM

In the context of estimating a person's motion, person's trajectories can be modeled by state sequences in which each discrete state represents a region in the real world. A hidden Markov model (HMM) can be used to model and predict a person's trajectory by a state evolving stochastically at discrete time steps t. An important property of HMMs is that they summarize the continuous state space representing all possible states, in this work the set of all possible locations, by a finite number of discrete states. The number of states therefore depends on the number of locations covered by the different trajectories, rather than the number of persons. The main assumption underlying HMMs is the Markov property: the complete history of the state at time t is adequately summarized by the state at the previous time step $t - 1$ only.

An HMM can be interpreted as being a graph containing nodes connected by directed edges. Each node represents one of the discrete states mentioned before, whereas the edges represent the possible transitions between these states. The existence of edges between nodes indicates that it is possible to move from one state to another within one time step. In HMMs the states itself are hidden, *i.e.*, they cannot be observed directly. Instead, an observation related to this state is perceived.

In this work, the state at time t is denoted by the vector $\boldsymbol{S}_t$. Each state is associated with a state prior which represents the probability of a trajectory starting in this state. Furthermore, a state is associated with an observation probability. For a state i, denoted by $\boldsymbol{S}_t^i$, the observation probability function determines the probability of observing a measurement $\boldsymbol{O}_t$ under the assumption of being in state i. The edges in an HMM are represented by

[1] The Thales sphere through two points A and B is the circle with diameter AB through both points A and B.

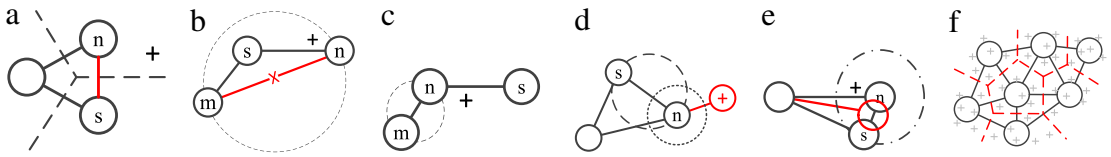

Fig. 2. Four steps of the modified instantaneous topological mapping algorithm. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

transition probabilities representing the probability of evolving from one state to another in one time step.

A growing HMM (GHMM) is an HMM with a varying number of states connected by a varying number of directed edges. The number of states after processing k observation sequences is $N_{s,k}$ and the full GHMM is parameterized by:

$$\lambda = \{\pi_k, A_k, b_k\}, \tag{1}$$

in which π_k is the set of $N_{s,k}$ prior probabilities for the $N_{s,k}$ states and A_k is the set of transition probabilities among the states. The transition probability from state i at time $t-1$ to the state j at time step t is denoted as $p(\boldsymbol{S}_t^j \mid \boldsymbol{S}_{t-1}^i) = a_{i,j}$. Finally, b_k parameterizes the observation probabilities $p(\boldsymbol{O}_t \mid \boldsymbol{S}_t^i)$, where $i = 1, \ldots, N_{s,k}$.

In this work, all observation probabilities will be represented by Gaussian distributions, hence $p(\boldsymbol{O}_t \mid \boldsymbol{S}_t^i) = \mathcal{N}^i(\boldsymbol{O}_t;\ \boldsymbol{\mu}_k^i, \Sigma_k^i)$ and therefore, $b_k = \{\boldsymbol{\mu}_k^1, \ldots, \boldsymbol{\mu}_k^{N_{s,k}}, \Sigma_k^1, \ldots, \Sigma_k^{N_{s,k}}\}$. Each mean $\boldsymbol{\mu}_k^i$ of a Gaussian distribution is associated with a node center in the topological map and equals the corresponding node position. The covariance matrix Σ_k^i of the i'th Gaussian distribution is set equal to the covariance of the set of associated points up to some constant scaling factor. Whenever possible, the subscript k is dropped for the sake of notational simplicity.

As explained in Section 4.1, each node center in the topological map is associated with a unique location. The GHMM has a state vector $\boldsymbol{S}_t = \begin{bmatrix}\boldsymbol{x} & \boldsymbol{x}'\end{bmatrix}^T$, in which $\boldsymbol{x}$ is the position vector of the state and $\boldsymbol{x}'$ is the vector representing the associated intended position. Both the current and intended positions are equal to the position of one of the topological map nodes. By combining the current position with the intended position in the GHMM state, different GHMM states can be associated with the same node in the topological map, *i.e.*, persons at the same location are associated with different states if their intended locations differ. Mathematically, the effect of adding the intended position to the GHMM state is the appearance of distinctive manifolds in the GHMM. Each manifold is associated with one unique intended position and describes how a person is expected to move towards this intended position. Since the observation probabilities are estimated using the topological map, states associated with the same topological map node have identical observation probabilities $p(\boldsymbol{O}_t \mid \boldsymbol{S}_t^i)$.

The advantage of storing the set of associated points in the topological mapping algorithm explained in Section 4.1 is shown in Fig. 3, where the simulated data of a T-junction is shown. The blue crosses represent observations, the green crosses the nodes in the topological map and the black dashed lines the edges in this map. In the original work [2], all covariances in b_k, represented by the red ellipses, were constant and set beforehand. This assumption will, in general, only be valid if the topological map has a fine resolution, *i.e.*, if $e_{\max}$ is low and, therefore, the number of nodes in the map is high as is the case in Fig. 3(a). Furthermore it requires tuning a reasonable covariance. With the node specific covariance matrix estimate the resolution can be decreased, as shown in Fig. 3(b). This lowers the computational complexity and does not deteriorate the performance, since the GHMM is used for estimating the intended position, rather than the path towards this position.

4.3. Updating the structure of the GHMM

After updating the topological map with an observation sequence, the structure of the GHMM is updated accordingly. Each new node in the topological map leads to a new state at the position of the node and with the intended position equal to the last position in the observed sequence. Both the new transition probabilities $a_{i,j}$ originating from new edges and the self-transition probability $a_{i,i}$ of new states are initialized with a pre-defined value a_0. The prior probability of being in a new state is set to a pre-defined value π_0. If edges are removed from the topological map, the corresponding transition probabilities in the GHMM are set to zero: $a_{i,j} = 0$. If nodes in the topological map are merged or removed, the corresponding states in the GHMM are removed as well. After updating a node in the topological map, both the corresponding means and covariance matrices in b_k are updated accordingly.

4.4. Updating the parameters of the GHMM

The GHMM parameters are updated using a modified version of the Baum–Welch method [21]. The Baum–Welch method locally maximizes the probability $p(\mathcal{O}_{1:T_k} \mid \lambda)$ of an observed trajectory $\mathcal{O}_{1:T_k}$ given the model in an iterative manner. T_k is defined as the length of the k'th observation sequence $\mathcal{O}_{1:T_k} = \{\boldsymbol{O}_1, \ldots, \boldsymbol{O}_{T_k}\}$. Since the observation probabilities are updated in the previous step already, only the transition probabilities and state priors have to be updated. For a more detailed explanation of the process of updating the parameters the reader is referred to [21,2]. A forward variable can be defined as [21]:

$$\alpha_t(i) = p(\boldsymbol{O}_1, \boldsymbol{O}_2, \ldots, \boldsymbol{O}_t, \boldsymbol{S}_t^i \mid \lambda), \tag{2}$$

which is the probability of observing the trajectory $\mathcal{O}_{1:T}$ up to time t and being in state i at time t given the GHMM. The forward variable is solved for inductively using the standard algorithm explained in [21]. Similarly, a backward variable is defined [21]:

$$\beta_t(i) = p(\boldsymbol{O}_{t+1}, \boldsymbol{O}_{t+2}, \ldots, \boldsymbol{O}_{T_k} \mid \boldsymbol{S}_t^i, \lambda), \tag{3}$$

which is the probability of observing the partial trajectory, $\mathcal{O}_{t+1:T_k}$, given the GHMM model and the state being i at time t. Solving for $\beta_t(i)$ can be done using the standard algorithm explained in [21].

By combining the probabilities represented by the forward and backward variables, the probability of being in state i at time t and in state j at time $t+1$, $\xi_t(i,j)$, can be calculated [21]:

$$\xi_t(i,j) = \frac{\alpha_t(i) a_{i,j} p(\boldsymbol{O}_{t+1} \mid \boldsymbol{S}_{t+1}^j) \beta_{t+1}(j)}{p(\mathcal{O}_{1:T_k} \mid \lambda)}, \tag{4}$$

which combines the forward and backward variables with the probability of going from state i to state j and the probability of the observation at $t+1$ if the state at $t+1$ indeed is j. The denominator acts as a normalization term. The probability of being in state i at time t given both the complete trajectory that was observed and the GHMM can now be calculated using (4):

$$p(\boldsymbol{S}_t^i \mid \mathcal{O}_{1:T_k}, \lambda) = \sum_{j=1}^{N_s} \xi_t(i,j). \tag{5}$$

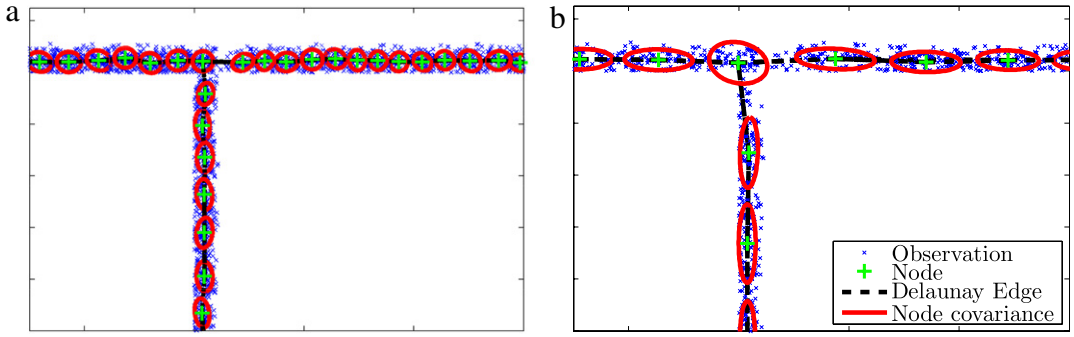

Fig. 3. Topological maps generated using the same simulated data for people approaching a T-junction. In order for a fixed covariance to be a reasonable estimate of the associated points, as in (a), a high topological map resolution is required. In (b), it is shown how an improved covariance estimate allows for a lower resolution of the topological maps. In this example, the map resolution is $e_{\max} = 0.09$ m in (a) and $e_{\max} = 0.25$ m in (b).

With these definitions, the GHMM parameters can be updated in four steps:

1. normalize the state priors π_i and the transition probabilities $a_{i,j}$ since they are not guaranteed to sum up to one after adding and removing edges and states to and from the GHMM;
2. compute the forward variables $\alpha_t(i)$, the backward variables $\beta_t(i)$ and the observation probability $p(\mathcal{O}_{1:T_k} \mid \lambda)$, which equal the sum of all forward variables at time T_k, using the standard algorithms in [21];
3. re-estimate the state priors by the expected number of times of being in state $\boldsymbol{S}^i$ as dictated by the Baum–Welch method:

$$\hat{\pi}_i \leftarrow \frac{\alpha_1(i)\beta_1(i)}{p(\mathcal{O}_{1:T_k} \mid \lambda)}. \tag{6}$$

Tests have shown that using Baum–Welch method leads to too much bias towards recent observations, therefore the update is done as proposed in [2], by a weighted average:

$$\pi_i \leftarrow \frac{(k-1)\pi_i + \hat{\pi}_i}{k}. \tag{7}$$

4. zero transition probabilities will never become non-zero due to parameter updates. Therefore, only non-zero transition probabilities are re-estimated by the expected number of transitions from state $\boldsymbol{S}^i$ to $\boldsymbol{S}^j$ divided by the expected number of transitions from state $\boldsymbol{S}^i$:

$$\hat{a}_{i,j} \leftarrow \frac{\sum_{t=2}^{T_k} \xi_t(i,j)}{\sum_{t=2}^{T_k} p(\boldsymbol{S}_t = i \mid \mathcal{O}_{1:T_k}, \lambda)} \tag{8}$$

and, again, to avoid too much bias [2]:

$$a_{i,j} \leftarrow \frac{(k-1)a_{i,j} + \hat{a}_{i,j}}{k}. \tag{9}$$

Since learning the GHMM parameters requires the full state to be available, an augmented observation sequence:

$$\bar{\mathcal{O}}_{1:T_k} = \left\{ \begin{bmatrix} \boldsymbol{O}_1 \\ \boldsymbol{O}_{T_k} \end{bmatrix}, \begin{bmatrix} \boldsymbol{O}_2 \\ \boldsymbol{O}_{T_k} \end{bmatrix}, \ldots, \begin{bmatrix} \boldsymbol{O}_{T_k} \\ \boldsymbol{O}_{T_k} \end{bmatrix} \right\} \tag{10}$$

serves as input for learning the GHMM. For efficiency reasons, all observation probabilities are pre-calculated each time a new observation sequence arrives. This avoids unnecessary calculation of the same observation likelihood multiple times during the re-estimation of the parameters. The implementation uses the C++ linear algebra library Armadillo [22].

4.5. Estimating the intended position

Once all the GHMM parameters are available, Bayes' rule is used to update the belief state, *i.e.*, the probabilities of being in any of the states given the observations up to time t:

$$p(\boldsymbol{S}_t \mid \mathcal{O}_{1:t}) = \frac{1}{Z} p(\boldsymbol{O}_t \mid \boldsymbol{S}_t) \sum_{\boldsymbol{S}_{t-1}} p(\boldsymbol{S}_t \mid \boldsymbol{S}_{t-1}) p(\boldsymbol{S}_{t-1} \mid \mathcal{O}_{1:t-1}), \tag{11}$$

where $p(\boldsymbol{S}_{t-1} \mid \mathcal{O}_{1:t-1})$ is the prior, *i.e.*, the state estimate at the previous time step or the state prior for the first observation $\boldsymbol{O}_1$, Z is a normalizing constant and the observation probability the Gaussian distribution introduced in Section 4.2. With this belief state, calculating the probability of an intended position $\boldsymbol{x}' = \ell$ is done using:

$$p(\boldsymbol{x}'_t = \ell \mid \mathcal{O}_{1:t}) = \sum_{\boldsymbol{S}_t} p(\boldsymbol{x}'_t = \ell \mid \boldsymbol{S}_t) p(\boldsymbol{S}_t \mid \mathcal{O}_{1:t}), \tag{12}$$

where the probability $p(\boldsymbol{x}'_t = \ell \mid \boldsymbol{S}_t)$ equals one for any state $\boldsymbol{S}_t = \begin{bmatrix} * & \ell \end{bmatrix}^T$ that contains ℓ as intended position and zero otherwise and $p(\boldsymbol{S}_t \mid \mathcal{O}_{1:t})$ is calculated using (11).

Only one intended position is used for the human motion prediction since a person will move towards one location at a time. The most probable intended position, which might change over time, is selected for this purpose. The strategy can therefore be interpreted as a multiple model tracking filter [23] in which exactly one model is considered at each time step. The single model which is considered may change over time depending on the probability mass function over possible intended positions as defined by (12). The way the most probable intended position is used during the prediction will be explained in Section 5.

5. Prediction

The GHMM can be used to predict human trajectories, however, these trajectories can only be represented by discrete state sequences generated by the GHMM. These states represent discrete regions and therefore, only non-smooth trajectories can be predicted. In addition, the GHMM only implicitly models interactions with the environment via the recursive refinement of the model and, therefore, is not able to adequately model short, temporarily interactions, *e.g.*, with dynamic objects. In order to overcome these drawbacks, the intended position estimate provided by the GHMM is combined with a motion model incorporating the concept of social forces. The motion model explained in this section is based on the model used in [3] and introduced in [18,19], where it was used to model pedestrian crowds. The motion model is part of a Kalman filter that will be used to do the final predictions and updates.

Let the state vector of a Kalman filter at time t be the concatenation of a person's estimated position and velocity vectors $\boldsymbol{y}_t$ and $\boldsymbol{v}_t$: $\mathbf{x}_t = \begin{bmatrix}\boldsymbol{y}_t & \boldsymbol{v}_t\end{bmatrix}^T$. The Kalman filter has a motion model:

$$\mathbf{x}_t = \begin{bmatrix}\boldsymbol{y}_{t-1} + \boldsymbol{v}_{t-1}\Delta t + 0.5\boldsymbol{a}\Delta t^2 \\ \boldsymbol{v}_{t-1} + \boldsymbol{a}\Delta t\end{bmatrix} \tag{13}$$

where the acceleration vector $\boldsymbol{a}$, not to be confused with a scalar transition probability $a_{i,j}$ used before, is related to a force using Newton's second law:

$$M_\rho \boldsymbol{a} = \mathbf{F}_\rho + \boldsymbol{\zeta}_\rho. \tag{14}$$

M_ρ is the mass of a person ρ, $\mathbf{F}_\rho$ is the sum of the social forces influencing this person and $\boldsymbol{\zeta}_\rho$ is a noise term representing individual fluctuations not modeled by the deterministic part of the model. The time index is left out to keep the notation less cluttered. The social force $\mathbf{F}_\rho$ equals the sum of three terms and will be calculated using the discrete GHMM state $\boldsymbol{S}_t$ as explained later in this section:

$$\mathbf{F}_\rho = \mathbf{F}_\rho^{pers} + \mathbf{F}_\rho^{soc} + \mathbf{F}_\rho^{phys}. \tag{15}$$

The first term $\mathbf{F}_\rho^{pers}$ models the human's intention, *i.e.*, the desire of a person to adapt his current velocity to his desired speed and direction within a certain relaxation time:

$$\mathbf{F}_\rho^{pers} = M_\rho \frac{\hat{v}_\rho \hat{\mathbf{e}}_e - \boldsymbol{v}_\rho}{\tau_\rho}, \tag{16}$$

where the relaxation time τ_ρ is the time interval needed to go from the current velocity $\boldsymbol{v}_\rho$ to the intended velocity $\hat{v}_\rho$ and the intended direction $\hat{\mathbf{e}}_\rho$, which direction will be determined using the estimated intended position provided by the GHMM using (12). Throughout this work, both the relaxation time and the desired velocity are assumed to be constant and identical for all persons. Future work could be to use better estimates, *e.g.*, the person's average velocity during some time interval. The force $\mathbf{F}_\rho^{pers}$ sums a driving term and a friction term [19].

The social interaction forces represent the repulsive effect of other objects or persons. It, therefore, is a summation over all instances in the world model $\mathcal{W}$, with the exception of the person with index ρ that is currently considered:

$$\mathbf{F}_\rho^{soc} = \sum_{w \in \mathcal{W}\setminus\{\rho\}} q_w \exp\left(\frac{r_{\rho,w} - d_{\rho,w}}{f_w}\right)\mathbf{n}_{\rho,w}. \tag{17}$$

In (17) q_w specifies the magnitude of the social interaction force, f_w specifies the range of this force, $d_{\rho,w}$ is the Euclidean distance between the centers of person ρ and the other world model object or person w, and $r_{\rho,w}$ is the sum of the radii of w and person ρ. Finally $\mathbf{n}_{\rho,w}$ is the normalized vector pointing from w to person ρ. The social interaction force corresponds to the model in [3] with the strength of the anisotropic factor set to one.

The last term in (15) represents hard physical constraints of the environment. Again it is a summation over all world model instances with the exception of person ρ, the person that is considered:

$$\mathbf{F}_\rho^{phys} = \sum_{w \in \mathcal{W}\setminus\{\rho\}} c_w g(r_{\rho,w} - d_{\rho,w})\mathbf{n}_{\rho,w}, \tag{18}$$

where c_w represents the magnitude of the exerted force and the function $g(\chi) = \chi$ if $\chi > 0$ and 0 otherwise. More details on the origin of the model can be found in [19] and the references therein.

6. Experiments

This section presents the results of an extensive set of experiments. A total number of 785 human trajectories are predicted and fed to the GHMM in the various sections. The number of persons in the field of view varied roughly between zero and ten. First, the parameters and some calculation time characteristics are presented in Section 6.1, then Sections 6.2–6.5 present the prediction results.

Table 2
Parameters used during experiments.

Parameter	Value	Unit	Description
a_0	0.1	–	Initial (self-)transition probability
π_o	0.1	–	Initial prior probability
q_w	25	N	Magnitude social force
f_w	50	pixels	Range social force
r_ρ	15	pixels	Radius person
$\hat{v}$	40	pixels/s	Intended velocity person
τ	4	s	Relaxation time
c_w	2.5	N	Magnitude physical force
$N_{p,\max}$	100	points	Maximum number of points per node
$e_{\max}$	30	pixels	Resolution of the topological map

Fig. 4. The full set of 785 trajectories used during the various analyses.

6.1. Algorithmic characteristics

In all of our experiments, the settings summarized in Table 2 are used. Experiments are carried out using a HP EliteBook 8530w with Intel 2.8 GHz duo core processor and the average time needed for a belief state update using (11) and (12) is below 0.5 ms. The total time needed for updating both the GHMM and the topological map is analyzed later. The belief state calculation times enable using the algorithm in an online setting where measurements might arrive at tens of Hertz.

The series of experiments is based on the data set introduced in [24]. It consists of a large set of trajectories of humans walking through the Informatics Forum at the University of Edinburgh. The data were recorded with a camera fixed approximately 23 m above the ground. The main entry/exit points are at the bottom left (front door), top left (cafe), top center (stairs), top right (elevator and night exit) and bottom right (labs). The data used here were recorded on September 10 and July 30. First, ten randomly selected trajectories were used for learning a GHMM, then 775 different trajectories were used for validation and recursive refinement of the model in the various sections below. The complete set of trajectories used is shown in Fig. 4. The topological map after learning all 785 trajectories is shown in Fig. 5.

6.2. Analysis of the GHMM

First, the model size of the intention estimation part is analyzed. Fig. 6 shows how the number of nodes in the topological map increases with the number of trajectories learnt. It in addition shows the number of states and edges in the GHMM. After learning approximately 500 trajectories, the numbers are more or less converged. The number of states increases faster than the number of topological map nodes since different states might be associated with the same node, *e.g.*, movements from A to B and from B to A might share the set of associated topological map nodes but appear on different manifolds and result in different states. In the same

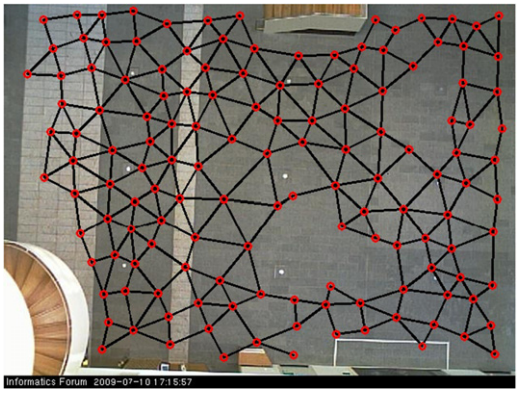

Fig. 5. Topological map, learned from the set of 785 trajectories, projected on the camera image.

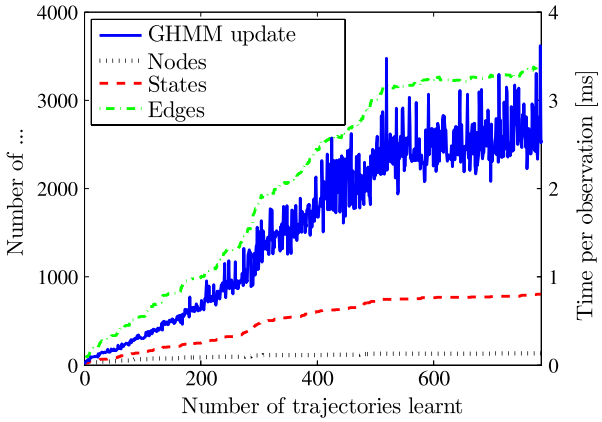

Fig. 6. Number of nodes in the topological map and number of states and edges in the GHMM as a function of the number of trajectories learnt together with the time needed to update the GHMM structure and parameters with one measurement.

figure, the time needed to update the GHMM is indicated by the blue solid line. This time includes updating both the topological map and the GHMM structure and parameters. As expected, the computation time increases with the model size. The time is always below 4 ms and most of the time below 3 ms, which allows for the online updating of the model even on a rather simple laptop.

Fig. 7 shows three of the manifolds present in the GHMM. Each of the figures show the discrete GHMM states associated with one intended position. The green circle represents the person's intended position, the red circles represent the states on the manifold. To keep the visualization uncluttered, both the self-transition and the directed transition probabilities from and to states are left out of the visualization. Instead, the black lines show the connectivity among the GHMM states. This figure shows the difference in complexity of the models on different manifolds. The

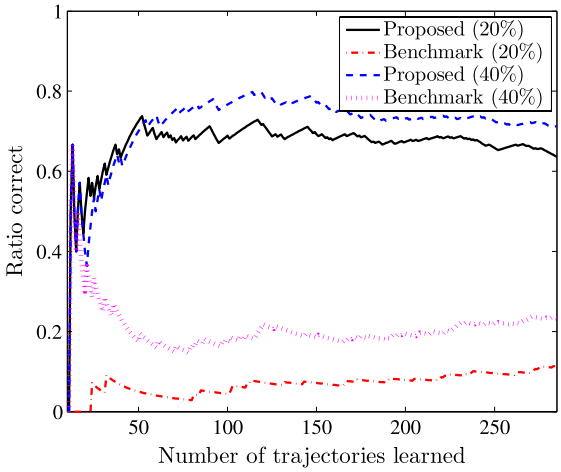

Fig. 8. Ratio of correctly predicted intended positions using both the proposed method and the benchmark strategy after observing 20% or 40% of the trajectory.

learning data set contained many different trajectories towards the front door hence the associated manifold contains many states, as shown in Fig. 7(a). Fig. 7(b) shows the states on the simplest manifold representing persons entering and leaving the scene near the front door. Finally, Fig. 7(c) shows a typical model resulting from a set of trajectories showing limited variability.

6.3. Failure rate

We have used both the motion model proposed in this work and the model as proposed by [3] to estimate the exit point of a person after observing 20% of the trajectory and repeated the analysis with 40% of the trajectory. An intended exit point estimation was classified as being correct if the real exit point differed at most 29 pixels. Changing this value will affect the absolute failure rate, whereas the relative performance of one method compared to the other is, more or less, invariant to this number. This part of the analysis is based on 285 trajectories recorded at September 10 with on average 1.49 persons within the field of view. The analysis focuses on the change in performance after learning a larger number of trajectories. The main results of the comparison are shown in Fig. 8, which shows the ratio of correct predictions as a function of the number of trajectories learnt.

Initially, after learning about ten trajectories, the performance in all cases is similar with the exception of the prediction using the benchmark strategy and 20% of the trajectory. After learning more than 40 trajectories, the proposed method outperforms the benchmark strategy with a factor three to five times if 40% of the trajectory is used and six to seven in case 20% is used. Using 40% instead of 20% improves the absolute success rate for the proposed method with approximately 10%, whereas it doubles the success rate for the benchmark strategy.

Fig. 7. Different manifolds of the GHMM. Each figure shows a manifold associated with a different intended position. The black lines show the connectivity within the GHMM, the green circle the intended position and the red circles the other states on the manifold. Directed edges representing the (self-)transitions within the model are left out to keep the visualization readable. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

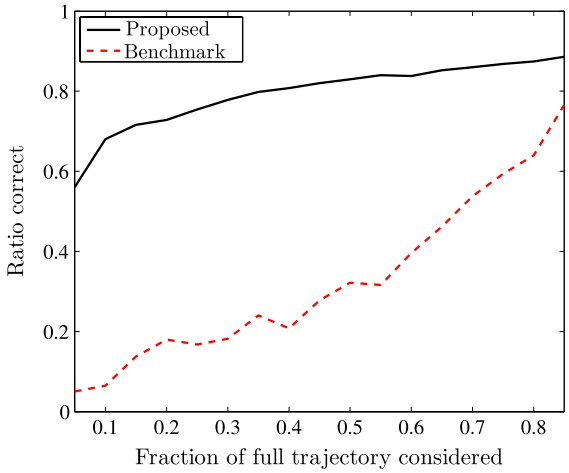

Fig. 9. Ratio of correctly predicted intended positions using both the proposed method and the benchmark strategy after observing the given fraction of the full path for predicting the remainder of the path.

Two conclusions can be drawn from this first analysis. First of all, the proposed method using the estimated intended position provided by the GHMM clearly outperforms the benchmark strategy for the set of 285 person trajectories used during this analysis. Second, the accuracy of the prediction increases after refining the GHMM using more trajectories, something that is enabled by the low computational costs of updating the model. The fact that the average success rate every now and then shows a mild decrease despite an increasing number of trajectories being used for learning is caused by new motion patterns which are observed for the first time. This effect will be addressed in more detail in the next section.

The second part of this analysis is based on the data recorded on a different day: July 30. This different day was selected since (i) the number of persons walking within the field of view is typically larger and (ii) it is considered useful to analyze the performance for the data recorded on different days. The paths of 500 different humans are predicted and the average number of people within the field of view on this day was 4.0. This time the fraction of the full path used to predict the remainder of the path is varied from 5% to 85%. The results for both the proposed and the benchmark strategy are shown in Fig. 9.

Fig. 8 showed how the number of correctly estimated intended positions increases as the number of trajectories learnt increases. Fig. 9 shows how the accuracy increases as a larger fraction of the complete trajectory is used in the query. This is in line with the intuition: the closer a person is to its intended position, the easier this intended position can be predicted. After observing only 5% of the full path, the proposed method is correct in 56% of the 500 paths tested for. As the fraction of the path which is observed increases this percentage increases to almost 90%. The benchmark strategy initially leads to a 5% success rate, which is more than a factor 11 lower than the proposed strategy. As more measurements arrive, this rate increases to approximately 77%. The proposed strategy leads to an absolute increase in the success rate which is between 12% and 62% for the 500 trajectories used during this part of the analysis.

6.4. Predicted path consistency for single persons

This section zooms in on some of the 785 predicted paths that led to the results presented in Section 6.3. Rather than giving a

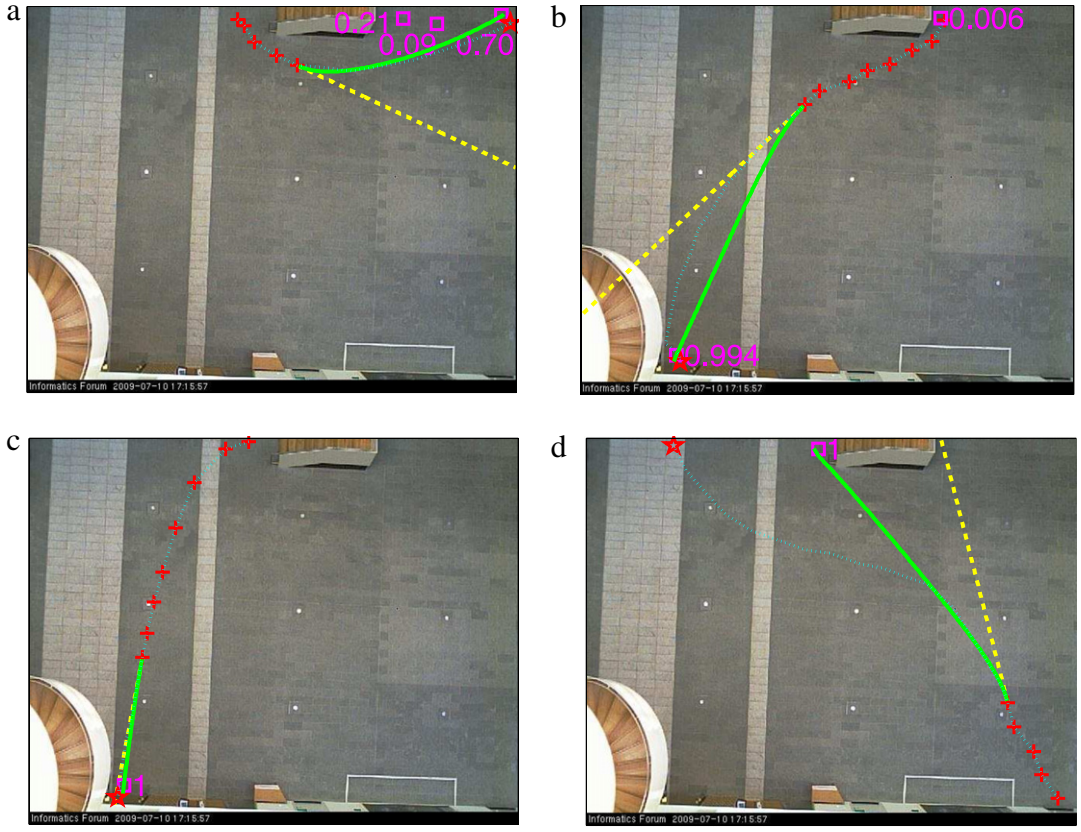

Fig. 10. Predictions for four different observation sequences (204, 219, 236 and 273). Solid green line is with intended position estimation, whereas the dashed yellow line shows the linear extrapolation as in, *e.g.*, [3]. Red crosses represent measurements, the red star the ground truth and the magenta squares the most probable estimated intended positions provided by the GHMM together with the associated probabilities. Cyan represents the ground truth.

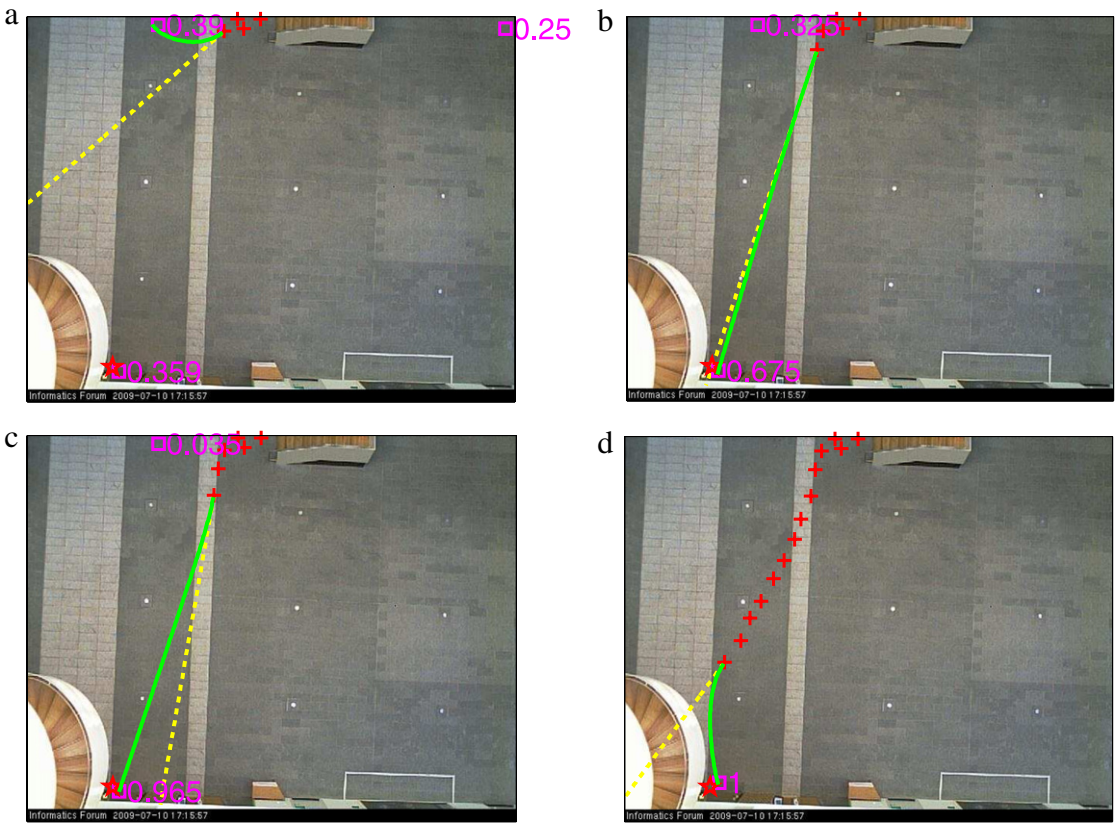

Fig. 11. Prediction for trajectory 201. Initially the intended position estimate is uncertain and the prediction is incorrect. As more measurements arrive, the intended position estimate gains trust and the path is predicted correctly and without much oscillations.

complete overview, representative scenarios are selected. Fig. 10 shows four single person scenarios in more detail. Single person scenarios are selected at this stage since this allows for comparing the proposed strategy with a benchmark strategy while keeping the figures readable.

The red crosses represent the incomplete observation sequence of a single person. The green solid line shows the predicted path using the proposed strategy, whereas the yellow dashed line shows the predicted path using the benchmark strategy as used in, *e.g.*, [3]. The red star represents the real exit point, the dotted cyan line the real path (ground truth) and the magenta squares show the probabilities of the most probable intended positions as estimated by the GHMM together with the associated probabilities. The total number of possible intended positions can be as high as the number of nodes in the topological map shown in Fig. 5. Using only the GHMM for predicting the full path would lead to non-smooth paths on the discrete topological map grid shown in Fig. 5. Such paths are considered unrealistic and, therefore, are not included in the performance evaluation.

In order for a robot to be able to correctly anticipate on a person's path two things are important: (i) the predicted path must lead to the correct exit point and (ii) the path towards this exit point must be consistent over time, *i.e.*, it should not change unrealistically each time a new measurement arrives. We start with analyzing the first of these points by further investigating some of the successes and failures that led to the results in Figs. 8 and 9. After that, the second of these points is analyzed.

Fig. 10(a) and (b) shows typical scenarios in which the linear extrapolation fails, whereas the GHMM is able to correctly estimate the intended position. The social forces motion model transforms the discrete intended position estimate into a smooth path that takes obstacles such as the stairs in Fig. 10(a) or other persons (if present) into account. The mismatch between the real path and the predicted path towards the right location in Fig. 10(b) is mainly caused by the parameters in the social forces model; not much effort was put into finding optimal parameters. Future work could include learning the parameters from the data.

The prediction shown in Fig. 10(c) is correct using both the proposed method and the method proposed in [3]. Fig. 10(d) shows a typical scenario in which both the proposed and the benchmark method fail. The requested trajectory from the lower right to the light gray tiles on the upper left corner is not included in the set of learnt trajectories by the time this prediction is performed. As a result, the GHMM provides an incorrect estimation of the intended position with a high probability. Although a high probability for an incorrect intended position clearly is not desired, it is inevitable in the case of learning from data that is incomplete. The bright side of using GHMMs is that the recursive refinement of the model is facilitated, hence after observing the full trajectory, the observation sequence can be added to the model thereby improving the accuracy of future predictions. Section 6.3 already showed that the benchmark strategy leads to a relatively large number of failures compared to the proposed strategy.

The second part of the analysis in this section investigates the consistency of predicted paths over time. Fig. 11 shows the predicted path for trajectory 201 after receiving four, five, six and fourteen measurements of the same path. Initially, the GHMM has three candidate intended positions that all have similar probabilities. The proposed prediction method favors the wrong one and as a result, the estimated path is incorrect. After receiving one more measurement, the belief state of the GHMM is updated, as shown in Fig. 11(b), and as a result the most probable intended position matches the actual intention of the person. After six measurements, the ambiguity is largely resolved and again the predicted path leads to the correct goal. From then on, the path stays more or less the same, as can be seen in Fig. 11(d). In the same sequence

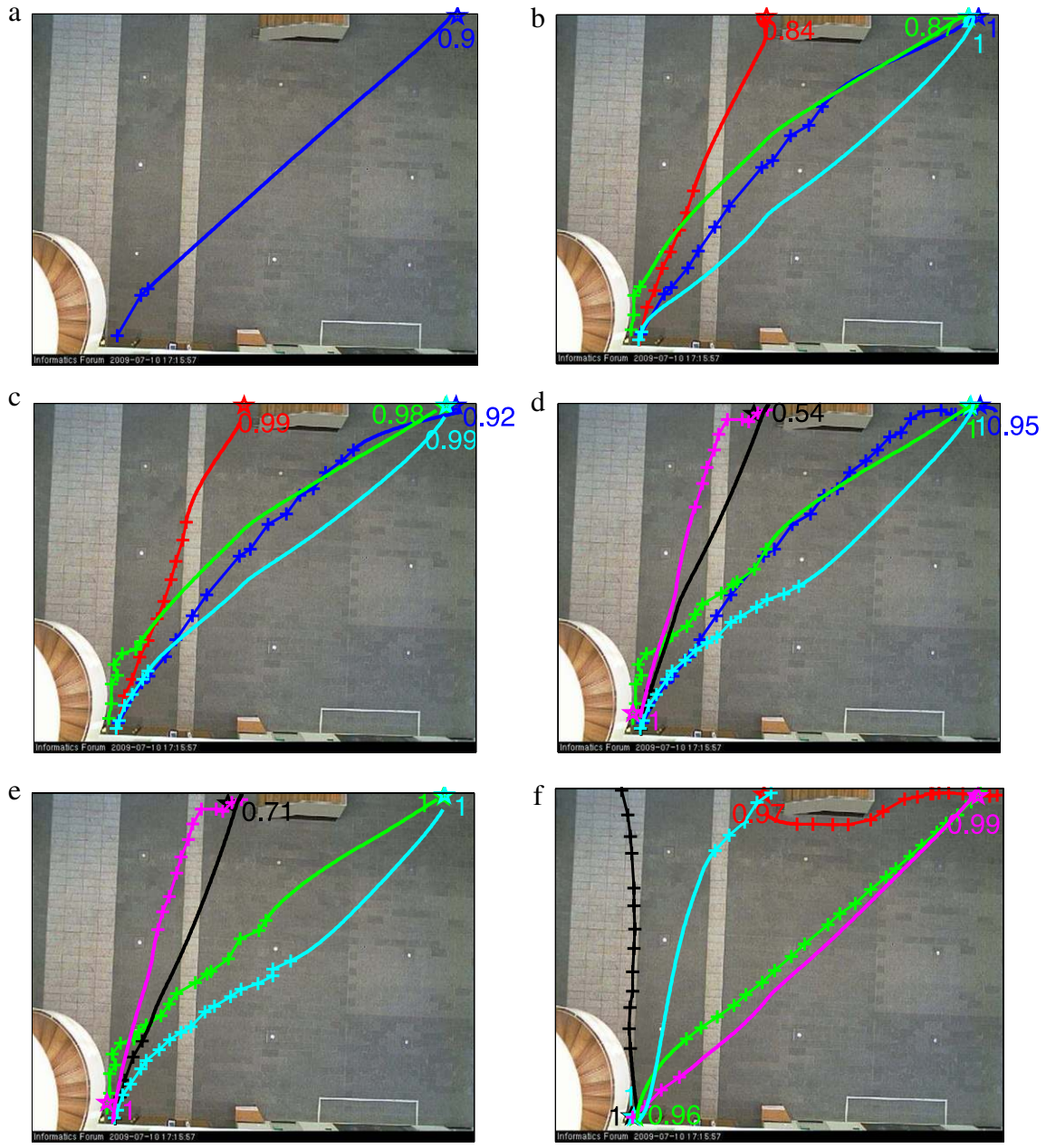

Fig. 12. Predictions in multiple persons scenarios. Each color represents a different person. The crosses represent measurements, the solid lines predicted paths, the numbers the probabilities associated with the estimated intended state and the stars the actual intended positions (ground truth).

of figures, the dashed yellow lines show the predicted path using the benchmark strategy. The collection of predicted paths shows a much larger variation. If a robot, *e.g.*, during a navigation task, wants to make sure humans do not have to deviate from their path it is forced to replan its own path each time the predicted path changes hence these variations are highly undesirable.

6.5. Multiple person scenarios

This section presents representative results for multiple person scenarios. In order to keep the figures readable, the benchmark strategy is left out of this part of the analysis. Fig. 12 shows some typical examples showing a subset of the 785 predicted trajectories. Each color represents a person, the crosses represent measurements, the stars the ground truth, the solid lines predicted paths and the numbers the probabilities associated with the most probable estimation of the intended state.

Initially, there is one person in the field of view, as shown in Fig. 12(a). The estimated intended position is correct and the path is rather straight forward. A few time steps later, three other persons have entered the field of view, as shown in Fig. 12(b). By this time, the first person (blue line) is getting closer to its intended position. The red line represents the predicted path of the second person. Again the intended position is correct and the path is rather straight forward. The third (green line) and fourth (cyan line) person entered short after each other and as a result, repelling forces are taken into account during the prediction. The distance of these persons to the second person (red line) is sufficient to prevent significant interaction forces. Fig. 12(c) shows the scene a little later. Now the intended positions are all known with a high probability and the paths are similar to the ones predicted

before. Again, persons three and four keep some distance before converging to the same intended position. In Fig. 12(d) various things can be noticed. First of all, the second person (red line) left the scene. Besides this, a fifth (magenta line) and sixth (black line) person have entered. The predicted paths of these persons almost overlap, however, the associated times are different. If these times would have differed less, the social forces would have enforced a larger spatial difference. The third and fourth persons are still on their way while keeping some relative distance and measurements of the first person keep appearing very close to its estimated goal position. Only moments later, Fig. 12(e) shows how the first person left the scene, leaving persons three, four, five and six behind. The probability of the estimated goal position for person six (black line) increased from 0.54 to 0.71 and the predicted path first moves away from the path of person five (magenta) and then goes in a straight line to the expected goal position. This sequence of figures shows how the proposed strategy works in a scenario with many humans sharing the same environment. It in addition shows the role of the estimated and intended velocities. These velocities determine when a person arrives where and this directly affects the social interaction forces with other persons. Throughout this work, the intended velocity is assumed constant for all persons. Based on the analyses performed throughout this work, this assumption appears reasonable, however, it will not be necessarily true in general. Better estimates could be used instead, as mentioned in Section 5.

Fig. 12(f) shows a last typical scenario with five different persons. The person associated with the red line moves around the stairs towards its intended position. The green, cyan and black lines represent persons moving towards the same position but arriving at different times. Finally, the magenta line represents a person moving to the upper right corner while keeping some distance from the other persons.

These multiple person scenarios only show a few of the 785 predicted paths, however, together with the scenarios presented earlier these are considered to be representative of the full set.

7. Conclusions

This work has shown how a two step approach can be used for human motion prediction. By learning a growing hidden Markov model (GHMM) from data, the intended position of a person can be estimated from a discrete number of possible goals in a known environment. The algorithm used for learning the GHMM is tailored for the goal estimation and only requires a few milliseconds per observation. In order to be able to do predictions without requiring a discretization of the world and while taken temporary object interactions into account, the GHMM is combined with a motion model that is based on the concept of social forces. Compared to the standard social forces motion model with linear extrapolation, the prediction accuracy increased significantly as was shown while predicting the paths of 785 humans. In addition, the oscillations that appear in the predicted paths using the benchmark strategy are drastically reduced when using the proposed strategy.

Future work will focus on incorporating semantics into the prediction framework. Rather than defining intended positions as coordinates in Cartesian space, intended positions can be defined relative to objects, *e.g.*, person X is moving towards the table. This will allow for re-using the learnt motion patterns in environments that differ from the learnt scene. Furthermore it increases robustness to movements of the objects associated with the goal positions. In addition, the possibility to learn the parameters needed in the social forces based motion model from the data used to train the GHMM can be investigated and the proposed motion models can be incorporated in a robot path planning module. For the integration in a robot navigation module, [25] seems a useful starting point. Finally, relaxing the assumption of a person independent constant intended velocity could further improve the accuracy of the predictions.

Acknowledgments

The research leading to these results has received funding from the European Union Seventh Framework Program FP7/2007–2013 under grant agreement no. 248942 RoboEarth. The authors would like to thank the anonymous reviewers for their valuable feedback.

References

[1] M. Bennewitz, W. Burgard, G. Cielniak, S. Thrun, Learning motion patterns of people for compliant robot motion, Int. J. Robot. Res. 24 (1) (2005) 31–48.
[2] D. Vasquez, T. Fraichard, C. Laugier, Incremental learning of statistical motion patterns with growing hidden Markov models, IEEE Trans. Intell. Transp. Syst. 10 (3) (2009) 403–416. http://dx.doi.org/10.1109/TITS.2009.2020208.
[3] M. Luber, J. Stork, G. Tipaldi, K. Arras, People tracking with human motion predictions from social forces, in: 2010 IEEE International Conference on Robotics and Automation, ICRA, 2010, pp. 464–469. http://dx.doi.org/10.1109/ROBOT.2010.5509779.
[4] S. O'Callaghan, S. Singh, A. Alempijevic, F. Ramos, Learning navigational maps by observing human motion patterns, in: 2011 IEEE International Conference on Robotics and Automation, ICRA, 2011, pp. 4333–4340. http://dx.doi.org/10.1109/ICRA.2011.5980478.
[5] S.-Y. Chung, H.-P. Huang, Learning the motion patterns of humans for predictive navigation, in: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2009. AIM 2009, 2009, pp. 752–757. http://dx.doi.org/10.1109/AIM.2009.5229922.
[6] K.-S. Tseng, A.-W. Tang, Goal-oriented and map-based people tracking using virtual force field, in: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2010, pp. 3410–3415. http://dx.doi.org/10.1109/IROS.2010.5650203.
[7] K. Yamaguchi, A. Berg, L. Ortiz, T. Berg, Who are you with and where are you going? in: 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2011, pp. 1345–1352. http://dx.doi.org/10.1109/CVPR.2011.5995468.
[8] F. Rohrmüller, M. Althoff, D. Wollherr, M. Buss, Probabilistic mapping of dynamic obstacles using Markov chains for replanning in dynamic environments, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008, IROS 2008, 2008, pp. 2504–2510. http://dx.doi.org/10.1109/IROS.2008.4650952.
[9] T. Kanda, D. Glas, M. Shiomi, N. Hagita, Abstracting people's trajectories for social robots to proactively approach customers, IEEE Trans. Robot. 25 (6) (2009) 1382–1396. http://dx.doi.org/10.1109/TRO.2009.2032969.
[10] A.F. Foka, P.E. Trahanias, Probabilistic autonomous robot navigation in dynamic environments with human motion prediction, Int. J. Social Robot. 2 (2010) 79–94. http://dx.doi.org/10.1007/s12369-009-0037-z.
[11] K.M. Kitani, B.D. Ziebart, J.A. Bagnell, M. Hebert, Activity forecasting, in: Computer Vision—ECCV 2012, Springer, 2012, pp. 201–214.
[12] Z. Chen, L. Wang, N.H. Yung, Adaptive human motion analysis and prediction, Pattern Recognit. 44 (12) (2011) 2902–2914 http://dx.doi.org/10.1016/j.patcog.2011.04.022.
[13] S. Pellegrini, A. Ess, L.J.V. Gool, Predicting pedestrian trajectories, in: Visual Analysis of Humans, Springer, 2011, pp. 473–491.
[14] S. Pellegrini, A. Ess, K. Schindler, L. van Gool, You'll never walk alone: modeling social behavior for multi-target tracking, in: International Conference on Computer Vision, 2009.
[15] S.-Y. Chung, H.-P. Huang, A mobile robot that understands pedestrian spatial behaviors, in: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2010, pp. 5861–5866. http://dx.doi.org/10.1109/IROS.2010.5649718.
[16] M. Luber, L. Spinello, J. Silva, K. Arras, Socially-aware robot navigation: a learning approach, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2012, pp. 902–907. http://dx.doi.org/10.1109/IROS.2012.6385716.
[17] B. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. Bagnell, M. Hebert, A. Dey, S. Srinivasa, Planning-based prediction for pedestrians, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009, 2009, pp. 3931–3936. http://dx.doi.org/10.1109/IROS.2009.5354147.
[18] D. Helbing, I. Farkas, T. Vicsek, Simulating dynamical features of escape panic, Nature 407 (6803) (2000) 487–490.
[19] D. Helbing, I.J. Farkas, P. Molnar, T. Vicsek, Simulation of pedestrian crowds in normal and evacuation situations, in: Pedestrian and Evacuation Dynamics, Vol. 21, 2002, pp. 21–58.
[20] J. Jockusch, H. Ritter, An instantaneous topological mapping model for correlated stimuli, in: International Joint Conference on Neural Networks, 1999. IJCNN'99. Vol. 1, 1999, pp. 529–534. http://dx.doi.org/10.1109/IJCNN.1999.831553.
[21] L. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE 77 (2) (1989) 257–286 http://dx.doi.org/10.1109/5.18626.

[22] C. Sanderson, Armadillo: an Open Source C++ Linear Algebra Library for Fast Prototyping and Computationally Intensive Experiments, Tech. Rep., NICTA, Australia, 2010.
[23] X.R. Li, V.P. Jilkov, A survey of maneuvering target tracking. Part V: multiple-model methods, IEEE Trans. Aerosp. Electron. Syst. 41 (4) (2005) 1255–1274.
[24] B. Majecka, Statistical models of pedestrians behavior in the forum, Master's Thesis, School of Informatics, University of Edinburgh, 2009. URL: http://homepages.inf.ed.ac.uk/rbf/FORUMTRACKING/.
[25] T. Kruse, A.K. Pandey, R. Alami, A. Kirsch, Human-aware robot navigation: a survey, Robot. Auton. Syst. 61 (12) (2013) 1726–1743 http://dx.doi.org/10.1016/j.robot.2013.05.007.

Jos Elfring (1985) received the M.Sc. from Eindhoven University of technology (TU/e) in 2009. Since December, 2009, he is a Ph.D. candidate in the Control Systems Technology group of the TU/e. He is involved in the FP7 RoboEarth project which focuses on sharing knowledge among robots. Besides this, he is a member of the Tech United RoboCup@Home team and he successfully finished the Dutch Institute of Systems and Control (DISC) course program. His research interests include creating, maintaining and sharing semantic world descriptions in robotics.

René van de Molengraft (1963) received the M.Sc. (cum laude) and Ph.D. degrees in Mechanical Engineering from TU/e in 1986 and 1990. In 1991 he fulfilled his military service. Since 1992 he is a staff member of the CST group. Since 2005, he is the project leader of the Tech United RoboCup team, which placed second in the 2008–2011 RoboCup Middle Size League world championships and first in 2012. Since 2008, he is an associate editor of IFAC Mechatronics. He is the co-author of more than 40 papers in refereed scientific journals and more than 85 papers in refereed scientific proceedings.

Maarten Steinbuch (1960) is a professor in Systems and Control and the head of the Control Systems Technology group at Eindhoven University of Technology, The Netherlands. He received the M.Sc. and Ph.D. degrees in 1984 and 1989 resp. From 1987 to 1999 he was with Philips Electronics. He is the Editor-in-Chief of IFAC Mechatronics. He is the Scientific Director of the Centre of Competence High Tech Systems of the Federation of Dutch Technical Universities, and he is the director of the TU/e Graduate Program Automotive Systems. His research interests are in the field of mechatronics, robotics, automotive power trains and control of fusion plasmas.