Spatial and Probabilistic Modelling of Pedestrian Behaviour

Dimitrios Makris & Tim Ellis
Information Engineering Centre
School of Engineering, City University
London, EC1V 0HB, UK
{d.makris, t.j.ellis}@city.ac.uk

Abstract

This paper investigates the combination of spatial and probabilistic models for reasoning about pedestrian behaviour in visual surveillance systems. Models are learnt by a multi-step unsupervised method and they are used for trajectory labelling and atypical behaviour detection.

Keywords: visual surveillance, route detection, behaviour analysis, HMM

1 Introduction

The aims of our research are to construct activity-related scene models and learn patterns of movement by observing the scene over long periods using video sequences of unconstrained outdoor environments. From this information, we can identify typical and atypical activity. In this paper, we investigate how the combination of geometric and probabilistic models can be used in a video surveillance system. The objectives are to build these models to answer questions like “Is this pedestrian’s behaviour typical?” “Where is this pedestrian moving towards?” and “What is the route of the pedestrian?”

Our approach is an unsupervised multi-step learning method. First, a geometry model is learnt and then probabilistic models are overlaid onto it. This approach is motivated by the method by which human brain develops its visual perception: it seems to firstly build high-level abstract models that are enriched with more detail over time.

In the earlier work [1], we constructed a geometric model of common routes through a scene using a discrete representation generated by observing a sample of pedestrian trajectories automatically extracted from video sequences. The model was described by a central spline bounded by an envelope representing the extrema of the observed trajectories ascribed to the route model.

In this paper, we expand the model by adding probabilistic representations to improve its reliability. The model is augmented by modelling the distribution of trajectories across the route, supporting a more detailed description of activity and then a Hidden Markov Model (HMM) is constructed. In many cases, a single HMM is unable to express the variation of the route usage at different times, e.g. during “rush hours”, late at night or over the weekend. Hence, we construct multiple HMMs that reflect this temporal variation.

BMVC 2002 doi:10.5244/C.16.54

In [1], geometrical route models were proposed that can encode the common pedestrians' paths and an algorithm that allows learning of these models from a set of trajectories. A similar method was presented in [2]. However, this method fails to encode the width of the routes, resulting in loss of information about the real geometry of the paths. In [3], low-level models (state models) are learnt initially and then mid-level models (behaviour models) are constructed. Object behaviour prediction is performed by a Markov Chain. However, this approach lacks high-level interpretation of the scene and the behaviour. A Hidden Markov Model is used in [4] for behaviour recognition. The method requires only the entry/exit areas of the scene, which are defined manually, and segments the scene by uncovering the hidden states of the model. However, results were only provided for video sequences derived from a well-controlled environment, so the efficiency of the method in real life is questionable. An Abstract Hidden Markov Model, which is a type of multi-layer Dynamic Probabilistic Network, was used in [5] to predict object behaviour over a wide spatial area. The method requires a hierarchy of locations, but these are defined manually. Summarizing, whilst many researchers have tried to learn either geometric or probabilistic models of activity, few of them provide a framework to learn both, automatically.

In section 2, we summarize the geometric route model notation and introduce a Fuzzy Logic Trajectory Classifier (FLTC) that is used during the learning process of the geometric models. In section 3, we present probabilistic models that are overlaid onto the geometrical ones and a Maximum Likelihood Trajectory Classifier (MLTC) as a more reliable alternative to the FLTC. In section 4, we present some results of applying the algorithms in a real video surveillance system.

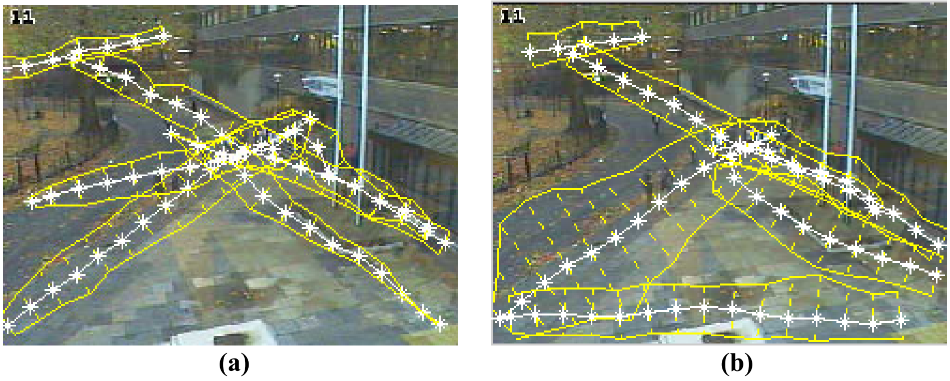
2 Geometrical Modelling

2.1 Route Modelling

Pedestrian motion can be detected, tracked [6] and simply represented as a trajectory, a sequence of points on the image plane. For a given object i that entered the scene at time t_0 and exited the scene at the time t_n, this can be expressed as $\vec{p}_{i,t_j} = \{x_{i,t_j}, y_{i,t_j}\}$, $t_j = t_0, t_1 \ldots t_n$. Assuming that the differences (t_{j+1}-t_j) are small , the sequence can be expressed as a continuous function of time $\vec{p}_i(t) = \{x_i(t), y_i(t)\}$, $t_0 \leq t \leq t_n$.

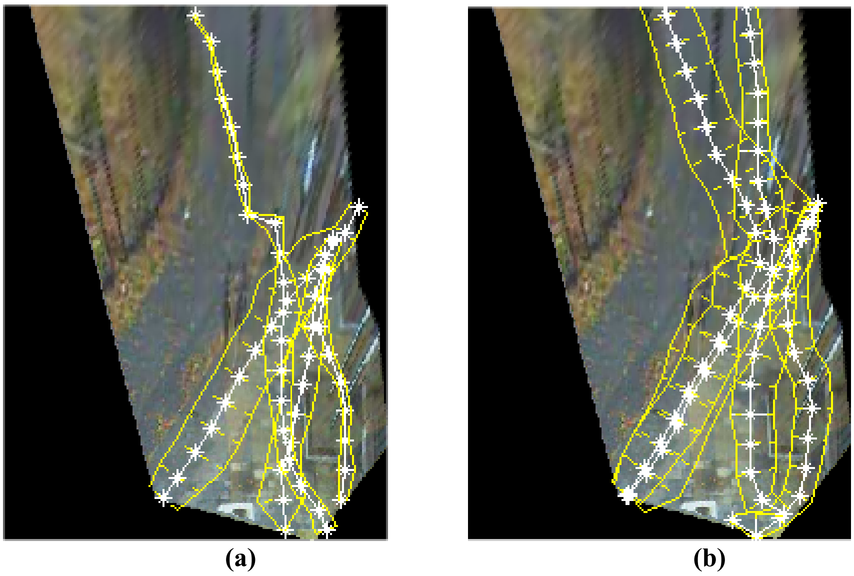
Each route model consists of a sequence of N nodes. Each node is characterised by the position of the node on the image plane $\vec{x}_k = \{x_k, y_k\}$, two bound points left and right of the node $\vec{l} = \{lx_k, ly_k\}$, $\vec{r}_k = \{rx_k, ry_k\}$ and a weight w_k which indicates the popularity of the node. The main axis of the route and its boundaries are formed by the set of node positions and bound points respectively. The geometric route models are built using an unsupervised learning algorithm [1].

Figures 1 and 2 show the main route models computed for both image plane and ground plane trajectories (low popularity routes have been ignored, because they represent a relatively low usage). Using 2D ground plane coordinates requires no change to the model but derives several benefits. Because the model aims to encode the physical scene, it is obvious that a ground-plane based representation is closer to reality and the parameters that are used in the learning algorithm correspond to physical quantities. As a result, the algorithm is more sensitive to detecting routes that are more distant from the camera. In addition, by utilising a common ground-plane coordinate system for a set of cameras, integration of route models across multiple cameras is possible. In this case, the trajectories are provided in ground plane coordinates by a camera calibration model [7].



(a) (b)

Figure 1: Route Models learnt over (a) 1-hour period using 647 trajectories and (b) 24-hour period using 3460 trajectories.



(a) (b)

Figure 2: Route models learnt on ground-plane coordinates, over (a) 1-hour period using 317 trajectories and (b) 24-hour period using 2744 trajectories.

2.2 Fuzzy Logic Trajectory Classifier (FLTC)

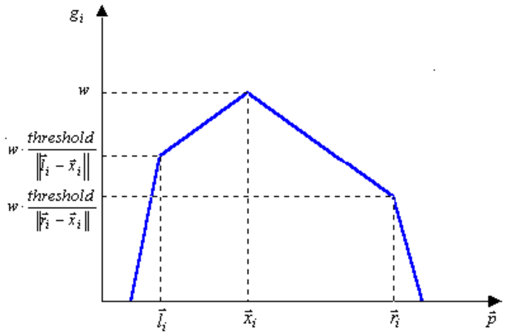
Trajectories or parts of trajectories are classified into route models, either during the learning process or during the normal operation of the surveillance system. In [1], the trajectories are classified by a Boolean classifier, according to a distance criterion. This approach is appropriate when trajectories are in areas where routes are not overlapped and no ambiguity exists. However, when substantial overlapping occurs, an uncertainty arises, which cannot be quantified by the Boolean classifier. An optimal approach (in a

Maximum Likelihood sense) would require knowledge of the distribution of the observations across each route. Because this distribution is not known prior to the generation of the geometric model, we introduce a method that encodes the trajectory distribution *across* a route, and uses Fuzzy Logic to bootstrap the route construction process.

Using this approach, certainty estimates are calculated for point-node matches and then an overall certainty is estimated for the trajectory-route match. An asymmetric membership function is used (Fig.3), that captures the characteristics of the distribution. The distribution is modelled by a function g of a point $\vec{p}$ lying in the normal direction of the node i and is defined as:

$$g_i(d) = \begin{cases} w_i\left(\cdot\dfrac{threshold - d}{\left\|\vec{b}_i - \vec{x}_i\right\|} + \dfrac{threshold \cdot d}{\left\|\vec{b}_i - \vec{x}_i\right\|^2}\right), & d \le 0. \\ w_i \cdot \dfrac{threshold - d}{\left\|\vec{b}_i - \vec{x}_i\right\|}, & 0 < d < threshold. \\ 0, & d \ge threshold. \end{cases} \qquad (1)$$

where $d = \left\|\vec{p} - \vec{x}_i\right\| - \left\|\vec{b}_i - \vec{x}_i\right\|$ and $\vec{b}_i = \vec{l}_i$ or $\vec{b}_i = \vec{r}_i$, under the condition $\left\|\vec{p} - \vec{x}_i\right\| \cdot \left\|\vec{b}_i - \vec{x}_i\right\| \ge 0$. If $\left\|\vec{b}_i - \vec{x}_i\right\| < threshold$, we set $\left\|\vec{b}_i - \vec{x}_i\right\| = threshold$ at (1). The threshold distance is defined as the distance either side of the route boundaries where matching is still possible.



b

Figure 3: Proposed membership function that models the distribution of observations across a route node. $\vec{x}_i$ is the position of the node and $\vec{l}_i$, $\vec{r}_i$ are the two bound points. The point $\vec{\text{p}}$ is supposed lying on the normal direction of the route.

To classify a trajectory within a route model j using the FLTC, the trajectory is sampled by the node normals of the route. For each trajectory sampled-point $\vec{p}_{i_j}$ a weighted estimate [8] g_{i_j} is given according to (1). Then certainty estimate of a trajectory-route match $h'_j(\{\vec{p}\})$ is given by the minimum of the certainty estimates of the point-node matches (2). Finally, the certainty estimate is normalized by dividing by the sum of the certainty estimates for all the trajectory-route possible matches (3). The trajectory { $\vec{p}$ } is classified to the route j that maximizes the certainty estimate.

$$h'_j(\{\vec{p}\}) = \min_{i_j}\left(g_{i_j}(\vec{p}_{i_j})\right) \tag{2}$$

$$h_j(\{\vec{p}\}) = \frac{h'_j(\{\vec{p}\})}{\sum_k h'_k(\{\vec{p}\})} \tag{3}$$

The FLTC is used during the learning process to match trajectories with existing route models. In the case that multiple matching occurs, all the matched route models are updated according to the particular certainty estimations.

3 Probabilistic Modelling

The geometric model that has been learnt in the previous section is used as a basis for the probabilistic models that are built in this section. In section 3.1, distributions of observations across the routes are learnt and the Maximum Likelihood Trajectory Classifier is introduced. In section 3.2, a Hidden Markov Model is built, aiming to enable the interpretation of pedestrian behaviour.

In probabilistic terms, the set of trajectories is considered as a stochastic process. We begin with a simplifying assumption that the occurrence of trajectories is stationary and does not vary significantly over time. In section 3.3, we discuss the validity of this assumption.

3.1 Cross-Route Distributions and ML Classifier

Having acquired the geometric model, distributions of observations across the routes are learnt and incorporated into the model. Figure 4 shows the estimated pdf of observations across selected nodes for a specific route.

The knowledge of the distributions across the route allows the use of a Maximum Likelihood Trajectory Classifier (MLTC) as a more accurate alternative to the FLTC. If g_{i_j} represents the pdf. of observations across node i_j of route j, the probability $h'_j(\{\vec{p}\})$ of a trajectory $\vec{p}$ under the condition that it belongs to the route j is given by (4), assuming independent observations $\vec{p}_{i_j}$. According to the ML criterion, the trajectory is assigned to the route that maximizes the probability h_j as given by (5).

$$h'_j(\{\vec{p}\}) = \prod_i g_{i_j}(\vec{p}_{i_j}) \tag{4}$$

$$h_j(\{\vec{p}\}) = \frac{h'_j(\{\vec{p}\})}{\sum_k h'_k(\{\vec{p}\})} \tag{5}$$

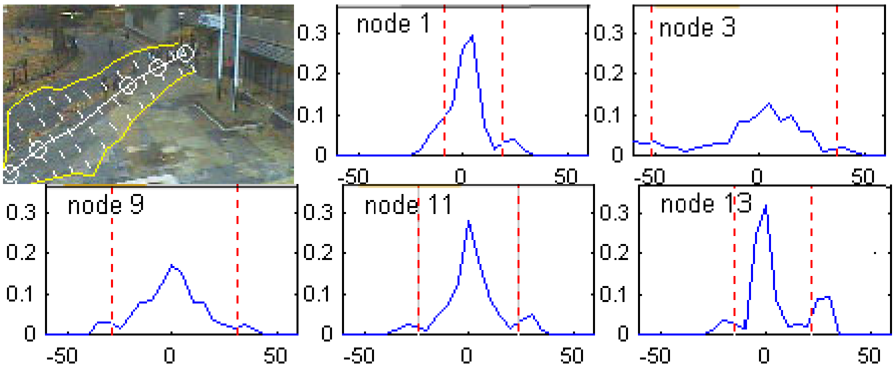


Figure 4: Route model and derived distributions of observations across five selected nodes. Node ordering is from bottom left to top right along the route. Red dashed lines indicate the boundaries (envelope) of the route at each node. The x-axis indicates the distance from the main axis of the route in pixels.

3.2 Hidden Markov Model

The geometric model of the scene allows the assignment of trajectory points to route nodes and by extension, the classification of trajectories to routes. The geometric model is a hierarchical discrete representation in two levels: the high level model represents routes as a series of paths and junctions, whilst the low level model is based on the route nodes. The discrete nature of the model allows discrete-state models like Markov Chains and Hidden Markov Model (HMM) to be applied. Therefore, four possible Markov models could be built, depending on the level of the scene interpretation and the type of the model. We select to build a Low Level (LLHMM) because it can encode a wider range of activity in the scene.

We define the states of a Low Level Hidden Markov Model to be the nodes of all the accepted route models, plus two extra states: an "out-of-any-node state", which indicates activity outside the modelled routes and an "end state", which indicates the end of the observation. It is sensible to derive the nodes from uni-directional routes, so that directionality information is incorporated at each node.

The elements of the LLHMM are:

- **S**=$\{S_i\}$, i=1..N, the set of states.
- **Q**=$\{q_k\}$, k=1..M, the sequence of the states.
- **A**=$\{a_{ij}\}$, i,j=1..N, the transition probability distribution, where $a_{ij}=P(q_{t+1}=S_j \mid q_t=S_i)$.
- $\pi=\{\pi_{i,}\}$ i=1..N, the initial state distribution, where $\pi_i=P(q_1=S_j)$.
- **O**=$\{O_k\}$, k=1..M, the sequence of the observations.
- **B**=$[bi(O_k)]$, i=1…N, O_k: observation vector, where $bi(O_k)$ is the pdf of the observation vector.

The HMM parameters are recommended to be learnt using iterative algorithms. Because of the large number of states, these algorithms are very slow and often impractical, especially for online learning. Instead, we use the pdf distributions of observations across the routes to encode the observation vector **B** (6). Then the LLHMM parameters are estimated cumulatively (7), (8).

$$b_i(O_{l.k}) = \frac{g_i(O_{l.k})}{\sum_{j}^{N} g_j(O_{l.k})} \tag{6}$$

$$\pi_i = \frac{\sum_{l=1}^{L} b_i(O_{l,1})}{L} \tag{7}$$

$$\mathrm{a}_{\mathrm{ij}} = \frac{\left(\sum_{l=1}^{L}\sum_{k=1}^{K_L} b_i(O_{l,k}) \cdot b_j(O_{l,k+1})\right)}{\sum_{l=1}^{L}\sum_{k=1}^{K_L} b_i(O_{l,k})} \tag{8}$$

where $O_{l,k}$ is the kth observation from the lth trajectory l=1..L, k=1..K_L.and $g_i(O_{l,k})$ is the estimate of the probability that the observation $O_{l,k}$ corresponds the state i, according to the associated learnt cross-route distribution.

HMMs can provide temporal predictions. The probability of a state j after K time-steps, assuming the observation vector **O** is given by the following equation:

$$P(Q_{k+K} = S_j \left| O_1 O_2 ... O_k \right.) = \sum_{i=1}^{N} \delta_k(i) \cdot \mathrm{a}_{\mathrm{i,j}}^{(\mathrm{K})} \tag{9}$$

where $a_{i,j}^{(K)}$ is the element {i,j} of the matrix A^K and $\delta_K(i)$, defined in [9], represents the maximum probability of the state i at the time-step K, given the observation vector.

In order to derive spatial predictions, for instance the probability of an object exiting the scene at node j, HMM theory cannot provide a well-defined solution to this problem. This is because it must consider the infinite number of different sequences of states that lead to termination of object at the node j. An implicit estimation is provided by a route-level interpretation and using the probability $\delta_K(i)$.

3.3 Long Term Temporal Variations

The HMM that we described in the previous section is built under the assumption that the trajectories are generated by a stationary stochastic process. In many cases, this assumption is invalid, as pedestrian behaviour may be dependent on the time of day. For instance, for the scene of Fig. 1, which is the entrance of the University, we expect that in general, between 8am and 10am, most pedestrians will walk to the entrance of University; from 1pm to 2pm, they will wander around outside the entrance; at 5pm many will leave the University and at midnight, hardly anyone will be around.

However, we can assume that pedestrian behaviour remains the same for short time periods (e.g. 15 min-1 hour) and this pattern is repeated every weekday at the same time. In addition, we assume that this time period is long compared to the lifetime of a trajectory. Therefore, we can establish a static HMM model for each time slot. It is obvious that such models need learning data from more than one day, so sufficient observations are provided for estimating the models of each time slot.

In (7) the initial probability π_i of the state i has been defined over the sum of the observed trajectories. However, a more reasonable approach is to multiply this relative probability with the frequency of the pedestrians' appearing to derive an absolute measure of the probability of a new appearance over time: $\pi_i' = \pi_i \cdot \frac{L}{T}$. To illustrate the advantage of π_i' over π_i, let's consider the case of 100 out of a total of 1000 pedestrians starting their route from node i, between 10am-11am and 1 out of a total 10 pedestrians starting from node i, between 10pm-11pm. In both cases, π_i takes the 0.1 value, but this fails to indicate the fact that a pedestrian appearing so late is anyway atypical, which is made clearer by using π_i'.

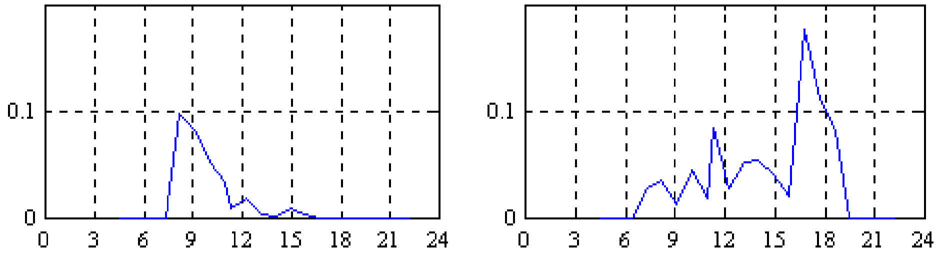


Figure 5: Initial probability π_i of the two terminator nodes of the route in fig.4, during a 24-hour period. x-axis indicated the time of the day. At 8-9am, almost all trajectories occur towards the entrance of the University, whilst around 5pm, people tend to leave.

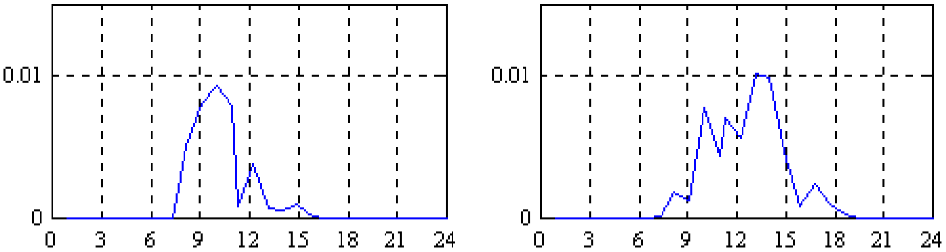


Figure 6: Initially probability π_i' over time for the two terminator nodes of the route in Fig.4. The peaks of the distributions have moved to 10am and 1-2pm, indicating that these are the most popular periods for people to come and leave the University respectively.

4 Experimental Results

Our motivation for investigating the probabilistic models is their applicability in a Video Surveillance system. In this section, we show how the proposed methodology is used for trajectory labelling and atypical behaviour detection.

Behaviour recognition is performed by classifying the object trajectory to one of the existing routes. A Boolean classifier based only on the geometric model is sufficient to classify trajectories that match only one route for their entire length, but it cannot classify incomplete trajectories that entirely lie on overlapped routes. Such an example is illustrated in Fig.7. In the beginning of the sequence, the orange route dominates due to its very high popularity, but gradually the green route is matched. In this case, the MLTC provides a more appropriate interpretation.

In the case that an object jumps from one route to another, the trajectory cannot be classified only to one route as different parts correspond to different routes. The problem can be solved by splitting the trajectory at the point where all the state probabilities have become zero, which indicates that single matching is impossible. Then, the rest of the trajectory is matched to a different route.

Identifying atypical trajectories is performed by estimating the probability of the trajectory, which is equivalent to the Observation Evaluation Problem. The solution, provided in [9], has been used for the results of Fig.8.

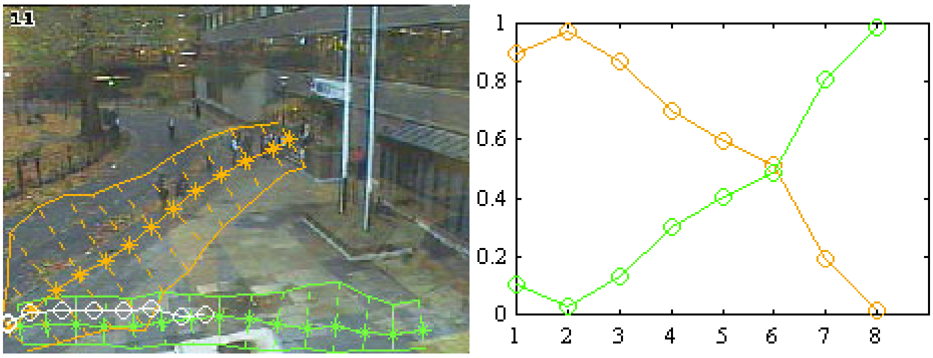


Figure 7: Classification of an incomplete trajectory online by the ML classifier. The graph shows a plot of likelihood against time (in seconds) for matching the trajectory to the two route models.

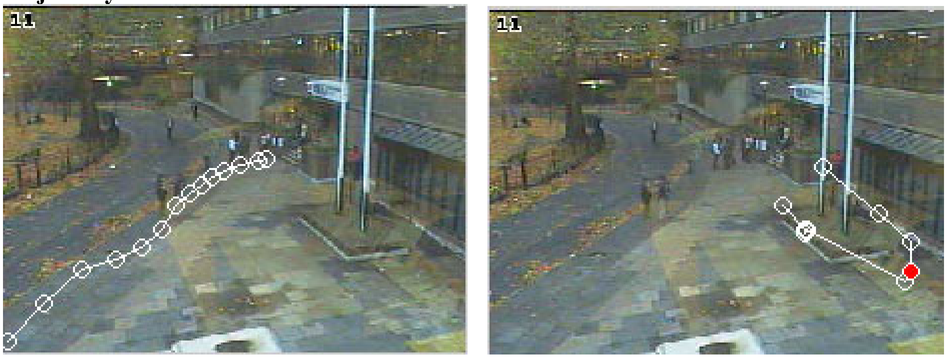


Figure 8: Evaluated trajectories are shown. Left trajectory characterised as typical, while the right trajectory is characterised as atypical (the red filled circle locates where this characterization is performed). Bear in mind that the left trajectory could be atypical if it occurred at a different time.

5 Conclusions – Future Work

A collection of models for pedestrian behaviour analysis has been presented. Geometric and probabilistic models are built sequentially, in a multi-step learning approach. First, a geometric model of the scene is learnt which defines its main routes. Then distributions are estimated that define the density of the observations across the routes. Finally, a low level HMM is superimposed on the geometric model. The entire learning process is fully automated and unsupervised. This set of the models has been used in an automatic video surveillance system for trajectory labelling and atypical behaviour detection.

Trajectory classification is required throughout the learning process and during online operation time. A Fuzzy Logic Trajectory Classifier is used during the geometric model learning. Next, the estimation of the cross-route distributions allows a Maximum Likelihood Trajectory Classifier to be used during both the HMM parameter estimation process and the pedestrian behaviour analysis.

In the future, we will incorporate the set of models into the real time motion tracking system that we are developing. We aim to integrate route models across multiple cameras. We also aim to extend the use of probabilistic models in order to encode more complex behaviours.

Acknowledgements

The authors acknowledge financial support from the Engineering and Physical Sciences research council (EPSRC) under grant number GR/M58030. We are grateful to Ming Xu for supplying the trajectory data and to James Black for supplying the camera calibration models.

References

[1] Dimitrios Makris and Tim Ellis. *Finding Paths in Video Sequences*, British Machine Vision Conference 2001, Manchester, UK, September 2001.

[2] Esther B. Koller-Meier and Luc Van Gool, *Modeling and Recognition of Human Actions Using a Stochastic Approached* , 2nd European Workshop on Advanced Video-Based Surveillance Systems, Kingston, UK, September 2001.

[3] Neil Johnson. *Learning Object Behaviour Models*, PhD thesis, School of Computer Studies, University of Leeds, UK, September 1998.

[4] Michael Walter, Alexandra Psarrou and Shaogang Gong, *Learning Prior and Observation Augmented Density Models for Behaviour Recognition*, British Machine Vision Conference, Nottingham, UK September 1999.

[5] Hung H. Bui, Svetha Venkatesh and Geoff West, *Tracking and Surveillance in Wide Area Spatial Environments Using the Abstract Hidden Markov Models*, International Journal of Pattern Recognition and Artificial Intelligence. To appear.

[6] Ming Xu and Tim Ellis, *Object Detection and Tracking in an Open and Dynamic World,* 2nd IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, PETS2001, Hawaii, USA, December 2001.

[7] James Black and Tim Ellis, *Multi Camera Image Tracking*, 2nd IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, PETS2001, Hawaii, USA, December 2001.

[8] K.R. Sasikala and Maria Petrou, *Properties of the generalised fuzzy aggregation operators*, Pattern Recognition Letters, Volume 22, Number 1, pp.15-24, January 2001.

[9] Lawrence R. Rabiner, *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*, Proc. of the IEEE, Vol. 77, no. 2, pp. 257-286, February. 1989.