Chapter 7

LEARNING AND INTEGRATING INFORMATION FROM MULTIPLE CAMERA VIEWS

Mapping an Ambient Environment

D.Makris, T.Ellis and J.Black

Digital Imaging Research Centre (DIRC), Kingston University, UK

{d.makris,t.ellis,j.black}@kingston.ac.uk

Keywords: Activity models, scene recognition, visual learning, video surveillance.

1. Introduction

The primary goal of visual surveillance and monitoring is to understand and interpret the activities of objects of interest (typically people and vehicles) within an environment. The current deployment of such systems mainly fulfils a role of security surveillance, providing remote viewing and recording of video information from CCTV cameras to help protect people and property, detecting illegal, anti-social or invasive actions. However, as such systems become more pervasive in urban environments, their potential is to be used to provide a much wider range of interactive services that can both facilitate and anticipate the needs and wants of people.

A fully functioning system would be capable of recognising and predicting the behaviour of individuals and groups, employing specific models representing the activity patterns. This should also enable the system to understand the typical or expected behaviours within a particular environment, and then to be capable of identifying unusual or atypical behaviours. To do so, the system would benefit from contextual knowledge relating to the environment. Such knowledge might be both local and global. For instance, consider the activities associated with a ticket office in a railway station, and its local environs. The office will generate particular types of activity as people enter the scene from (often) well-defined regions and move towards the office. At busy times people will need to queue to access the ticket seller(s), undertaking their transaction then departing to some other part of the station.

It is also desirable for this high level knowledge to be derived by the system itself, rather than be given explicitly by a human operator. Therefore, the system can be installed with a minimum effort ("plug'n'play") and it can adapt to any changes of its environment. Such a capability is very useful, especially for large scale distributed surveillance systems where the calibration of newly installed cameras or recalibration of old cameras can be a time-consuming and skilled task.

For pedestrians we can begin by identifying a number of 'primitive' activities such as walking, sitting and queuing that can be inferred from tracking data. Whilst these activities can be derived for individual objects, it is valuable to be able to recognise typical activities, commonly performed by many objects, and to relate these to scene-dependant contextual knowledge. It is interesting to note that many of the behaviours that we might want to identify as part of a surveillance or monitoring task occur when the objects are stationary (or near-stationary), so the detection of 'stopped' objects is a significant event in the tracking process.

The interaction of people with the environment is constrained by the spatial geometry. Most modern man-made environments are ergonomically designed to facilitate efficient access to popular locations, anticipating flow-rates and densities to minimise the effects of over-crowding, or to ensure that there are no isolated areas where individuals may feel vulnerable or threatened. Whilst it is possible to incorporate geometrical models of an environment into a surveillance recognition system, such models would need further augmentation to express the typical activities of people interacting with the environment. Furthermore, these activities are not stationary, in the sense that at different times of the day, week or year, they can change. In the case of the ticket office example, there will be certain times when the office is closed and when people enter the scene, their actions will be quite different.

In this chapter we develop models to represent the common activities that are observed within an outdoor scene. The models will be learnt from an analysis of trajectory data extracted from the scene over a long observation period (10-20 hours). The models combine spatial and probabilistic information to create a representation that can be used to describe both the spatial geometry associated with the activity, as well as a statistically-derived likelihood of usage. The models are used to encode regions in the imaged scene where various primitive activities occur: objects entering or leaving the camera view (entry and exit zones); the common paths or routes taken by objects moving through the scene; and stop zones, where objects come to rest for some minimum period.

The models can be used to support a variety of different uses in the online surveillance system, such as annotation and atypical behaviour detection. In annotation, individual trajectories can be described using only a small number of parameters associated with a particular activity or set of activities: for example: "object #27 entered the scene at entrance C at 16:51:20, moved along path 4, stopped at location L2 from 16:51:56-16:52:11, then proceeded to exit the scene from exit A at 16:52:44". Such annotation provides a meta-data layer in the trajectory database, allowing a more efficient means to describe objects and create global usage statistics of activities. In addition, the annotation can be combined with specific contextual scene knowledge (e.g. that location L2 corresponds to the region in front of the ticket office) to semantically enrich the annotation and facilitate human-centric queries to the database (e.g. extract all visits to the ticket office between 11:30-12:30). The typical behaviour of objects is encoded into the probabilistic component of the model, and can be used in combination with the spatial component to identify an action that is unusual, at least across the learning set from which the models were constructed.

We also employ the entry/exit models for each camera view to automatically learn the topology of an arbitrary network of video cameras observing an environment. The topology is learnt in an unsupervised manner by temporally correlating objects transiting between adjacent camera viewfields, establishing the correspondence of links between the entry and exit zones of cameras in the network. A significant benefit of the method is that it doesn't rely on establishing explicit correspondence between trajectories, and results in a measure of inter-camera transition times, which can be used to support predictive tracking across the camera network.

The trajectory data used to learn these models is extracted by querying the tracking database described in chapter 6.

1.1 Semantic Scene Model

A semantic description of activity is usually given in relation to semantic elements of the scene, e.g.: "John entered the house from the front door, walked along the corridor, sat at the desk and then left from the back door". However, to allow automatic description of such activities (video annotation) from visual surveillance, semantic labels, like "front door", "corridor", "desk" must be defined. Ideally, these features would be automatically recognised by the visual surveillance system.

In the context of this chapter, semantics are defined in relation to the activity of targets. For instance, doors are characterised as a feature associated with entry/exit events, a desk is an element of interest that targets may stop nearby and a pavement is a common path that pedestrians move along.

A semantic model of the scene is introduced. The semantic labels of the scene regions are associated with the activities that are performed on these regions. The model must provide both a spatial and probabilistic representation in order to characterise the target activity in terms of spatial features of the scene and express the level of usage and the associated uncertainty, supporting a predictive capability. The model includes regions associated with a particular semantic interpretation, such as entry/exit zones, paths, routes, junctions and stop zones.

Figure 7.1 shows a simplified depiction of an outdoor scene, consisting of a number of interconnected pathways. The seat icon (I, J) represents regions where targets may stop, while other labels are associated with entry/exit regions (A, C, E, G, H) and junctions (B, D, F) where targets moving along the pathways may change their routes. The segments between entry/exit zones or/and junctions (AB, CB, BD, DF, FG, ...) represent paths, while routes are represented by the sequence of paths between an entry zone and an exit zone (ABDFH, CDBFE, EDFG,...).

Entry zones are regions where targets enter the scene. Similarly, exit zones are regions where targets leave the scene. An entry zone and an exit zone may

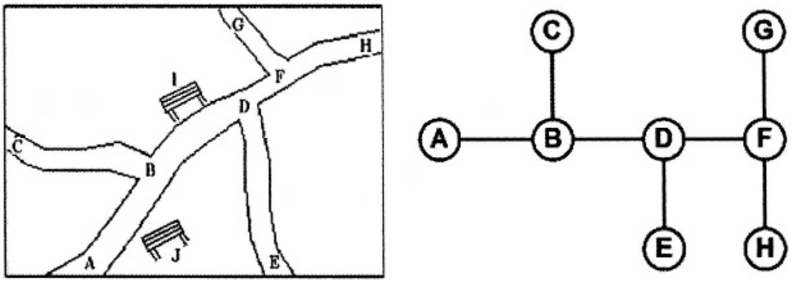

Figure 7.1. (a) Topographical map of the scene, (b) topological map of the scene. Entry/exit zones and junctions are the nodes of the network.

be coincident, e.g. as in most pedestrian environments, or not e.g. in road traffic environments, where traffic is constrained by road traffic regulations. Typically, entry/exit zones are either scene-based, e.g. doors and gates, or view-based, e.g. located parts on the boundaries of the camera view. The distinction between scene-based and view-based entry/exit zones can be useful when information from multiple cameras is integrated.

Junctions are the areas where two or more pedestrian pathways or roads meet. At junctions, there is an uncertainty about the future motion of a target, as it can follow more than one path.

Whilst in common English, paths and routes have similar meanings, they are distinguished in the context of this chapter in the following manner. Paths are segments of either pedestrian pathways or roads in between entry/exit zones and junctions. A target route is the complete history of a target activity around the scene, from its entry zone to its exit zone, through various paths and junctions. More precisely, paths should be referred to as path segments, but the above given definitions are kept for the sake of simplicity.

Stop zones are defined as the regions where targets are stationary or almost stationary, for some minimum period of time. For example, pedestrians are stationary when they stop in order to sit, rest, queue, wait to access a resource, merely observe the scene or just wander around. Stop zones are included in the scene model for two reasons: firstly, a stop zone is usually related to a physical scene feature, such as a bus stop, an ATM machine, a park seat, a shop window, a cashier, a computer, a printer, etc. Secondly, although the majority of research in video surveillance has focused on detecting and tracking motion, it is actually when targets stop and interact with each other or with these fixed elements of the scene that the system is more likely to be interested in them.

Two different presentations of the scene model are suggested: topographical and topological. The scene model is naturally represented by a topographical map (Figure 7.1a) based on either an image plane(s) or a ground map representation. Ground map representations have an advantage over image-based representations not only because they can represent the physical features of the scene in proper proportion, but also because they allow integration of information from multiple cameras.

The scene model can be also represented by a topological map, i.e. an abstract network of nodes and connections, as shown in Figure 7.1b where entry/exit zones and junctions comprise the nodes of the network connected by paths. Augmented network representations can be constructed to include additional semantic feature, such as paths and stop zones.

The two different representations are used to illustrate two different aspects of the model. More specifically, the topographical map visualises the spatial characteristics of the scene elements, as interpreted by a human, whereas the topological map can be the basis of a probability network that can be used for a probabilistic analysis of the activity.

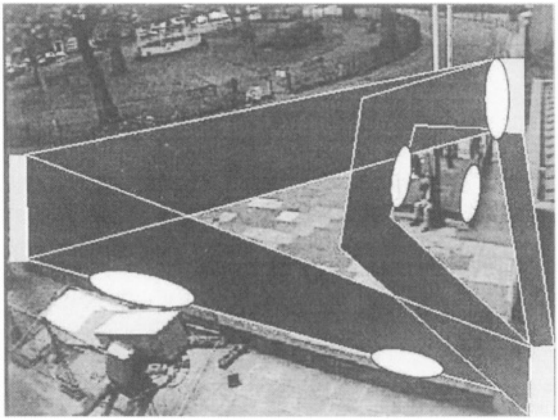

Figure 7.2. A manually derived semantic description for regions exhibiting common activities in the observed scene. The boxes at the edges correspond to the entry and exit areas of the scene, the closed polygons to commonly used paths and the ellipses to areas where pedestrians normally stop.

Spatio-probabilistic representations are learnt for the different scene semantics. Entry, exit and stop zones are related to single-point events, therefore sets of single points are used as training data. Routes, paths and junctions are related to motion that is represented by sequence ofpoints (trajectories), therefore trajectory sets are used as training data.

2. Learning point-based regions

Targets enter and exit the scene from either the borders of the image (view-based feature), or at doors or gates (scene-based feature). The first and the last successfully tracked positions of an object (in other words the first and the last point of its observed trajectory) are used to indicate the entry and the exit event. The distribution of trajectory start coordinates for a given entry zone

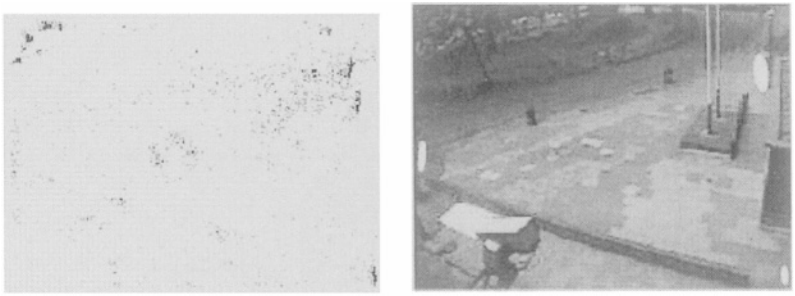

Figure 7.3. (a) Entry-point dataset (4250 samples) with both types of noise present, (b) Three entry zones derived by the multi-step algorithm. Dense clusters in the top left are eliminated. Clusters are represented as ellipses at one standard deviation.

is influenced by the speed of targets across the image plane as they enter the camera viewfield. This is dependent on the target's actual speed in the scene, the video frame capture rate and the direction of motion with respect to the camera view. Clustering many such observations allows a region of the scene to be associated with a given zone.

We choose to model the shape of the entry/exit zones using Gaussian Mixture Models (GMMs), because they compactly represent the variability and the uncertainty of the observed entry/exit events. Figure 7.3 depicts the entry/exit point datasets and fitted GMMs for a typical outdoor scene.

The entry-point/exit-point datasets can be contaminated by two different types of noise: a) tracking failure noise is caused by the failure of the tracking algorithm to continuously track objects. It is usually distributed over all the activity areas; b) semi-stationary motion noise is caused by the apparent motion of objects (trees, curtains, computer screens). It is usually densely-distributed within small areas of the image.

We conjecture that trajectory start and finish coordinates are spatially correlated with localized image regions, whilst tracking failure is spatially uncorrelated. Hence, the regions with dense observations are more likely to correspond to real entry/exit zones, whilst tracking failure noise is identified by wide Gaussian distributions over the activity areas. In contrast, although the semi-stationary motion noise can be identified by tight Gaussian distributions, they can be characterized as exhibiting very short trajectory paths, typically starting and finishing within a small area. These criteria can be used to filter the noise clusters from the signal (entry/exit zone) clusters.

We use a multi-step learning method that is based on the Expectation-Maximization (EM) algorithm [4] to learn the clusters. One of the advantages of EM is that it can successfully distinguish overlapped distributions. Therefore, if noise is overlaid onto signal, then it is possible to generate separate clusters for the signal and the noise, subject to different statistics. The two types of noise that are present in the trajectory data are also found in the entry/exit-point datasets and they have distinguishable statistical characteristic.

The multi-step algorithm first discards the semi-stationary noise and then the tracking failure noise. The actual number of entry/exit zones in the scene is unknown, however EM is a parametric method and the model order must be predefined. In order to overcome this problem, the number of entry/exit zones in the scene is initially (conservatively) overestimated and the number of the signal clusters is estimated by eliminating the noisy ones. A summary of the algorithm is given below:

1 The EM algorithm with model order N is applied to the entry-point dataset E and a GMM is derived.

2 If all the points of a trajectory belong to a single GM, as derived in the previous step, the trajectory is considered semi-stationary. A new cleaned entry-point dataset **E'** is formed that does not contain the entry points of the semi-stationary trajectories.

3 The EM algorithm with model order N' is applied to the clean entry-point data-set **E'**.

4 Gaussian clusters are classified as either signal clusters or noise clusters, according to a density criterion. More specifically, if$omega_i$ is the prior probability of a cluster *i* and Σ_i is its covariance matrix, where i=1.. N', then a measure of the density d_i is given by:

$$d_i = \frac{\omega_i}{\pi\sqrt{|\Sigma_i|}} \tag{7.1}$$

A threshold value *T* is defined by the clean entry-point dataset **E'**:

$$T = \frac{\alpha}{\pi\sqrt{|\Sigma|}} \tag{7.2}$$

where α is a user-defined weight and Σ is the covariance matrix of the dataset **E'**.

The clusters derived at the steps (1) and (3) are characterised according to their density. High-density Gaussian clusters correspond to either entry zones or semi-stationary motion noise, while low-density clusters correspond to tracking failure noise. The algorithm eliminates semi-stationary motion noise at step (2) and tracking failure noise at step (4).

Results for a camera viewing a road traffic environment are shown in Figure 7.3. Two entry zones and two exit zones were identified that are non-coincident, because they correspond to two unidirectional road lanes. (Although in this scene pedestrian traffic is normally present, the dataset was derived over a weekend, when pedestrian traffic was very light.)

In Figure 7.4, the clusters at the left side of the images are wider than the ones at the right side. The is due to the higher image plane speed of the vehicles

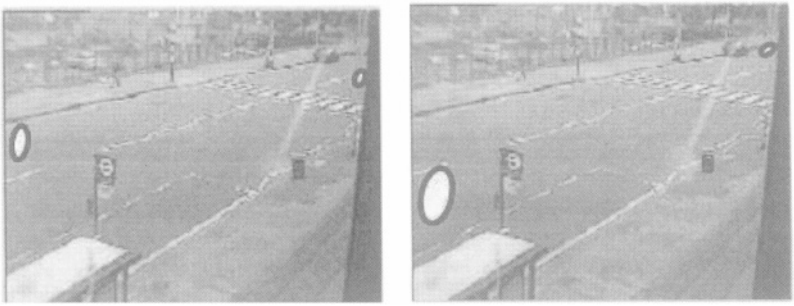

Figure 7.4. (a) Two detected entry zones in a road traffic environment (b) Two detected exit zones in a traffic environment road.

at the left side (closer to the camera), which, in combination with a low frame rate (5fps), results in a wider distribution of the first/last tracked positions for the vehicles. This difference of the distributions of the entry/exit zones is actually desirable, as the cluster explicitly determines where the targets should be initialised/terminated, according to their entry/exit speed and the system frame rate.

Stop zones are defined as regions where the targets are stationary or almost stationary. A variety of different areas can be characterised as stop zones, such as where people rest, wait for the opportunity to continue their journey (e.g. at a pedestrian crossing or a road traffic junction), access a particular resource (e.g. an automatic teller machine or at a bus stop), or merely observe the scene. Targets may also become stationary when they meet and interact with other targets, for example two people meeting in a park and sitting on a bench to chat or a vehicle waiting for a pedestrian to walk across a pedestrian crossing.

A stop event is detected when a target's speed becomes lower than a predefined threshold. A stop-event dataset is formed by checking the target trajectories for stop events. Speed is estimated in ground plane coordinates because the apparent speed on the image plane may be strongly affected by the perspective view of the camera. Therefore the stop-event dataset is formed using ground plane coordinates (see Figure 7.5).

As with the entry/exit zones, a GMM is used to model the spatio-probabilistic characteristics of the stop zones and EM is applied to learn them.

3. Learning trajectory-based regions

3.1 Route model

In road traffic environments, vehicles must follow specific predefined routes. Similarly, in pedestrian environments, people normally walk on well-prescribed pathways. Even in cases where no predefined routes exist, the structure of the

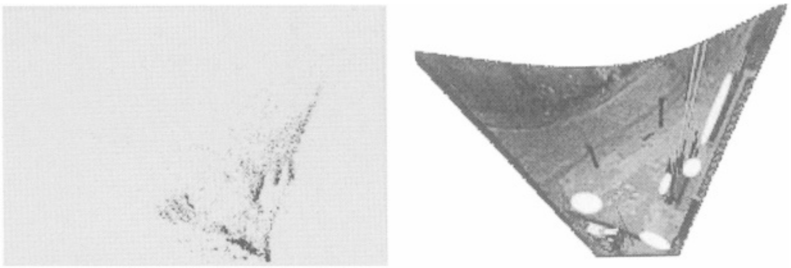

Figure 7.5. (a) Stop events (9455) on the ground plane. Stop events were detected when targets' speed drop lower than 0.25m/sec. (b) Five signal stop zones as derived by the EM algorithm.

scene affects the behaviour of pedestrians and normal route-patterns of activity exist, which is verified by results presented in this section.

A route model that is both consistent with the human interpretation and fulfils the requirements for probabilistic analysis is required. A route can be described intuitively by its start and end areas, its main axis between the start and the end and its boundaries along the main axis. Additionally, quantitative information about the usage along and across the route is required to describe the statistics of typical usage.

The scene is assumed to contain multiple routes that may have overlapped sections. A single route model must encode the following properties, using both spatial and probabilistic representation:

- The main axis of the route.
- The terminators (start and end points) of the route.
- A description of the width along the route.
- Indication of the level of usage of the route, both along and perpendicular to the direction and in comparison to the other routes.

The route model that we have developed (see Figure 7.6) consists of a discrete representation defined by a central spline axis, composed of a sequence of equidistant nodes that represents some average of the route. The constant distance between adjacent nodes is referred to as the resample factor R of the model. In addition, two bound splines around the central axis form an envelope and represent the width of the path. A route has two terminator nodes (start and end) that typically correspond to entry/exit zones of the scene. Finally, a weight factor represents the usage frequency of the route.

This route model allows explicit representation of the spatial extent of routes and therefore it is consistent with the semantic representation requirement. In addition, the probabilistic representation of the usage, both along and across the route, and the discrete nature of the model allow direct deployment of a probabilistic network (e.g. a HMM).

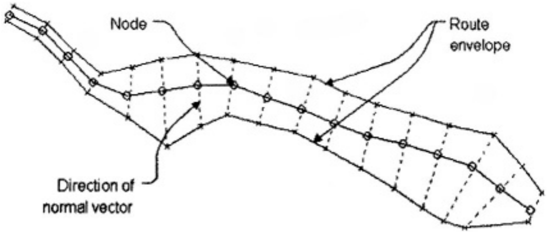

Figure 7.6. Spatial representation of the route model.

3.2 Learning algorithm

The input data of the algorithm is a set of trajectories, derived by a motion tracking algorithm that estimates the location of the centroid of the moving objects, from a single fixed camera. It is desirable to learn paths using representative trajectories unconstrained by tracking failure. For this reason, short trajectories or trajectories with many sudden changes of direction are filtered. Further validation of the trajectory dataset is based on knowledge of the entry/exit zones. Specifically, trajectories are accepted only if they start from a valid entry zone and terminate at a valid exit zone.

The model order of the scene routes is not defined explicitly, but it is determined implicitly by the algorithm parameters and the dataset. The first trajectory of the dataset initialises the first route model. Other route models will also be initialised automatically by trajectories that do not match an existing model.

Theoretically, there is no restriction in the number of route models of the scene. In practice, route models with very low usage are discarded through the learning process, for computational efficiency.

A summary of the learning algorithm is as follows:

1 The first trajectory of the dataset initialises the first route model

2 Each new trajectory is compared with the existing route models.

3 If a trajectory matches a route model, then the route model is updated.

4 If a trajectory does not match to any route model, a new model is initialised.

5 The updated route model is resampled, so inter-node distances are kept equal to R.

6 Each updated route model is compared with the other route models.

7 If two route models are sufficiently overlapped, they are merged.

The algorithm requires two parameters: a) the resample factor R which defines the level of detail for each route model. Very small values for the

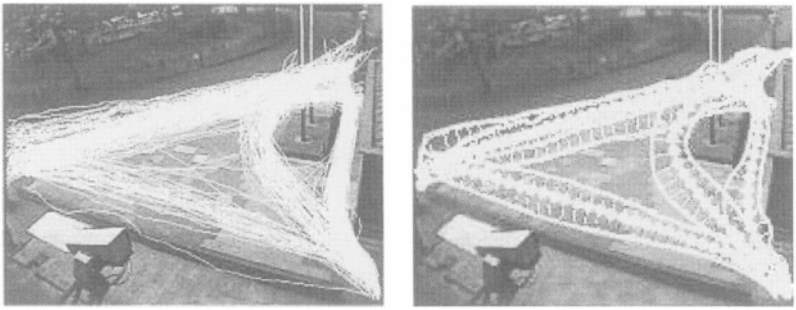

Figure 7.7. (a) Dataset of 752 trajectories, (b) Set of five route models, as derived by the route learning algorithm, for R=20 pix and T=30 pix.

resample factor are not recommended, because this selection can make the algorithm computationally expensive without significant benefit. b) a distance threshold T which defines the minimum allowable gap between different routes. Its recommended value is related to the quantity of learning data; specifically the less data the larger the value of T.

3.3 Segmentation to paths and junctions

Intuitively, a junction is the area where two routes cross. A more rigorous definition is adopted here: a junction is the region of intersection of two routes, where route directions differ by more than an angle ω. This definition reflects the fact that while a target is on a junction, some uncertainty is raised about its future direction.

Paths are considered as route parts between junctions and/or entry/exit zones. Paths may also relate to route overlapping. If route parts are overlapped and the route directions are similar along the overlapped route parts, their union represents a path. For instance, the upper path of Figure 7.8b is formed by two route parts with similar direction.

Accumulative statistics could be used to identify the areas where target directions are similar (paths) or different (junctions). However, from the above definitions, it can be concluded that junctions and paths are closely related to the geometry of the scene route models; therefore, they can be easily extracted from a set of route models, using computational geometry and constraints on the direction of the routes.

The route models of Figure 7.7 are visualised as polygons in Figure 7.8(a). Junctions are detected in regions where the direction of the routes differs by more than 5° and results are shown in Figure 7.8(b). Junctions are used to split the route models into paths. Figure 7.8b also visualises the segmentation of routes to paths and junctions.

The partitioning of routes to paths and junctions is performed not only to identify semantic features, but also to support activity analysis. For example,

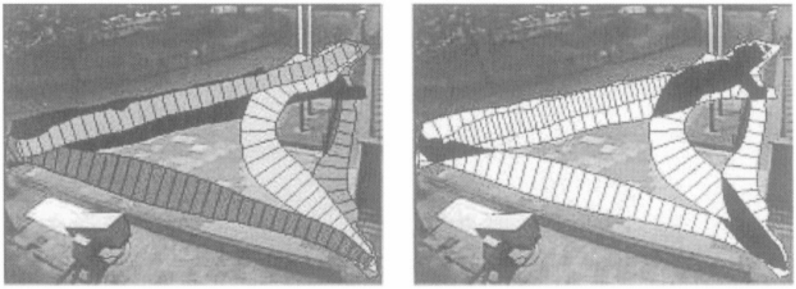

Figure 7.8. (a) Five routes detected by the route learning algorithm. (b) Segmentation of the scene route models to paths (white areas) and junctions (black areas).

entry/exit zones, paths and junctions can be considered as primitives of routes and rare, complicated routes can be described as sequences of these primitives. Also, junctions are regions of interest for a long-term prediction module, because a targets' motion within junctions reflects the targets' intention to move towards specific regions.

4. Activity analysis

Target activity can be related to elements of the scene model. For example, a target will enter the scene by an entry zone, follow a path, then reach a junction and take another path, stop at a stop zone and finally exit the scene through an exit zone. If it is detected outside the scene model, its activity will be characterised as atypical.

The discrete character of the scene model, as illustrated in a topological representation, allows discrete-state models like Markov Chains and Hidden Markov Models (HMM) to be applied. Both Markovian models can be used for activity modelling and analysis. The suggested approach is HMM-like for two reasons: firstly, HMMs can distinguish observations and states and model the uncertainty of correspondences of observations to states using membership functions. Secondly, the probabilistic nature of each of the scene elements allows the required membership functions to be easily determined for each of the states of the model.

Two network representations can be derived by the scene model. The first consists of scene elements like entry/exit zones, paths, junctions and stop zones. The second consists of all the nodes of all the scene route models. HMMs can be overlaid onto both types of network and can be used for activity analysis.

A Route-Based Hidden Markov Model (RBHMM) [6] describes a HMM that is superimposed on the set of all the nodes of the route models of a scene and is used to detect atypical activities. For instance, while Figure 7.9a depicts a common trajectory, Figures 7.9b-c depict trajectories that are atypical, according to the RBHMM model of the scene.

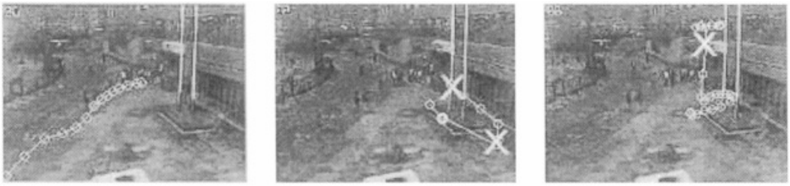

Figure 7.9. Three trajectories are shown. The left trajectory is a very common one. The middle one contains two rather atypical examples ('X' symbols). The right one is a very uncommon one of somebody 'climbing' (actually a tracking association error).

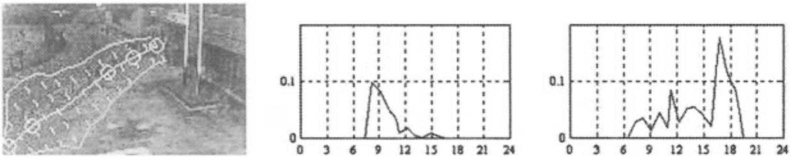

Figure 7.10. (a) Route model. (b) Probability that a pedestrian will move along the route, towards the entrance. (c) Probability that a pedestrian will move along the route, away the entrance. The probabilities are given for a 24-hour period. x-axis indicates the time of the day.

However, deciding whether an event is typical or atypical must involve information of the time at which it occurs. For instance while the trajectory of Figure 7.9 can be assumed as typical at 9am, it should be considered as atypical if it occurred at midnight.

Therefore, the RBHMM should be time-variant to reflect the variability of the expected activity over time. Figure 7.10 illustrates this variability of usage of a specific route. Figure 7.10b shows of the probability that a pedestrian will use the route to move towards the entrance, while Figure 7.10c shows the probability that a pedestrian will use the route to move away the entrance. At 8-9am, almost all trajectories occur towards the entrance of the University, whilst around 5pm, people tend to be leaving the building.

5. Integration of information from multiple views

Multi-camera surveillance systems cover wide-area scenes and aim to track targets within this scene. The key issue in these systems is to effectively integrate information from multiple cameras in order to provide complete histories of targets' activities within the environment, sampled by multiple cameras. To integrate information in the spatial domain, a world coordinate system is required. Usually, a ground plane coordinate system [16] is used which is consistent with the assumption that all the scene activity is coplanar. A ground plane map is used to illustrate results in ground plane coordinates. For example, Figure 6.12 of a later chapter illustrates a ground map constructed from geometric calibration models and views from six cameras. Four different field of views (FOV) are defined on the ground plane, to determine how the scene is viewed by a camera surveillance network:

- Visible FOV defines the regions that a single camera images, excluding occluded areas and obscured areas where activity cannot be interpreted.
- Camera FOV encompasses all the regions within the camera view, including occluded regions
- Network FOV is the union of all the visible FOVs of all the cameras of the network.
- Virtual FOV covers the network FOV and all the gaps in between the visible camera FOVs, within which targets may exist.

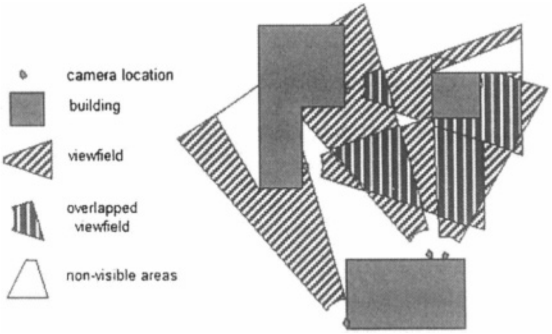

Figure 7.11. Visible FOVs of a network of cameras.

The visible FOVs of different cameras are related with different types of associations, depending on their spatial locations and how activity is seen. Specifically, two FOVs may be a) overlapped, i.e. they have common parts and a target may be seen simultaneously by the two cameras, or b) adjacent, i.e. they do not have common parts, but they are relatively close, such that targets exiting one FOV may enter the other FOV, c) distant, i.e. they are far apart and there is no direct relationship of the targets activities in these two views. This chapter uses the term “camera network topology” to represent the set of all the relationships of the cameras of the network.

Activity scene models can be applied to both the individual camera views and the common ground plane. Trajectory data has been converted to ground plane coordinates, and used as input to the entry/exit zones learning algorithm and the route learning algorithm. Results are illustrated in Figure 7.12.

Two issues are related to the ground plane approach of constructing integrated models for the entire covered scene. Firstly, the method requires explicit geometric calibration of all the cameras of the system. Secondly the model covers only the network FOV, failing to represent activity on the gaps of the virtual FOV. The next section introduces a technique that deals with these two issues.

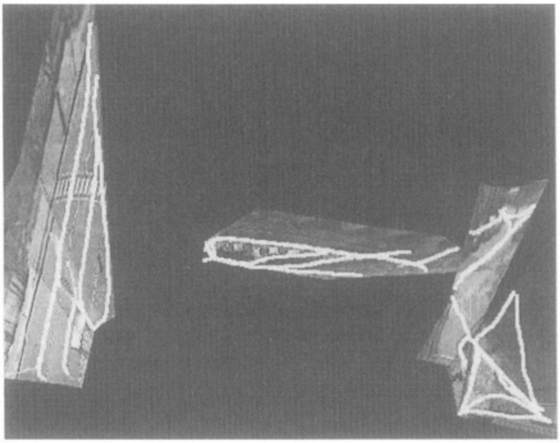

Figure 7.12. Routes learnt on a ground plane map.

5.1 Multiple Camera Activity Network (MCAN)

In the previous section, semantic scene models have been derived for individual camera FOVs and for the network FOV. However, it is desirable to extend the scene model to the entire virtual FOV, to cover activities that occur within the "blind" areas of the system. Although these activities are not directly viewed by the system, reasonable assumptions are derived.

Lets first assume that no geometric information exists that allows localisation of the camera FOVs. Cameras with overlapped visible FOVs can be identified [1] [15] and calibrated using a homography. A common semantic scene model can be derived for each set of cameras with overlapped FOVs. However, possible gaps between cameras do not allow the semantic scene model to be established for the whole system, and isolated scene models are derived instead [14].

To overcome the lack of a common scene model for the virtual FOV, the isolated scene models must be linked. However, no spatial linking is achievable, due to the lack of geometric calibration. Instead, a probabilistic-temporal linking of the isolated views is proposed.

All the entry/exit zones of the camera FOVs are represented collectively as a network of nodes.The links of the network represent transitions between the entry/exit zones, either visible (through the Network FOV), or invisible (through the "blind" areas). A markovian chain or a HMM can be overlaid on the topology representation, to create a probabilistic framework for activity analysis and long-term predictions.

Visible links are learnt by the route learning algorithm using trajectories derived by a single-camera tracker or overlapped multiple camera tracker and are physically represented in spatial terms, according to the route model

However, the challenge is to identify the invisible links and this is the focus of the method proposed in this section. Invisible links are estimated in temporal terms and more specifically by pdfs that show the distribution of the target transition periods through the blind areas.

Theoretical formulation. A MCAN is formulated by the set ofall entry/exit nodes that are detected within all the cameras of the system. No information is provided regarding the spatial relationship of these nodes. It is required to identify the directional links ofthis network expressed in probabilistic-temporal terms, which represent target transitions from one node to the other.

A graph model (shown in Figure 7.13) is used to represent a possible link between two nodes, i and j. Targets disappear from the node i with rate $n_i(t)$ and appear at the node j with rate $m_j(t)$. A third virtual node k represents everything out of the nodes i and j. Targets transit from node i to node j in time τ withprobability $\alpha_{ij}(\tau)$, otherwise they transit to the virtual node k with probability α_{ik}.

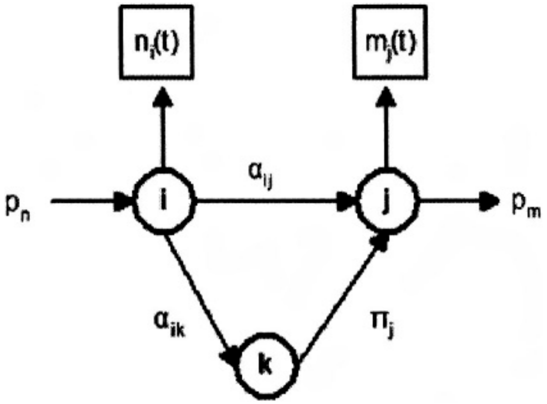

Figure 7.13. Graph model showing the probabilistic links between two nodes (i and j) and the virtual node k.

Transition probabilities fulfill the equation:

$$\alpha_{ij}(\tau)d\tau + \alpha_{ik} = 1 \tag{7.3}$$

Also, targets from the node k move to node j with rate π_j; i.e. new targets can be "generated" at node k, and are detected on entering at node j.

The surveillance system is able to observe the signals $n_i(t)$ and $m_j(t)$. The two signals are assumed individually and jointly stationary. Therefore, the cross-correlation function is defined:

$$R_{ij}(\tau) = E[n_i(t)m_j(t+\tau)] \tag{7.4}$$

If it is assumed that the two signal $n_i(t), m_j(t)$ are taken digital values from the set 0, 1, then $p_n = pn_i(t) = 1 = En_i(t)$ and $p_m = pm_j(t) = 1 = Em_j(t)$. The cross-correlation $R_{ij}(\tau)$ and the covariance $C_{ij}(\tau)$ defined as:

$$C_{ij}(\tau) = R_{ij}(\tau) - p_n p_m \tag{7.5}$$

and are used to identify possible links. If

$$C_{ij}(\tau)d\tau = 0 \tag{7.6}$$

then the two signals are uncorrelated and, because according to the proposed graph model, their relationship can be only linear, they are independent [12] and no real link should exist between them. Otherwise, the two signals are dependent and a valid link i®j must exist. In this case, the transition probability is estimated by the formula:

$$\alpha_{ij}(\tau) = \frac{C_{ij}}{p_n(1-p_n)} \tag{7.7}$$

Summarising, the MCAN is defined by the nodes, expressed as Gaussian distributions on the separate camera views and directional links between nodes, defined by transition probabilities that depend on the transition time.

The topology of the camera views is determined by the set of valid links and their transition times. If a link is detected between the zones of two cameras, the two cameras are either adjacent or overlapped. If the transition time between the exit zone i and the entry zone j is approximately zero, then the two zones of the two cameras are overlapped. If the transition time is positive, then the targets move from one zone to the other through an invisible path. Finally, if the transition time is negative, then the targets move from one zone to the other through a path that is partially or entirely visible by the two cameras.

6. Database

The surveillance database supports several activities ranging from real-time storage of tracking data to allowing a user to recall certain types of object activity. In the previous chapter on distributed multi-view tracking we described how these requirements were implemented using a hierarchical database structure [2]. The image framelet layer stores the video associated with detected objects, and the object motion layer contains the tracking data of each object observed by the system. In this chapter we describe the layers of the database that facilitate the reporting requirements of the database.

Semantic Description Layer. The object motion layer provides input to a machine-learning algorithm that automatically learns a semantic scene model, which contains both spatial and probabilistic information. Regions of activity

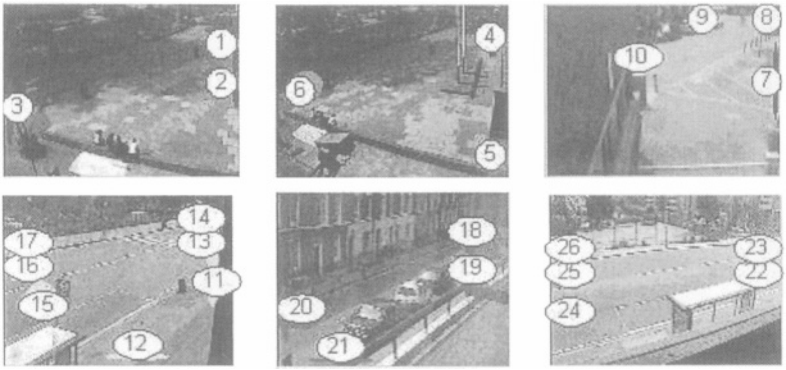

Figure 7.14. The detected entry/exit zones for the six cameras of the network. The zones are numbered as nodes of the activity network.

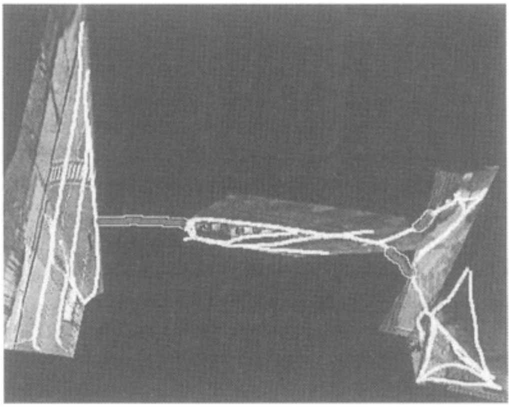

Figure 7.15. Invisible links (gray lines) detected using the MCAN and visible routes (white lines) detected using the route learning algorithm, as shown in Figure 7.12.

can be labelled in each camera view, for example entry/exit zones, paths, routes and junctions. These models can also be projected on the ground plane as is illustrated in Figure 7.15. These paths were constructed by using 3D object trajectories stored in the object motion layer. The gray lines represent the hidden paths between cameras. These are automatically defined by linking entry and exit regions between adjacent non-overlapping camera views. These semantic models enable high-level queries to be submitted to the database in order to detect various types of object activity. For example we can generate spatial queries to identify any objects that have followed a specific path between an entry and exit zone in the scene model. This allows any object trajectory to be compactly expressed in terms of a routes and paths stored in the semantic description layer.

Table 7.1. Attributes stored in semantic description layer (entry/exit zones).

Field Name	*Description*
Camera	The camera view of the entry or exit zone
Zoneid	The identification of the entry or exit zone
Position	The 2D centroid of the entry or exit zone
Cov	The covariance of the entry or exit zone
Poly_zone	A polygonal approximation of the entry or exit zone

Table 7.2. Attributes stored in semantic description layer (routes).

Field Name	*Description*
Camera	The camera view of the route
Routeid	The identification of the route
Nodes	The number of nodes in the route
Poly_zone	A polygonal approximation of the envelope of the route

Table 7.3. Attributes stored in semantic description layer (route nodes).

Field Name	*Description*
Camera	The camera view of the route node
Routeid	The identification of the route
Nodeid	The identification of the route node
Position	The central 2D position of route node
Position_left	The left 2D position of the route node
Position_right	The right 2D position of the route node
Stddev	σ Gaussian distribution of object trajectories observed at the route node
Poly_zone	Polygon representation of region between this route node and its successor

The main properties stored in the semantic description layer are described in Table 7.1, Table 7.2 and Table 7.3. Each entry and exit zone is approximated by a polygon that represents the ellipse of the region. Using this internal representation in the database simplifies spatial queries to determine when an object enters an entry or exit zone. The polygonal representation is also used to approximate the envelope of each route and route node, which reduces the complexity of the queries required for online route classification that will be demonstrated in the next section. An example of the routes, routenodes, entry and entry regions is shown in Figure 7.16. The black and white ellipses indicate entry and exit zones, respectively. Each route is represented by a sequence of nodes, where the black lines represent the main axis of each route, and the white lines define the envelope of each route.

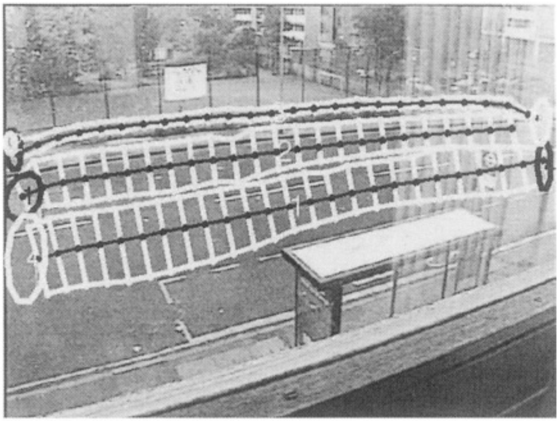

Figure 7.16. Example of routes, entry and exit zones stored in semantic description layer.

6.1 Metadata Generation

Table 7.4. Metadata generated (object_summary).

Field Name	*Description*
Videoseq	The identification of the capture video sequence in the image framelet layer
Trackid	The trackid of the object
EntryTime	The time when the object was first detected
ExitTime	The time when the object was last seen
EntryPosition	The 2D entry position of the object
ExitPosition	The 2D exit position of the object
Path	A sequence of points used to represent the object's 2D trajectory
Appearance	The average normalized colour components of the tracked object

Metadata is data that describes data. The multi-layered database allows the video content to be annotated using an abstract representation. The key benefit of the metadata is that it can be more efficiently queried for high-level activity queries when compared to the low level data. It is possible to generate metadata online when detected objects are stored in the image framelet and object motion layers. Initially, the video data and object trajectory is stored in the image framelet and object motion layers. The object motion history is then expressed in terms of the model stored in the semantic description layer to produce a high-level compact summary of the object's history. The metadata contains information for each detected object including: entry point, exit point,

Table 7.5. Metadata generated (object_history).

Field Name	*Description*
Videoseq	The identification of the captured video sequence in the image framelet layer
Trackid	The trackid of the object
Routeid	The identification of the route
EntryTime	The time the object entered the route
Entrynode	The first node the object entered along the route
EndTime	The time the object left the route
ExitNode	The last node the object entered along the route

time of activity, appearance features, and the route taken through the FOV. This information is tagged to each object detected by the system. The key properties of the generated metadata are summarised in Table 7.4 and Table 7.5. Each tracked object trajectory is represented internally in the database as a path geometric primitive, which facilitates online route classification.

Visual Queries. One application of the surveillance database is to support object activity related queries. The data stored in the surveillance database provides training data for machine learning processes that learn spatial probabilistic activity models in each camera view. By integrating this information with tracking data in the surveillance database it is possible to automatically annotate object trajectories. The image framelet layer of the database contains the low-level object pixel data that was detected as a moving object by the single view camera. This layer is used to support video playback of object activity at various time intervals. The second layer is comprised of the object tracking data that is captured by the single view tracking performed by each intelligent camera. The data consists of the tracked features of each object detected by the system. The tracked features stored as a result of single view tracking include: bounding box dimensions, object centroid, and the normalised colour components. Data is extracted from the object motion layer in order to learn spatial probabilistic models that can be used to analyse object activity in the scene. The semantic description of the scene allows the information in the object motion layer to be expressed in terms of high-level meta-data that can support various types of activity based queries. The query response times are reduced from several minutes to only a few seconds.

An example of the results returned by an activity query is shown in Figure 7.17. The semantic description of the scene includes all the major entry and exit regions identified by the learning process. The major entry and exit regions are labelled on each image. The first example in Figure 7.17a shows a sample of pedestrians moving between entry region B and exit region A. The second example in Figure 7.17b shows a sample of pedestrians moving between entry region C to exit region B. The activity based queries are run using the meta-data layer of the database, resulting in considerable savings in terms of execution time, compared to using the object motion, or image framelet layers.

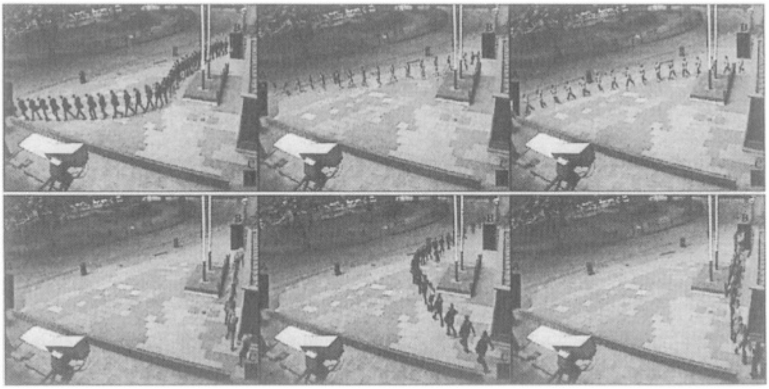

Figure 7.17. Visualisation of results returned by spatial temporal activity queries.

Figure 7.18 illustrates how the database is used to perform route classification for two of the tracked object trajectories. Four routes are shown that are stored in the semantic description layer of the database in Figure 7.18. In this instance the first object trajectory is assigned to route 4, since this is the route with the largest number of intersecting nodes. The second object trajectory is assigned to route 1. The corresponding SQL query used to classify routes is shown below. Each node along the route is modeled as a polygon primitive provided by the PostgreSQL database engine. The query counts the number of route nodes with which the object's trajectory intersects. This allows a level of discrimination between ambiguous choices for route classification. The '?#' operator in the SQL statement is a logical operator that returns true if the object trajectory intersects with the polygon region of a route node. Additional processing of the query results allows the system to generate the history of the tracked object in terms of the route models stored in the semantic description layer. A summary of this information generated for the two displayed trajectories is given in Table 7.6. It should be noted that if a tracked object traversed multiple routes during its lifetime then several entries would be created for each route visited.

```
select routeid, count(nodeid)
from routenodes r, objects o
where camera=2
and o.trajectory ?# r.polyzone
and o.videoseq =87
and o.trackid =1
```

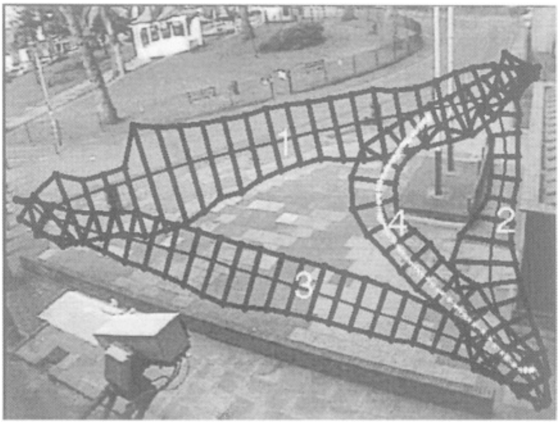

Figure 7.18. Example of online route classification.

Table 7.6. **Results returned by the SQL query to the database.**

Videoseq	*Trackid*	*Start Time*	*End Time*	*Route*
87	1	08:16:16	08:16:27	4
87	3	08:16:31	08:16:53	1

7. Summary

This chapter described a methodology for learning activity-based semantic scene models from observing activity, in the application area of automatic visual surveillance. More specifically, a semantic scene model was introduced, consisting of features like entry/exit zones, stop zones, routes, paths and junctions.se The semantic labels are learnt by unsupervised algorithms that exploit the vast amount of motion observations that can be gathered by surveillance systems.

Entry/exit zones and stop zones are semantic labels associated to single-point events. These regions are modelled by GMMs and learnt by an EM-based algorithm. Routes, paths and junctions are semantic labels assocaited to sequence-of-point events (trajectories). Route models can be learnt automatically from a set of trajectories. Then, the routes are segmented to paths and junctions using computational geometry.

The activity of a scene can analysed using a Route-Based Hidden Markov Model (RBHMM) that is a HMM superimposed on the scene routes. RBHMM are proposed to be time-variant so they can encode the variability of the activity with respect to the time of the day.

Scene models from multiple camera views are intergrated using the Multiple-Camera Activity Network (MCAN). This model allows tempo-probabilistic linking of camera views and does not require any manual camera calibration. Instead, the network is learnt though an automatic correlation-based algorithm.

The methodology that was described in this aimed to provide surveillance systems with a high-level knowledge of their environments. Also, it was inspired by the idea of autonomous surveillance systems that can be self-calibrated and can adapt to changes of their environments.

The database stores several different representations of the tracking data, which supports spatial-temporal queries at the highest level, to the playback of video data at the lowest level. In the earlier chapter on object tracking the image framelet and object motion layers of the database were discussed. In this chapter the semantic description layer of the database was described along with its applications for performing online route classification and metadata generation. The advantage of using a hierarchical database is that the metadata can be utilized to give much faster response times to various object activity queries than would be possible when querying the original tracking data.

Acknowledgments

This work was partly undertaken with support from the Engineering and Physical Science Research Council (EPSRC) under grant number GR/M58030.

References

[1] James Black, Tim Ellis, “Multi Camera Image Tracking”, Second International Workshop on Performance Evaluation of Tracking and Surveillance, PETS2001, Kauai, Hawaii, December 2001.

[2] James Black, Tim Ellis, Dimitrios Makris, "A Hierarchical Database for Visual Surveillance Applications", IEEE International Conference on Multimedia and Expo, ICME2004, Taipei, Taiwan, June 2004.

[3] Dempster, N. Laird, D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", Journal of the Royal Statistical Society, B-39, pp.1-38, 1977.

[4] Tim Ellis, Dimitrios Makris, James Black, "Learning a Multicamera Topology", Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, VS-PETS2003, pp. 165-171, Nice, France, October 2003.

[5] Dimitrios Makris, Tim Ellis, "Finding Paths in Video Sequences", British Machine Vision Conference, BMVC2001, pp.263-272, Manchester, UK, September 2001.

[6] Dimitrios Makris, Tim Ellis. "Spatial and Probabilistic Modelling of Pedestrian Behaviour", British Machine Vision Conference, BMVC2002, pp.557-566, Cardiff, UK, September 2002.

[7] Dimitrios Makris, Tim Ellis, "Path Detection in Video Surveillance", Image and Vision Computing, vol.20(12), pp.895-903, October 2002.

[8] Dimitrios Makris, Tim Ellis, "Automatic Learning of an Activity-Based Semantic Scene Model", IEEE Int. Conf. on Advanced Video and Signal Based Surveillance, AVSS2003, pp. 183-188, Miami, FL, July 2003.

[9] Dimitrios Makris, Tim Ellis, James Black, "Learning Scene Semantics", Early Cognitive Vision Workshop", ECOVISION 2004, Isle of Skye, May 2004.

[10] Dimitrios Makris, Tim Ellis, James Black, "Bridging the Gaps between Cameras", IEEE Conference on Computer Vision and Pattern Recognition, CVPR2004, Washington DC, USA, June 2004.

[11] Dimitrios Makris, "Learning an Activity-Based Semantic Scene Model", PhD Thesis, City University, London, 2004.

[12] Athanasios Papoulis, "Probability, Random Variables and Stochastic Processes", Third Edition, McGraw-Hill, 1991.

[13] 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.

[14] Stauffer, K. Tieu, "Automated multi-camera planar tracking correspondence modelling". IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2003, vol.1, pp 259-266, Madison Wisconsin, June 2003.

[15] G.P. Stein, "Tracking from Multiple View Points: Self-calibration of Space and Time", Image Understanding Workshop, Montery, CA, November 1998.

[16] T. N. Tan, G. D. Sullivan, K.D. Baker, "Recognising Objects on the Ground-plane", Image and Vision Computing, vol.12, pp. 164-172, 1994.