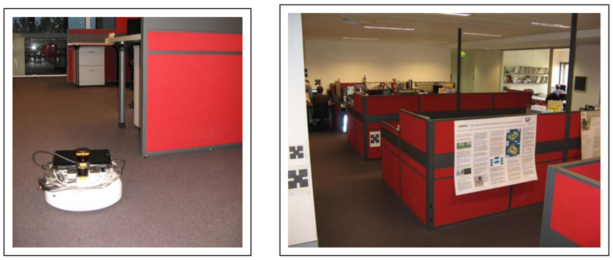
© 2010 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Using Common Motion Patterns to Improve a Robot's Operation in Populated Environments

Stephan Sehestedt, Sarath Kodagoda and Gamini Dissanayake
ARC Centre of Excellence for Autonomous Systems (CAS)
The University of Technology, Sydney
Australia
Email: {S.Sehestedt, S.Kodagoda, G.Dissanayake}@cas.edu.au

***Abstract*—Robotic devices are increasingly penetrating the human work spaces as stand alone units and helpers. It is believed that a robot could be easily integrated with humans, if the robot can learn how to behave in a socially acceptable manner. This involves a robot to observe, learn and comply with basic rules of human behaviors. As an example, one would expect a robot to travel in an environment without intruding human workspaces unnecessarily. Thus, identifying common motion patterns of people by observing a specific environment is an important task as people's trajectories are usually not random, however are tailored to the way the environment is structured. We propose a learning algorithm to construct a Sampled Hidden Markov Model (SHMM) that captures behavior of people through observations and then demonstrate how this model could be exploited for planning socially aware paths. Experimental results are presented to demonstrate the viability of the proposed approach.**

***Index Terms*—Human Robot Interaction, Socially Aware Planning, Motion Patterns, Sampled Hidden Markov Models**



(a) (b)

Fig. 1. a) The IRobot Create in its environment. b) The office space.

I. INTRODUCTION

There is significant interest in research trying to find solution to problems associated with the world's aging population and the rising cost of labor especially for simple tasks. One line of thinking is to provide robotic helpers to support the elderly in day to day activities or for simple tasks in work places. This is nontrivial and there is a great number of challenges to be solved before any deployment. These challenges can range from reliability, safe operation and effective Human Robot Interaction (HRI). It is believed that the human robot interaction becomes efficient and convenient, if the robot understands human behaviors and acts accordingly. Therefore, in this paper our focus is on socially acceptable robot navigation based on learned common motion patterns in a populated environment.

Human motion in an environment is usually not random but is influenced by a complex combination of physiological and social constraints. Therefore, a path a person takes is not necessarily the shortest path, as opposed to the usual criteria used in robot path planning. From a physiological point of view a short and smooth path is preferred, where smoothness means no sharp turns or other difficult motions, such as side stepping. This was noted in literature [1] where human motion was found to be similar to non-holonomic locomotion. Social constraints stem from the notion of personal space [2], which states that the space needed by a person is more than the volume of the body. It was shown that unwanted intrusion on the personal space causes discomfort and in some cases it is perceived as invading the privacy [3]. Therefore, if a robotic co-worker is to be seamlessly integrated into the society, it needs to be aware and consider such aspects while carrying out intended tasks. One way of realizing such a robot is to provide it with the capability of learning social rules by observing common tasks carried out by humans.

The learning of motion patterns is of great interest in video surveillance applications (i.g. [4] and [5]) and was also identified as an area of interest for mobile robotics, e.g. in [6] and [7]. However, the published approaches share the important limitation that the models cannot be learned on a mobile platform. This stems from the fact that the observer has to be stationary and be able to observe the complete area of interest at all times. Some of the presented work also uses off-line learning with training data sets. However, the full potential of such a model of motion patterns can only be exploited when learning happens on-line on a mobile robot. Therefore, we previously published a new method for learning common human motion patterns based on Sampled Hidden Markov Models (SHMM) [8].

Our approach does not assume full observability of the environment, because it can deal with partial observations. Therefore, a robot can learn while in operation rather than waiting until a complete observation of the environment is available. As shown in Fig. 1 the environment in which our robot operates consists of tall cubical walls, which restrict the field of view of the robot. In such scenarios learning based on partial observations, building partial maps and combining such information is crucial. Furthermore, it will be shown, that this model yields some interesting properties and can be used to improve a mobile robot's navigation capabilities.

The remainder of this publication is organized as follows. Section II outlines our approach to learning SHMMs on a mobile robot. Section III presents the psychological and technical background to our proposed socially aware path planning method. In Section IV, we present experimental results. Finally, in Section V, we discuss the findings and future work.

II. LEARNING MOTION PATTERNS

A. *Hidden Markov Models*

A Hidden Markov Model is a statistical model that represents a system as a directed graph. Here we briefly outline HMMs following the notation used by Rabiner [9]. HMMs are defined by N states of a system $S = s^1, s^2, ..., s^N$, observation symbols $V = v^1, v^2, ..., v^K$ with K being the number of symbols and state transition probability distribution $A = a^{ij}$, which is given as

$$a^{(ij)} = P(q_{t+1} = s^{(j)} | q_t = s^{(i)}), 1 \leq i \leq N \qquad 1 \leq j \leq N \tag{1}$$

Furthermore, the observation probabilities in state j, $B = b^{ij}$ are formulated as

$$b^{(ij)} = P(v^{(i)} | s^{(j)}), 1 \leq i \leq K \qquad 1 \leq j \leq N \tag{2}$$

Finally, the initial state distribution $\pi = \pi^i$ is defined as

$$\pi^{(i)} = P(q_1 = s^{(i)}), 1 \leq i \leq N \tag{3}$$

Most HMM frameworks highly depend on prior topological knowledge of the model and learning is performed through previously obtained data sets. There is no easy way to update these models over time [9]. Thus, these implementations are not suitable for the application at hand.

1) Learning a Sampled Hidden Markov Model: Consider a person walking from the bottom to the top in Fig. 2(a). The person is tracked and the probability distribution is represented as weighted samples of the series of poses (see Fig. 2(a)). This leads to a series of sample clusters C, which is given by,

$$C = \left[\begin{array}{cccc} c^{(0)} & c^{(1)} & ... & c^{(M)} \end{array} \right] \tag{4}$$

To extract an HMM each of those clusters in C can be interpreted as a state of an HMM as

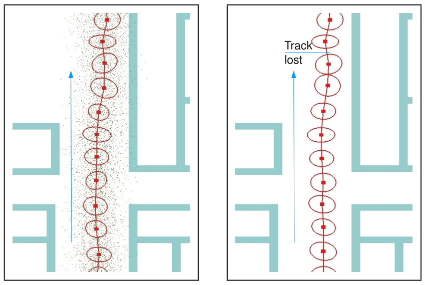
$$S = s^{(i)} = \left[\begin{array}{c} \mu^{(i)} \\ \Sigma^{(i)} \end{array} \right] \; 1 \leq i \leq N \tag{5}$$

where $\mu^{(i)}$ and $\sigma^{(i)}$ are mean and covariance of the $i-th$ state and N is the number of states. Assuming zero states at the beginning, $N = M$ after adding C to the initially empty model. $\mu^{(i)}$ and $\sigma^{(i)}$ are computed from the underlying sample set and thus represent a 4-dimensional distribution over $x-y-\theta-v$. In Fig. 2(a) a 2D projection of SHMM states can be seen as the red covariance ellipses in x and y. This figure also shows the learned model based on a single observed trajectory and the underlying samples.

The transition from state i to state j is given by the sequence of sample clusters and thus the transition matrix A consists of

$$a^{(ij)} = \left[\begin{array}{c} N^{(ij)} \\ P(s^{(j)} | s^{(i)}) \end{array} \right] \begin{array}{l} 1 \leq i \leq N \\ \\ 1 \leq j \leq N \end{array} \tag{6}$$

where $N^{(ij)}$ is the number of times the transition was observed and $P(s^{(j)} | s^{(i)})$ is the probability of the transition. Naturally, the probabilities of the newly learnt transitions in this example are 1.



(a) (b)

Fig. 2. Motion pattern learning with people walking from the bottom to the top of the image along the blue arrow a) 2D projection of the SHMM, b) Updated model with an observation of a second person.

2) Updating The Model: When observing another trajectory, a sequence of sample clusters is produced and a data association step will be carried out. This will associate the observations to the learned model, while initializing new nodes for previously unobserved parts. To do this data association, the symmetrized Kullback-Leibler distance (KLD) [10] is used. It calculates the distance of clusters in C to states in the model. The symmetrized KL-distance is defined as follows

$$\begin{aligned} KLD_{sym}(s^{(i)} | c^{(j)}) &= KLD(s^{(i)} | c^{(j)}) \\ &+ KLD(c^{(j)} | s^{(i)}), 1 \leq i \leq N \\ & \qquad 1 \leq j \leq M \end{aligned} \tag{7}$$

where $KLD_{sym}(s^{(i)} | c^{(j)})$ denotes the symmetrized KL-distance of state $s^{(i)}$ to cluster $c^{(j)}$ taking into account all N states and all M clusters of a trajectory. If an association is found between the i-th state and the j-th cluster, the cluster's samples belonging to j will be added to the state. The KL-distance is commonly used in literature for this purpose, nevertheless it is to be noted that other distance measures such as the Mahalanobis distance could be used as well. To keep the number of samples used to model a state constant and to discard low weighted samples, a resampling procedure is employed. This is done similar to a normal particle filter with systematic resampling [11]. Finally, the transition probabilities are updated as

$$P(s^{(j)} | s^{(i)}) = \frac{N^{(ij)}}{\sum_{j=0}^{N} N^{(ij)}} \tag{8}$$

If a cluster could not be associated to an already existing state of the SHMM, it is added as a new state and the state transition matrix A gets extended accordingly.

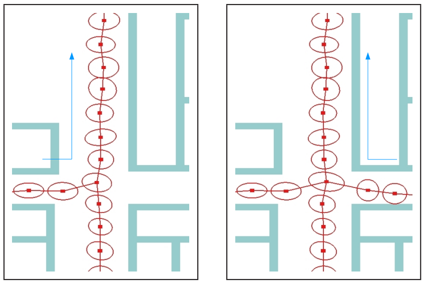

(a) (b)

Fig. 3. Dynamic motion learning, a) A person is tracked coming from the left and then walking along the previously learned path. b) Same situation with a person coming from the right. The robot (not visible) changed its location during observations. (real data)

To update the state transitions the knowledge about the sequence of newly added and associated states can be exploited. Where a transition is already known the values can be updated by incrementing $N^{(ij)}$ and updating the transition probabilities accordingly.

Fig. 2(b) shows the updated model after a second person was observed moving along a similar trajectory as the first person (as indicated by the blue arrow). Towards the top of the figure (as marked) the robot lost track of the person and it can be seen (when carefully comparing Fig. 2(b) with Fig. 2(a)) that only the states below that point were updated. Since the two trajectories were very similar no new states and transitions had to be added.

In contrast Fig. 3(a) shows the SHMM after another person was observed coming from the left, again following the trajectory indicated by the blue arrow. It can be seen that new states were added coming from the left and that a transition from the new part of the model to the former model was learned. The situation is similar in Fig. 3(b) where a person was tracked coming from the right and the model was updated accordingly. Further considerations such as computational complexity and more details about the sampling scheme can be found in [8].

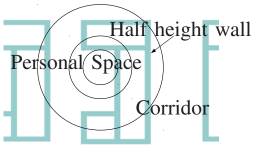
III. SOCIALLY AWARE PATH PLANNING

As noted in the introduction, human path planning is influenced by a number of constraints due to physiological and social requirements. Here we will briefly discuss the situation for an office environment and the significance to robot path planning. In psychology literature personal space is defined as a set of areas around a person which denote the expected proximity for certain kinds of interaction (see Table I) and is a widely accepted concept [2]. It has been shown that unexpected or unwanted intrusion causes discomfort and privacy concerns [3].

In past work it has been noted that if the knowledge of human motion patterns is given, a person's motion can be predicted with high confidence. Therefore, such information can be used to improve a robot's reactive behavior in populated environments [6] [7] [12]. Intuitively, the same holds true for human-human interaction. Based on past knowledge humans can estimate where another person is likely to go and reactive behavior is performed accordingly.

TABLE I
PERSONAL SPACE ZONES AS DEFINED IN [2].

Personal Space Zone	Distance (m)
Intimate	$0.0 - 0.45$
Personal	$0.45 - 1.2$
Social - Consultive	$1.2 - 3.0$
Public	$3.0+$



From the above we derive requirements for robot path planning in an office as follows. A robotic co-worker should not unnecessarily intrude the personal space of a person in his desk area. Seeing the map of an open office space as in Fig. 4 a robot needs to avoid planning paths through someone's personal space where he is sitting as given in Table I. Such paths lead to similar motion patterns as its human peers, which could allow humans to understand and interact with the robots effectively and efficiently.

A. Models of Motion Patterns in Path Planning

A robot who can learn and behave as a human can easily and efficiently integrate with the environment. Following this philosophy, we have included the previously learned model of motion patterns into the path planning algorithm to make the robot's operation more socially acceptable. The information that can be derived from the learned motion patterns relevant for path planning are the motion patterns (trajectories) themselves and traffic density.

B. The A^* Algorithm

The A^* algorithm and its derivatives are a popular solution for the path planning problem [13] [14] [15]. It performs the best-first search on a grid which is precomputed using a collision detector with defined configuration values. More precisely a configuration space C is computed which contains all static obstacles in the d-dimensional space of the robot. C_{free} exists, which contains all collision free configurations. A path planning algorithm searches a path such that the path lies in C_{free}.

A^* is defined by the functions $g(x)$ which is the shortest path from start to goal by Euclidean distance (often called the path-cost function). Furthermore, $h(x)$ is used as a heuristic estimate of the length of the path and $f(x)$ is the sum of $g(x)$ and $h(x)$. The algorithm searches for a path using a priority queue, where the priority of node x is higher the lower its $f(x)$ is. Hence, it is called a best-first search.

This can be exploited to include prior information about dynamics in the environment. A cost function is used to evaluate the cost of a path with respect to a model of motion patterns

$$g_D(x) \sim D(x) \tag{9}$$

Where D denotes the learned model of motion patterns. $g_D(x)$ returns a low value if node x is in an area of high traffic density and a high value if it is in an area of low traffic density. Consequently, instead of $g(x)$ int the standard A^*-algorithm the function

$$G(x) = g(x) + g_D(x) \tag{10}$$

which can be used to calculate $f(x)$. This cost function applies whenever the robot is supposed to prefer a commonly taken path. However, this may not always be a good solution. E.g. consider a robot with limited capabilities which should probably avoid busy areas or a robot whose has the task to empty all trash bins. Clearly Eq. 9 should be reformulated to account for the requirements of a task the robot might have. Hence, a factor w is introduced

$$G(x) = g(x) + w * g_D(x), 0 \leq w \leq 1 \tag{11}$$

Where the factor w is chosen depending on the current requirements. If it is set to zero, the model will be ignored and paths are planned using normal A^*. If the robot prefers common paths w is set to 1 and any number in the range of 0 to 1 denotes whether it prefers the pure A^* distance or the combination with $g_D(x)$. Note that for some applications a robot may be required to avoid busy areas. For this we can change G(x) to be

$$\begin{aligned} G(x) = g(x) + w_g * g_D(x) + w_{\overline{g}} * \overline{g}_D(x), 0 \leq w_g \leq 1 \\ 0 \leq w_{\overline{g}} \leq 1 \end{aligned} \tag{12}$$

Where $\overline{g}_D(x)$ returns a low cost value for paths within low traffic areas. The factors w_g and $w_{\overline{g}}$ determine whether to prefer high or low traffic density areas or to ignore this information. Naturally, only one of the two values should be non-zero. Other aspects and the integration of the method into Probabilistic Roadmap based path planning can be found in [16].

IV. Experimental Results

All experiments are performed with our IRobot Create platform which carries a Hokuyo UTM-30LX laser range finder for localization and tracking and a small size notebook (eeePC 1005H).

A. The Robot's Environment

The operating environment in is an open office space of approximately $20x25m$, which we consider a complex space for experiments due to clutter and symmetries. We do not remove items like chairs or trash bins before conducting experiments and no beacons are installed in the environment to assist navigation.

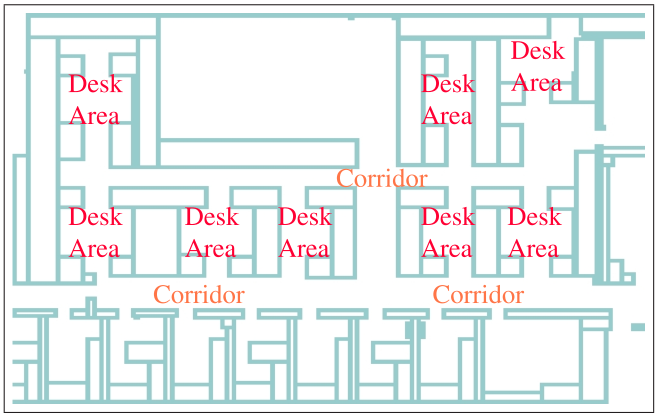


Fig. 4. The map of the open office space.

Fig. 4 shows a map of the office where desk areas and corridor areas are marked accordingly. The figure in Table I shows the personal space overlaid on a desk space with the center being approximately located at a chair. This shows that if the robot moves through the desk area, it would have to cross a sitting person's personal space and may even intrude the personal zone. People commonly avoid crossing these areas if the detour is not too cumbersome.

B. Model Learning and Representation

In the following we present experiments to illustrate the ability of the learning approach to adapt in a dynamic human populated environment. Firstly, Fig. 5 shows a typical human trajectory modeled as an SHMM. It can be seen, as noted in literature [1], that human trajectories are usually not straight lines but curves. Also when walking around a corner, a smooth curve rather than a 90 degree turn on the spot can be observed.

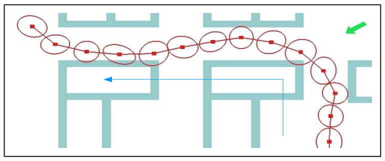


Fig. 5. A typical curved human trajectory modeled as an SHMM. The thin blue arrow denotes the direction of motion and the thick green arrow the robot's position.

The second experiment illustrates the adaptability of the proposed model of motion patterns. In Fig. 6 the robot observed people walking from the bottom to the top of the image. Fig. 6(b) shows the result after 3 trajectories were perceived. Then an obstacle was put close to the learnt path, so that people would have to alter their trajectories slightly. Fig. 6(b) shows how the model slowly adapts to the change until it converges after a while (Fig. 6(c)).

The third experiment shows a similar situation, however this time with a larger obstacle blocking the normal path, forcing people to substantially alter their trajectories. Due to the notable difference in the observations, it can be seen that the new trajectory is added to the model as a branch in Fig. 7. Above this, it can be seen that initially the transition from A to C has a lower probability than the transition from A to B, as indicated by the thickness of the lines in Fig. 7(b). With more observations, the transition from A to C becomes more

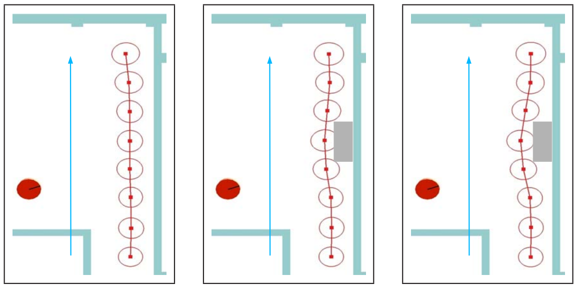
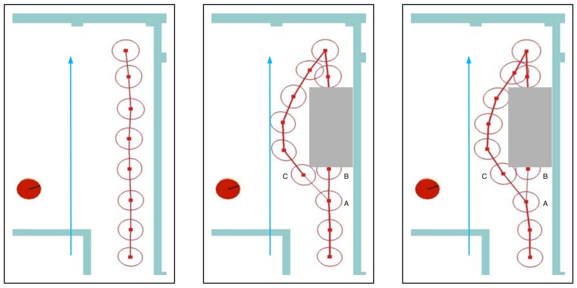

(a) (b) (c)

Fig. 6. People are moving from the bottom to the top while a robot learns the patterns. A) The initial model. b) An obstacle is introduced. c) The model converged to a slightly different shape due to people avoiding the obstacle.

prominent as shown in Fig. 7(c). It could also be noted that changes of trajectories cause some of the states to shift slightly.



(a) (b) (c)

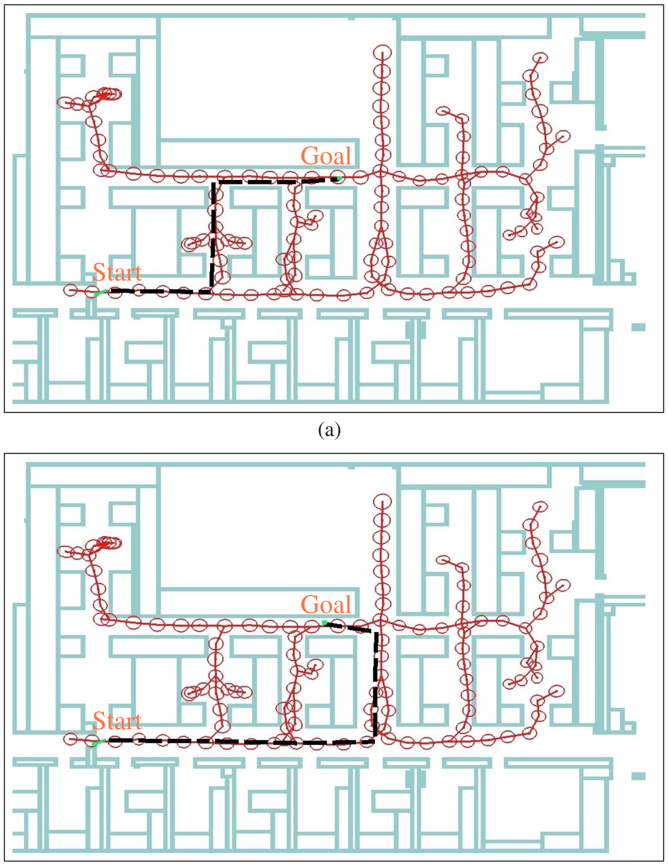
Fig. 7. People are moving from the bottom to the top while a robot learns the patterns. a) Initial trajectory without obstacles, b) An obstacle is introduced, c) The model adapted and converged according to the new information.

Finally, a larger model is presented in Fig. 8(a) which is the result of observing more than 50 trajectories. While observing the robot traveled between various locations in the environment. Fig. 8(b) shows the associated traffic density which was observed. It could be noted that as expected the traffic density in desk areas is lower than in corridors. In the following experiments, the model learned here will be used for path planning.

C. Path Planning

In the following we present our path planning results based on the above learned model. Examples are shown where the robot is supposed to avoid office spaces, i.e. a high value for w_g in Eq. 12, and for comparison normal shortest path results are presented alongside.

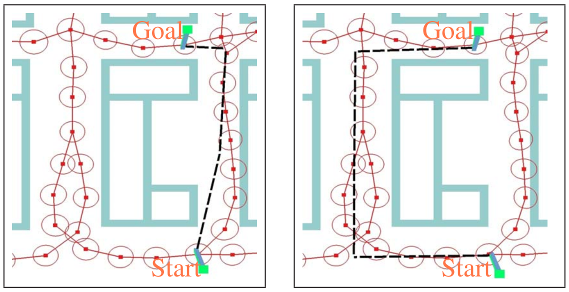
Fig. 9(a) shows a typical result of a planned path using only basic A^* without exploiting further knowledge. It plans the shortest path through the desk areas, which may disturb the occupant. In contrast when using the proposed model of motion patterns, the path will be consistently chosen not to invade any work spaces, as can be seen in Fig. 9(b) with the trade-off of a slightly longer path.



(a)

(b)

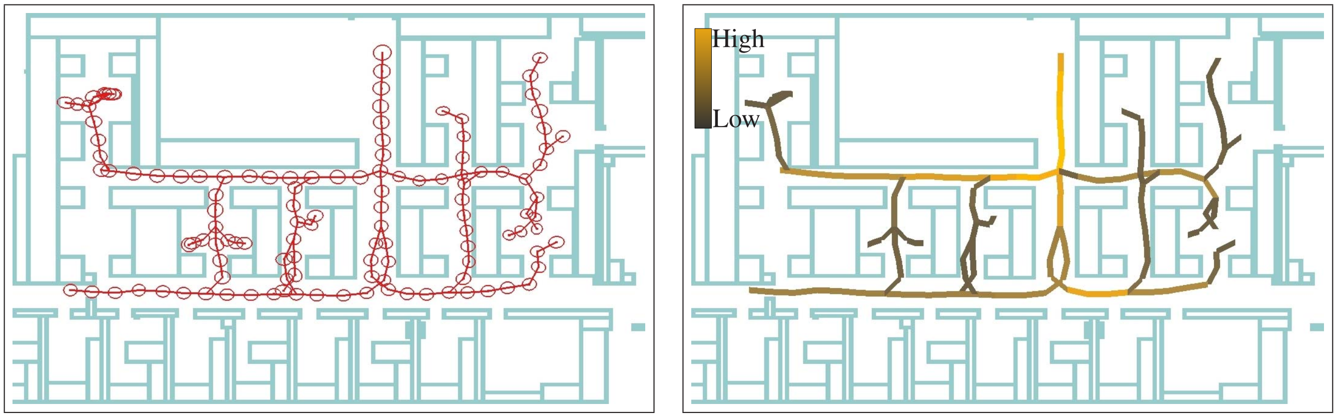
Fig. 9. Path planning, a) A path generated using basic A^*. b) Path generated with regard to the motion pattern model



(a) (b)

Fig. 10. a) A path generated using basic A^*. b) Path generated with regard to motion pattern model, which is significantly longer than the shortest path.

Consider another example, in which the shortest path is obviously going through someones work space. In this case, as expected, basic A^* will always generate a path which would send the robot through the desk areas (see Fig. 10(a)). However, setting a high value for w_g in Eq. 12 will guarantee that the robot follows the learned model avoiding areas where it potentially disturbs occupants. This phenomenon is evident in Fig. 10 where the robot actually plans a considerably longer path in order to stay out of the desk area. From these two examples it can be seen that different values of w_g would yield different results.



(a)

(b)

Fig. 8. Learning human motions, a) A model of motion patterns learned by our mobile robot. More than 50 observed trajectories result in this model. b) The observed traffic density.

V. CONCLUSIONS

In this paper, we have discussed the aspects of a robotic co-worker operating in an office environment with respect to its social awareness. An approach was presented to learn common motion patterns of people using the sensors mounted on a mobile robot without having to rely on infrastructure mounted sensors. People were tracked using a laser range finder's observations and the tracks were used to dynamically learn a Sampled Hidden Markov Model. One of the key features of the learned model is its adaptability to new situations while maintaining a computationally simple representation. As the trajectories of people are not random, this model leads to rich information with respect to robotics path planning and other tasks.

Furthermore, in the context of path planning it was outlined what influences can lead to the appearance of motion patterns and from this it can be seen how the SHMM can be applied to improve robotic path planning.

Furthermore, the learned human motion patterns were effectively utilized in robotic path planning. A modified cost function for the popular A^* algorithm was presented, which allows the robot to influence the path planner to behave in different ways based on the tasks at hand.

Future research is underway to further exploit SHMMs to improve people tracking by using the learned motion models. Further, it is intended to develop sensor utilization algorithms for mobile robotics applications based on motion models.

ACKNOWLEDGMENT

This work is supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government.

REFERENCES

[1] G. Arechavaleta, J.-P. Laumond, H. Hicheur, and A. Berthoz, "The nonholonomic nature of human locomotion: a modeling study," in *IEEE / RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics*, 2006.

[2] E. T. Hall, *The Hidden Dimension - Man's Use of Space in Public and Private*. London: The Bodley Head Ltd, 1969.

[3] C. B. Danielsson, "Differences in perception of noise and privacy in different office types," *The Journal of the Acoustical Society of America*, vol. 123, no. 5, p. pp. 2970, 2008.

[4] E. Swears, A. Hoogs, and A. G. A. Perera, "Learning motion patterns in surveillance video using hmm clustering," in *IEEE Workshop on Motion and video Computing*, Jan. 2008, pp. 1–8.

[5] D. Vasquez, C. Fraichard, and C. Laugier, "Incremental learning of statistical motion patterns with growing hidden markov models," in *International Symposium of Robotics Research*, November 2007, pp. 348–353.

[6] E. Kruse, R. Gutsche, and F. Wahl, "Acquisition of statistical motion patterns in dynamic environments and their application to mobile robot motion planning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 2, Sep 1997, pp. 712–717.

[7] M. Bennewitz, W. Burgard, and S. Thrun, "Adapting navigation strategies using motions patterns of people," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, September 2003, pp. 2000–2005.

[8] S. Sehestedt, S. Kodagoda, A. Alempijevic, and G. Dissanayake, "Efficient learning of motion patterns for robots," in *The Australasian Conference on Robotics and Automation (ACRA)*, 2009.

[9] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. IEEE*, pp. 257–286, Feb. 1989.

[10] B. Fuglede and F. Topsoe, "Jensen-shannon divergence and hilbert space embedding," in *International Symposium on Information Theory (ISIT)*, June 2004, p. 31 ff.

[11] M. Bolić, P. M. Djurić, and S. Hong, "Resampling algorithms for particle filters: A computational complexity perspective," *EURASIP Journal on Advances in Signal Processing*, vol. 2004, no. 15, pp. 2267–2277, 2004.

[12] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Piscataway, NJ, USA: IEEE Press, 2009, pp. 3931–3936.

[13] S. M. LaValle and M. S. Branicky, "On the relationship between classical grid search and probabilistic roadmaps," *The International Journal of Robotics Research*, vol. 23, no. 7-8, pp. 673–692, 2004.

[14] T. Allen, A. Hill, J. Underwood, and S. Scheding, "Dynamic path planning with multi-agent data fusion - the parallel hierarchical replanner," in *Robotics and Automation (ICRA). IEEE International Conference on*, May 2009, pp. 3245–3250.

[15] M. Likhachev, D. Ferguson, G. Gordon, A. T. Stentz, and S. Thrun, "Anytime dynamic a*: An anytime, replanning algorithm," in *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, June 2005.

[16] S. Sehestedt, S. Kodagoda, and G. Dissanayake, "Robot path planning in a social context," in *IEEE International Conference on Robotics, Automation and Mechatronics (RAM)*, June 2010.