International Journal of Computer Vision 67(1), 21–51, 2006

DOI: 10.1007/s11263-006-4329-6

Beyond Tracking: Modelling Activity and Understanding Behaviour

TAO XIANG AND SHAOGANG GONG
Department of Computer Science, Queen Mary, University of London, E1 4NS, UK
txiang@dcs.qmul.ac.uk
sgg@dcs.qmul.ac.uk

Received May 11, 2004; Revised June 28, 2005; Accepted July 27, 2005

First online version published in February, 2006

Abstract. In this work, we present a unified bottom-up and top-down automatic model selection based approach for modelling complex activities of multiple objects in cluttered scenes. An activity of multiple objects is represented based on discrete scene events and their behaviours are modelled by reasoning about the temporal and causal correlations among different events. This is significantly different from the majority of the existing techniques that are centred on object tracking followed by trajectory matching. In our approach, object-independent events are detected and classified by unsupervised clustering using Expectation-Maximisation (EM) and classified using automatic model selection based on Schwarz's Bayesian Information Criterion (BIC). Dynamic Probabilistic Networks (DPNs) are formulated for modelling the temporal and causal correlations among discrete events for robust and holistic scene-level behaviour interpretation. In particular, we developed a Dynamically Multi-Linked Hidden Markov Model (DML-HMM) based on the discovery of salient dynamic interlinks among multiple temporal processes corresponding to multiple event classes. A DML-HMM is built using BIC based factorisation resulting in its topology being intrinsically determined by the underlying causality and temporal order among events. Extensive experiments are conducted on modelling activities captured in different indoor and outdoor scenes. Our experimental results demonstrate that the performance of a DML-HMM on modelling group activities in a noisy and cluttered scene is superior compared to those of other comparable dynamic probabilistic networks including a Multi-Observation Hidden Markov Model (MOHMM), a Parallel Hidden Markov Model (PaHMM) and a Coupled Hidden Markov Model (CHMM).

Keywords: dynamic scene modelling, graph models, discrete event recognition, activity representation, behaviour recognition, dynamic probabilistic networks, Bayesian model selection

1. Problem Statement

Over the past decade, numerous efforts have been made to model actions and activities captured in video (Xiang and Gong, 2003; Xiang et al., 2002; Gong and Buxton, 1992; Haritaoglu et al., 2000; Intille et al., 1997; McKenna et al., 2000; Stauffer and Grimson, 2000; Wada and Matsuyama, 2000; Johnson et al., 1998; Brand et al., 1996; Oliver et al., 2000; Babaguchi et al., 2002; Hongeng and Nevatia, 2001; Gong and Xiang, 2003; Bregler, 1997; Pavlovic et al., 1999). An action is commonly referred to as a sequence of movements executed by a single object and it often has a statistical nature. Action modelling is usually concerned with analysing the statistical sequential properties of movements of human body or body parts such as people 'sitting' and 'walking' (Rao et al., 2002; Bobick and Davis, 2001; Gong et al., 1999; Bregler, 1997; Povlovic et al., 1999). An activity on the other hand is a larger-scale 'scene' most likely involving multiple objects interacting or co-existing in a shared common space (Johnson et al., 1998; Gong and Xiang, 2003; Oliver

et al., 2000). Examples of activities include 'people playing football' and 'shoppers checking out at a supermarket'. Activity modelling is thus concerned with not only modelling actions executed by different objects in isolation, but also the interactions and causal relationships among these actions. The goal of activity modelling is to understand behaviour. Behaviour is the meaning of activity given as a semantic description extracted through activity modelling. Modelling activity and understanding behaviour are fundamental for human computer interaction, visual surveillance, and video content analysis for indexing.

In this paper, we develop a unified Bayesian Information Criterion (BIC) based framework for modelling activities of multiple objects where the behaviour of each different object is constantly constrained and affected by those of others. Further, their activities are largely correlated either explicitly or implicitly in space and over time. A model that can capture accurately activities of multiple objects needs to take into account the uncertainty and variability in the behaviours exhibited by individual objects in different scenarios. This suggests a learning based approach where the structure and parameters of the model can be learned from data with little human intervention and without over-rigid assumptions. To this end, we propose to build a data-driven probabilistic model based on unsupervised learning. This approach aims to provide a unified bottom-up and top-down Bayesian solution to the activity modelling problem. In particular, we address the following issues:

1. How to select visual features that best represent activities of multiple objects.
2. How to model the temporal and causal correlations among those objects that are considered to form meaningful activities.
3. How to learn both the structure and parameters of an activity model from data.
4. How to infer semantic description of activities from the learned model.
5. How to use an activity model to improve the interpretation of the behaviour of each individual object.

1.1. Activity Representation: From Continuous Trajectory to Discrete Event

Previous approaches to both action and activity modelling have been mostly relied upon segmentation and tracking of objects in the scene (Gong and Buxton, 1992; Haritaoglu et al., 2000; Intille et al., 1997; McKenna et al., 2001; Stauffer and Grimson, 2000; Wada and Matsuyama, 2000; Johnson et al., 1998; Rao et al., 2002). This is due to the fact that actions and activities have traditionally been considered to be discriminable by the trajectories of object motion, modelled either statically as templates or dynamically as state machines. However, there are a number of fundamental limitations due to the following assumptions made either implicitly or explicitly about a scene by a trajectory based approach:

1. Sufficient details about the objects of interest are available in videos of high fidelity which allow for elaborative object models to be built using local image features. This is not feasible with video footages captured from cluttered scenes dominated by busy activities, for example in the case of CCTV surveillance of public space. Such footages are characterised by low resolution, drastic lighting change, and cluttered background containing non-stationary objects.
2. Objects can be tracked consistently in space and over time. This assumption is often invalid in busy scenes involving movements of multiple objects with frequent overlapping motion patterns resulting in brickle and often discontinuous object trajectories and inconsistent labelling, despite various methods been proposed to cope with occlusions and lighting changes (Haritaoglu et al., 2000; McKenna et al., 2000). For example, in an overcast day with smooth and moderate traffic volume, the moving vehicles on a motorway can be tracked successfully (see Fig. 1(a)). Their activities can thus been modelled based on the established trajectories. Now consider an aircraft docking scenario where aircraft arrival is followed by the activities of various ground service vehicles (see Fig. 1(b)). The movements of different objects are heavily overlapped and highly discontinuous. As a result, a large amount of fragmental, noisy trajectories are obtained which makes the trajectory based activity modelling infeasible.
3. Trajectory alone captures all the information about object behaviours. This is not true in many cases. An example shown in Fig. 1(c) illustrates a shopping scenario where shopper can either take a can of drink and pay for it or just browse and leave. In this scenario, very similar trajectories can be generated by activities of significantly different meanings, i.e.

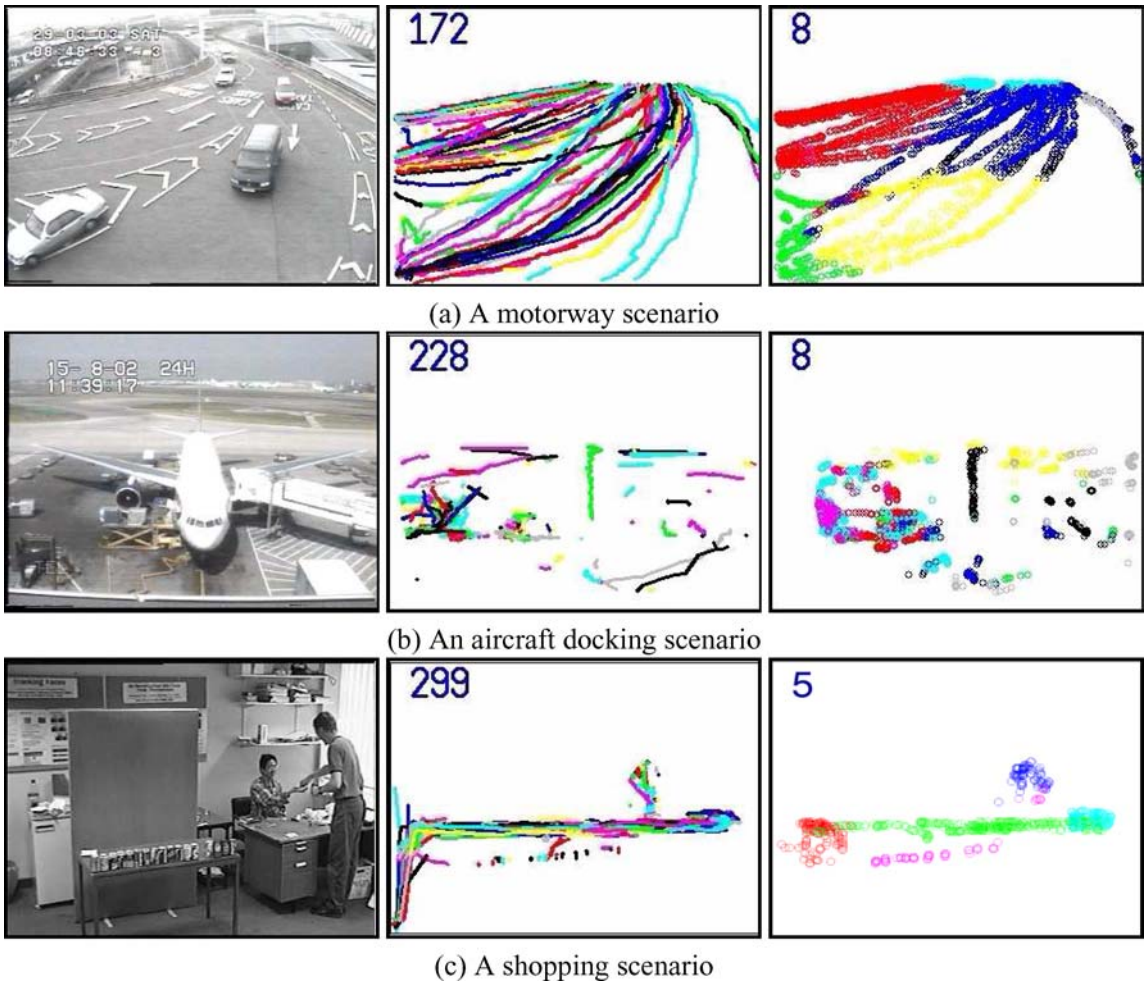


Figure 1. A shopping scenario Comparing trajectory and event based activity representation. *Left*: Example frames from three different scenes. *Middle*: Trajectories computed over time where different trajectories are illustrated in different colours with the total number of trajectories accumulated so far in each scene shown at the top-left corner of the frame. *Right*: Labelled scene events detected over time where 8 class of events have been detected both in the motorway and aircraft docking scenes and 5 classes of events have been detected in the shopping scene. Events of different classes are illustrated using different colours with the total number of event classes shown at the top-left corner of the frame.

trajectory information alone is insufficient for discriminating/recognising different activities.

More recently, several attempts have been made to circumvent the problems intrinsic to the trajectory based approach to activity modelling. Instead of computing trajectories through object tracking, these methods focus on correlations of discrete semantic events at the pixel level through learning (Chomat et al., 2000; Gong et al., 2002; Sherrah and Gong, 2001). However, these purely pixel-level based approaches can be sensitive to noise due to ignoring any spatial correlations among salient pixel changes. They can also be computationally expensive due to the large number of pixel events to be monitored simultaneously.

To address this problem, we exploit a bottom-up approach for modelling object-independent discrete scene events using Pixel Change History (PCH) and Schwarz's Bayesian Information Criterion (BIC) (Schwarz, 1978) based unsupervised clustering and classification. This enables us to represent activity without the need for object tracking and trajectory matching. An event is defined as a group of significant pixel changes in a local image neighbourhood *over time*. Events are detected and classified by unsupervised clustering using Gaussian Mixture Model

(GMM) with automatic model selection based on BIC. An activity is thus represented as a group of co-occurring events and modelled through interpretation of the temporal and causal correlations among different classes of events. The right hand side frames in Fig.1(a), (b) and (c) show activities represented by discrete scene events detected and classified in the three example scenarios. Activities are characterised by the temporal order of different events and the correlations among them. This event based representation is more simplistic and robust compared to the trajectory based representation. It is thus more suitable for complex activity modelling.

It is important to emphasise that our definition of event corresponds to scene level visual changes and is object independent. Although events are detected in each image frame, each event is represented by and estimated based on accumulated visual changes over time, i.e., pixel change histories. This is different from most other definitions of event in the literature (Babaguchi et al., 2002; Hongeng and Nevatia, 2001; Medioni et al., 2001; Rao et al., 2002) which are closer to the concept of 'atomic action'. Our definition of event is similar in spirit to that of Wada and Matsuyama (2000).

It is also worth pointing out that no matter how sophisticated a event detection and recognition algorithm is, errors are inevitable. We believe that this problem cannot be solved solely based on the visual information available in individual frames. It can only be tackled by using information accumulated over time collectively. This is the main motivation and essence of the approach adopted in the paper. The activity model proposed in this paper does not rely on a perfect event detection and classification algorithm. Instead, we proposed to use Dynamic Probabilistic Networks (DPNs) to reason about the temporal and causal correlations among events. Our activity model is thus robust to the errors in event detection and recognition results.

1.2. Activity Modelling: Discovering the Structure of Dynamic Correlations

Represented as discrete events of multiple classes, an activity can be modelled as a set of structured states in a state space using a probabilistic dynamic graph. The states are correlated by a set of causal or/and temporal connections referred to as the structure of the model. The model requires both the determination of the states, often through unsupervised clustering of a training dataset, and the discovery of the underlying structure performed by the factorisation of the state space given the training dataset.

Probabilistic graph models have received enormous attention in recent years for modelling and recognising activities captured in video, ranging from visual surveillance, gesture recognition, visually mediated human-computer interaction, sport analysis to virtual character synthesis (Gong and Buxton, 1992; Buxton and Gong, 1995; Bobick and Wilson, 1997; Intille and Bobick, 1998; Johnson et al., 1998; Gong et al., 1999; Brand and Kettnaker, 2000; Stauffer and Grimson, 2000; Oliver et al., 2000; Sherrah and Gong, 2000; Vogler and Metaxas, 2001; Gong and Xiang, 2003). These include both temporal sequential models such as Hidden Markov Models (HMMs) and static causal models such as Bayesian Belief Networks (BBNs). However, both conventional BBNs and HMMs are unsuitable for modelling activities involving multiple objects/people either in interaction or as a group. If we consider the actions of each individual object are intrinsically governed by a temporal process, the activity of multiple objects are underpinned by not only temporal but also causal correlations among multiple temporal processes. Despite that BBNs have been shown to be capable of reasoning about the behaviours of object activities, they are limited to modelling static causal relationships without taking into consideration the temporal ordering (Buxton and Gong, 1995; Intille and Bobick, 1998). This is only applicable for well structured activities with clear causal semantics. For modelling less structured group or interactive activities involving multiple temporal processes, Dynamic Probabilistic Networks (DPNs) are required (Ghahramani, 1998; Heckerman, 1995).

Hidden Markov Models (HMMs) are perhaps the most commonly used DPNs. A standard HMM has only one hidden state node and one observation node at each time instance modelling a single temporal process. This often results in the high dimensionality of both the state space and observation space. A HMM thus requires a large number of parameters to describe if it is to model multiple temporal processes simultaneously. This implies that a single state or observation variable is to represent implicitly multiple sources of variations at any given time instance. Unless the training data set is very large and relatively 'clean,' poor model learning is expected. To address this problem, various topological extensions to the standard HMMs can be considered to factorise explicitly the

observation and/or state space by introducing multiple hidden state variables and multiple observation variables for modelling different temporal processes explicitly and simultaneously. For example, a Multi-Observation-Mixture+Counter Hidden Markov Model (MOMC-HMM) was introduced by Brand and Kettnaker (2000) to factorise the observation space. Other extensions have been proposed to factorise both the state and observation space. Vogler and Metaxas (2001) proposed Parallel Hidden Markov Models (PaHMMs) that factorise the state space into multiple independent temporal processes without causal connections in-between. Clearly this assumption of different temporal processes being independent of each other is invalid in most cases, especially when dealing with group or interactive activities. Brand et al. (1996) and Oliver et al. (2000) exploited Coupled Hidden Markov Models (CHMMs) to take into account the temporal and causal correlations among hidden state variables. They are essentially fully coupled pairs of HMMs such that each state is conditionally dependent on all past states of all processes at the previous time instance. However, it can be shown that such a fully connected state space cannot be factorised effectively. A CHMM is thus sensitive to observation noise especially when the number of temporal processes are large (Ghahramani, 1998).

In this paper, we develop a Dynamically Multi-Linked Hidden Markov Model (DML-HMM) for modelling group activities represented as multiple classes of discrete events, with its topology being discovered automatically using Schwarz's Bayesian Information Criterion (BIC) based factorisation. The number of temporal processes in a DML-HMM is given by the number of classes of events detected using unsupervised clustering and classification. Thus both the structure and parameters of a DML-HMM are learned from data in an unsupervised manner. The two key advantages of using DML-HMM for activity modelling is its unsupervised nature which avoids the tedious hand labelling of data and its data-driven topology learning which avoids the often unreliable hand crafting of the model structure.

It is worth pointing out that the Gaussian Mixture Model (GMM) used for event detection and classification can also be considered as a probabilistic graph model (Ghahramani, 1998). In the meantime, the model selection criterion we formulate for both GMM in event detection and classification and DML-HMM for activity structure discovery is a Gaussian approximation of Bayesian Model Selection (Kass and Raftery, 1995). Therefore, our approach provides a unified Bayesian treatment to the activity modelling problem.

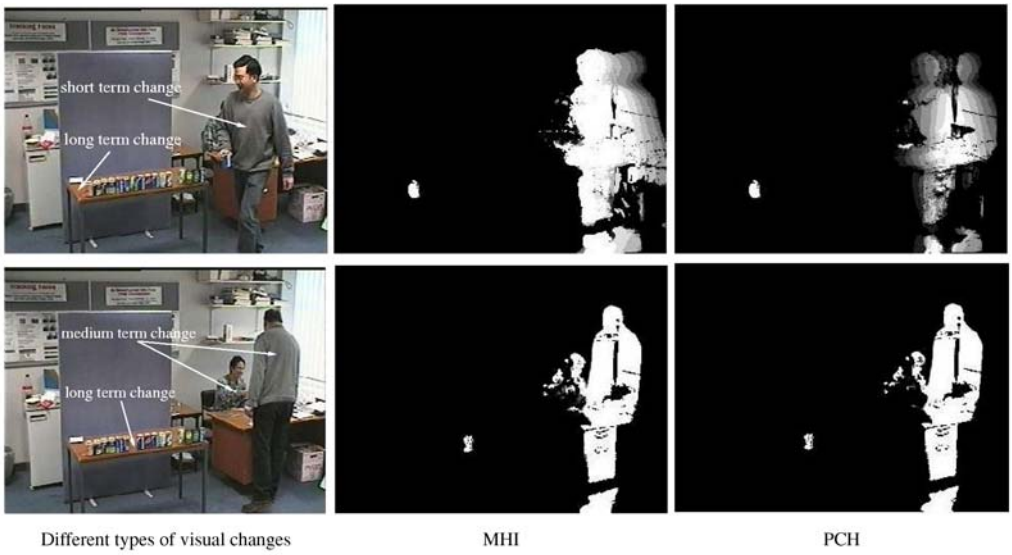
The remaining of this paper is organised as follows. Section 2 addresses the problem of activity representation based on discrete events that are automatically detected and classified using unsupervised clustering. Automatic model selection is also addressed. Section 4 centres on the development of a suitable dynamic probabilistic network for activity modelling involving multiple objects. Our approach is further illustrated in Section 4 using two examples of activity modelling and behaviour understanding in an indoor and outdoor scenes. We show comparative experimental results on activity modelling using different dynamic probabilistic networks before we conclude in Section 5. A derivation of BIC as Gaussian approximation of Bayesian Model Selection is given in Appendix A.

2. Event Recognition

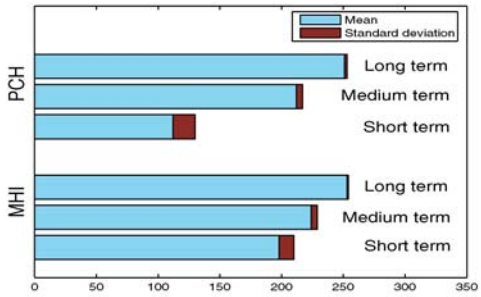
We define events as significant scene changes characterised by the location, shape and direction of the changes. They are object-independent and location specific as illustrated by the right hand side frames in Fig. 1(a), (b) and (c). We also consider that these events are autonomous, meaning that both the number of these events and their whereabout in a scene are determined automatically bottom-up without top-down manual labelling using predefined hypotheses.

2.1. Seeding Event: Measuring Pixel Change History

Adaptive mixture background models are commonly used to memorise and maintain the background pixel distribution of a dynamic scene (McKenna et al., 2000; Ng and Gong, 2001; Stauffer and Grimson, 2000). The major strength of such a model is its potential to cope with persistent movements of background objects such as waving tree leaves given appropriate model parameter setting. However, an adaptive mixture background model cannot differentiate, although may still be able to detect the presence of, pixel-level changes of different temporal scales. In general, a pixel-level change of different temporal scales can have different significance in its semantics:



(a) Examples of MHI and PCH images.



(b) The mean and standard deviation values of MHI and PCH for different types of visual changes.

Figure 2. Comparing MHI with PCH. (a) Examples of MHI and PCH images from a shopping sequence (see Section 4.1.1 for details about the sequence). The parameters of PCH were: $\varsigma = 12$ and $\tau = 10$. The decay factor τ was set to \$10\$ for computing MHI. For each of the first 1000 frames of the shopping sequence, pixels with non-zero MHI and PCH values were manually labelled into three classes according to the types of visual changes (i.e. short, medium or long term changes). The mean and standard deviation of MHI and PCH for different classes are plotted in (b). It can be seen from (b) that PCH exhibits much greater discriminative power compared to MHI in differentiating short term from medium and long term changes. This is also shown quantitively in Table 1.

1. A short term change is most likely to be caused by instant moving objects (e.g. passing-by people or vehicles).
2. A medium term change is most likely to be caused by the localised moving objects (e.g. a group of people standing and talking to each other).
3. A long term change is most likely to be caused by either the introduction of novel static objects into the scene, or the removal of existing objects from the scene (e.g a piece of furniture is moved in the background or a car is parked in a carpark).

Examples of visual changes of different temporal scales are shown in Fig. 2(a).

We seek a single, unified multi-scale temporal representation that can capture and differentiate changes of such different rates/scales at the pixel level. Temporal wavelets were adopted for such a multi-scale analysis (Sherrah and Gong, 2001). However, the computational cost for multi-scale temporal wavelets at the pixel level is very expensive. They are therefore unsuitable for real-time performance. Alternatively, Motion History Image (MHI) is less expensive to compute by

keeping a history of temporal changes at each pixel location which then decays over time. MHI has been used to build holistic motion templates for the recognition of human movements (Bobick and Davis, 2001) and moving object tracking (Piater and Crowley, 2001). An advantage of MHI is that although it is a representation of the history of pixel-level changes, only one previous frame needs to be stored. However, at each pixel location, explicit information about its past is also lost in MHI when current change are updated to the model with their corresponding MHI values 'jumping' to the maximal value. To overcome this problem, Pixel Signal Energy was introduced to measure the mean magnitude of pixel-level temporal energy over a period of time defined by a backward window (Ng and Gong, 2001). The size of the backward window determines the number of frames (history) to be stored. However, this approach suffers from its sensitivity to noise and also being expensive to compute.

Here we propose a new representation, Pixel Change History (PCH), for measuring multi-scale temporal changes at each pixel. The PCH of a pixel is defined as:

$$P_{\varsigma,\tau}(x,y,t) = \begin{cases} \min\left(P_{\varsigma,\tau}(x,y,t-1)+\frac{255}{\varsigma}, 255\right) \\ \quad \text{if } D(x,y,t)=1 \\ \max\left(P_{\varsigma,\tau}(x,y,t-1)-\frac{255}{\tau}, 0\right) \\ \quad \text{otherwise} \end{cases} \quad (1)$$

where $P_{\varsigma,\tau}(x,y,t)$ is the PCH for a pixel at (x,y), $D(x,y,t)$ is a binary image indicating the foreground region, ς is an accumulation factor and τ is a decay factor. When $D(x,y,t)=1$, instead of jumping to the maximum value, the value of a PCH increases gradually according to the accumulation factor. When no significant pixel-level visual change is detected at a particular location (x,y) in the current frame, pixel (x,y) will be treated as part of the background and the corresponding pixel change history starts to decay. The speed of decay is controlled by a decay factor ς. The accumulation factor and the decay factor give us the flexibility of characterising pixel-level changes over time. In particular, large values of ς and τ imply that the history of visual change at (x,y) is considered over a longer backward temporal window. In the meantime, the ratio between ς and τ determines how much weight is put on the recent change.

If the binary image $D(x,y,t)$ in Eq. (1) is determined by the temporal difference between the current frame and the dynamic background maintained by an adaptive mixture model, a PCH based foreground model can be introduced to detect the medium and long term pixel changes. Specifically, we detect those pixels that are associated with medium term changes by the following condition:

$$|I(x,y,t)-I(x,y,t-1)| > T_M \quad (2)$$

where T_M is a threshold. Pixel level changes that do not satisfy the above condition are caused by long term changes such as the introduction of static novel objects into the scene or the removal of existing objects from the scene. Note that Eq. (2) is used for distinguishing the detected foreground pixels according to the nature of the visual changes. It is not used for detecting foreground pixels using background subtraction

We consider that Motion History Image (MHI) is a special case of PCH in that PCH image is equivalent to MHI when ς is set to 1. Figure 2 and Table 1 show an example of how PCH can provide better representation of the visual changes captured in the image frame. It is evident from Fig. 2(b) and Table 1 that the mean value of PCH of pixels corresponding to short term changes (e.g. a shopper passing by) is significantly lower than those for medium and long term changes (e.g. a shopper paying or a drink can being removed) and therefore provides us with a good measurement

Table 1. Compare the discriminative power of MHI and PCH on differentiating different types of visual changes. The discriminative power is indicated as the absolute difference value between the mean values for different visual changes shown in Fig. 2(b) (e.g. 100.62 for PCH and 26.14 for MHI between medium and short term changes, with the ratio of the two being 3.85).

	Long vs. short term	Medium vs. short term	Long vs. medium term
MHI discriminative power	55.39	26.14	29.51
PCH discriminative power	140.14	100.62	39.54
PCH vs. MHI ratio	2.53	3.85	1.34

for discriminating different types of visual changes. Compared to PCH, MHI has much weaker discriminative power (see Table 1). Furthermore, similar to that of Pixel Signal Energy (Ng and Gong, 2001), a PCH also captures a zero order pixel-level change, i.e. the mean magnitude of change over time. In addition, it is capable of capturing higher order temporal changes occurred at a pixel including the speed, trend (uphill or downhill) and phase of a change over time. It is important to point out that this measurement is different from that computed by multi-scale spatio-temporal filtering widely adopted for estimating apparent image motion such as optic flow. No spatio-temporal correspondence is established by computing PCH. It is also worth mentioning that our PCH is asymmetric in the time direction, i.e., it contains visual information accumulated only up to the current frame. Although adopting a symmetric measure could in theory lead to better representation, our PCH measure makes real-time event detection and recognition possible.

2.2. *From Pixel Groups to Unsupervised Clustering and Classification of Events*

Given detected pixel changes in each image frame, we aim to form discrete events. The connected component method is adopted to group those changed pixels. Small groups are then removed by a size filter and the rest groups with an average PCH (of the PCHs for all the pixels within each group) larger than a threshold T_B are referred to as salient pixel groups and considered as events. An event is represented by a 7-dimensional feature vector

$$\mathbf{v} = [\bar{x}, \bar{y}, w, h, R_m, M_p x, M_p y] \tag{3}$$

where $(\bar{x}, \bar{y})$ is the centroid of the salient pixel group, (w, h) are the width and height of the salient pixel group, R_m represents the percentage of those pixels in the group that satisfy Condition (2), and $(M_p x, M_p y)$ are a pair of first order moments of the PCH image within the salient pixel group. Among these features, $(\bar{x}, \bar{y})$ are location features, (w, h) are shape features, R_m is visual change type feature and $(M_p x, M_p y)$ are motion features capturing the direction of object motion direction.[1] Note that in our approach, salient pixel groups are defined within each image frame. Alternatively, salient groups can be defined in a spatio-temporal volume which could in theory lead to better clustering. One could adopt a method such as the one proposed by Greenspan et al. (2004). Alternatively, we have also developed an approach for salient event detection over a spatio-temporal volume using multi-scale entropy ratio over space and time presented elsewhere (Hung and Gong, 2004). However, such a spatio-temporal volume based events detection and recognition approach is always computationally expansive, and may not be tractable given the complexity of the activities captured in video footages. In order to achieve real-time performance, we decided to make a comprise between the speed and performance of the event detection and recognition algorithm by defining the salient groups within each image frame.

Salient pixel groups are clustered and classified unsupervised into different events in the 7-D feature space using a Gaussian Mixture Model (GMM). The GMM is estimated using Expectation-Maximisation (EM) (Bishop, 1995) and the model order of the GMM is determined using the Bayesian Information Criterion (BIC) (Schwarz, 1978).

Given a training data set $\mathbf{O}$ of salient pixel groups from some training image sequences, we aim to determine the best model order $\hat{k}$ (as the most likely number of different event classes) from a set of K competing models $\mathbf{m}_k$ parameterised by $\theta\mathbf{m}_k$ where $k \in \{1, \ldots, K\}$. The BIC model selection is formulated as:

$$\hat{\mathbf{m}}_k = \arg\min_{\mathbf{m}_k} \left\{ -\log P(\mathbf{O}|\mathbf{m}_k, \hat{\boldsymbol{\theta}}_{\mathbf{m}_k}) + \frac{D_k}{2} \log N \right\} \tag{4}$$

where $\hat{\boldsymbol{\theta}}_{\mathbf{m}_k} = \arg\max_{\boldsymbol{\theta}_{\mathbf{m}_k}} \{P(\mathbf{O} \mid \mathbf{m}_k, \boldsymbol{\theta}_{\mathbf{m}_k})\}$ is the Maximum Likelihood Estimation (MLE) of $\boldsymbol{\theta}_{\mathbf{m}_k}$, D_k is the dimensionality of $\boldsymbol{\theta}_{\mathbf{m}_k}$, and N is the size of the training data set. BIC can be derived as an approximation of Bayesian Model Selection (see Appendix A). The model order is therefore the number of GMM components k. If k ranges from 1 to K for the candidate GMMs, the optimal model order $\hat{k}$ estimated by the BIC is given by:

$$\hat{k} = \arg\min_{k} \left\{ -\sum_{i=1}^{N} \log f(\mathbf{y}_i \mid k, \hat{\boldsymbol{\theta}}(k)) + \frac{D_k}{2} \log N \right\} \tag{5}$$

where $f(\mathbf{y}_i \mid k, \hat{\boldsymbol{\theta}}(k))$ is the class-conditional Gaussian density function, $\mathbf{y}_i$ is the feature vector representing one data simple, and $\hat{\boldsymbol{\theta}}(k)$ are the mixture parameters estimated using EM and D_{k} is the number of parameters

needed for a k-component GMM. If full covariance matrix is used, Eq. (5) can be re-written as:

$$\hat{k} = \arg\min_{k} \left\{ -\sum_{i=1}^{N} \log f(\mathbf{y}_i \mid k, \hat{\boldsymbol{\theta}}(k)) + \frac{k-1}{2} \log N + \frac{q^2+3q}{4} k \log N \right\} \quad (6)$$

where q is the dimensionality of the feature space. To summarise, $\hat{k}$ estimated by Eq. (6) yields the most likely number of event classes given the training data set O. Salient pixel groups detected in novel image frames can then be classified as one of the $\hat{k}$ event classes in the 7-D feature space.

2.3. *Alternative Model Selection Criteria for Event Classification*

Model selection is key to unsupervised statistical modelling of data. Suppose that a data set arises from one of a set of candidate models, the problem is to choose the best candidate model for the given data set. Most model selection criteria are derived based on the principle that the optimal model not only best fits a given data set but also satisfies simplicity. This principle is known as the Ockham' Razor principle after the 13th century philosopher William of Ockham, and is widely adopted for determining model complexity, especially in the form of probabilistic model selection criteria (Mclachlan and Peel, 1997). Other model selection criteria include heuristic methods such as Fuzzy Hyper-Volume (FHV) (Gath and Geva, 1989) and evidence density (Roberts, 1997), and cross-validation method (Bishop, 1995; Smyth, 2000). Existing probabilistic model selection criteria can be classified into two categories: (1) Methods based on approximating the Bayesian Model Selection criterion (Raftery, 1995), such as Bayesian Information Criterion (BIC) (Schwarz, 1978), Laplace Empirical Criterion (LEC) (Roberts et al., 1998), and the Integrated Completed Likelihood (ICL) (Biernacki et al., 2000); (2) Methods based on the information coding theory such as the Minimum Message Length (MML) (Figueiredo and Jain, 2002), Minimum Description Length (MDL) (Rissanen, 1989), and Akaike's Information Criterion (AIC) (Akaike, 1973). Among these criteria, BIC is the most popular choice for determining the model order of a mixture model (Roberts et al., 1998; Figueiredo and Jain, 2002; Biernacki et al., 2000). It should be noted that BIC is formally, though not conceptually, coincides with Rissanen's Minimum Description Length (MDL) (Rissanen, 1989; Figueiredo and Jain, 2002).

In this section, we compare BIC with two popular model selection criteria, namely the Akaike Information Criterion (AIC) and cross-validation (CV). The basic idea of AIC is to select the model (represented as distribution densities) that minimises the difference between the density corresponding to a fitted model and that of the true model that generates the data. Akaike discovered that under the assumptions that (a) the true model is among the candidate models and (b) a set of regularity conditions holds that ensure the asymptotic properties of $\hat{\boldsymbol{\theta}}_{\mathbf{m}_k}$ (the MLE of $\boldsymbol{\theta}_{\mathbf{m}_k}$), the model that minimises

$$AIC = -\log P(\mathbf{O} \mid \mathbf{m}_k, \hat{\boldsymbol{\theta}}_{\mathbf{m}_k}) + D_k \quad (7)$$

should asymptotically approach the true model when the sample size N is approaching infinity, in the sense that the Kullback-Leibler (KL) divergence between these two models is approaching zero. In cross-validation a data set is repeatedly split into a training set and a test set, which are then used for model construction and model evaluation respectively. The test log-likelihood, i.e. the log-likelihood of observing the test set using the model built on the training set, is utilised for selecting the model order. It is shown in Smyth (2000) that the negative test log-likelihood is an unbiased estimator (within a constant) of the KL distance between the true model and candidate models and the selected model order would converge to the true model order given infinite sample size. Seemingly different, cross-validation is similar to AIC in that both criteria aim to select models that best predict unseen data.

BIC and AIC are asymptotically approximations to Bayesian Model Selection and K-L divergence model selection respectively. They are essentially the maximum likelihood of observing the data given a candidate model plus a penalty term which penalises the model complexity. Both BIC and AIC are accurate only when the sample size is large. However, comparing Eq. (4) with Eq. (7) indicates that BIC has stronger penalty term. It is especially the case when the sample size of the data set is large due to the factor that the penalty term of AIC does not increase with the sample size while the negative of the maximum likelihood of observing the

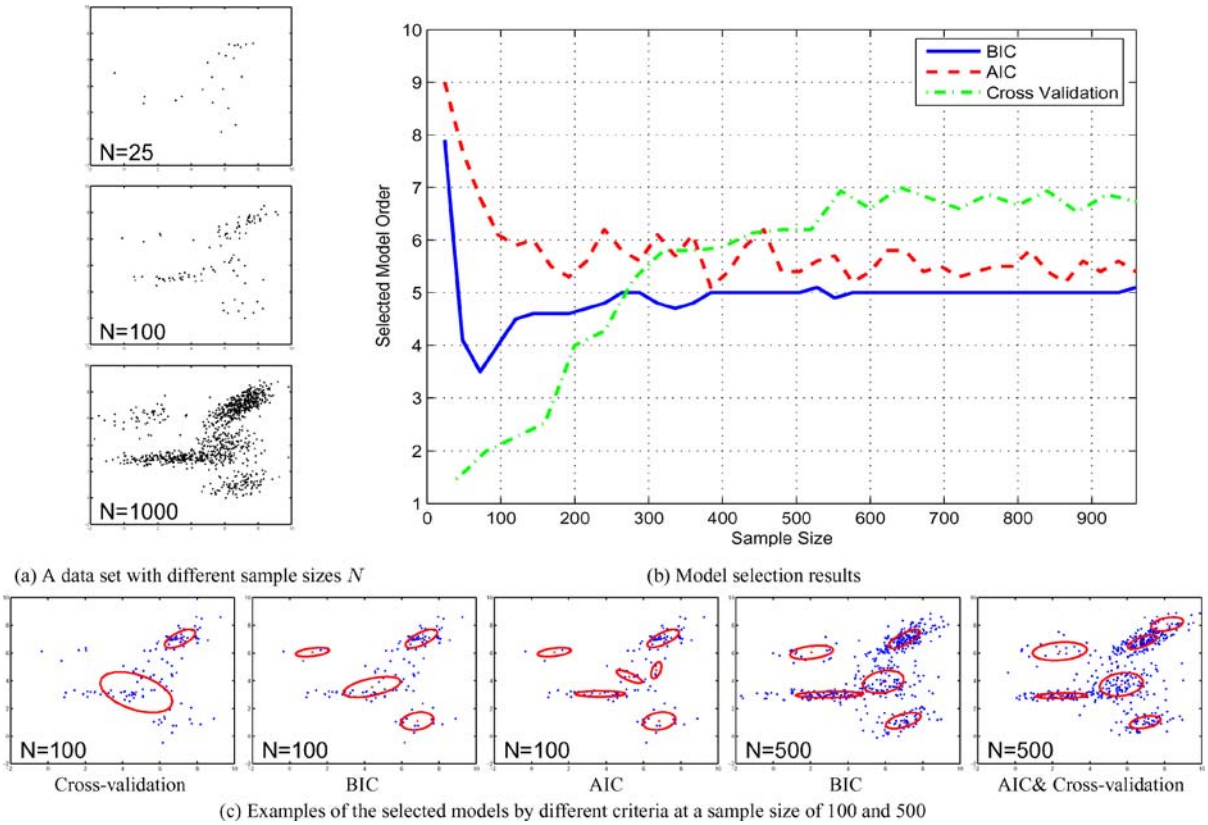


Figure 3. Model selection for a synthetic Gaussian data set using BIC, AIC, and cross-validation. The synthetic data were generated using a 5-component bivariate Gaussian mixture. (a) Shows examples of the data set with different sample sizes. The average number of mixture components determined by BIC, AIC, and cross-validation over 10 trails are plotted against the sample size in (b). Examples of models selected by BIC, AIC, and cross-validation are shown in. (c) F was set to 50 for the cross-validation experiment. It can be seen from (b) that when the sample size is large (e.g. $N > 10\ D_k$ where $D_k = 29$ in this case), the number of components determined by BIC converged to the true number 5. Whilst both AIC and cross-validation over-estimated the number of components, cross-validation gave the poorest result.

data does. AIC thus favours more complex models compared to BIC. This is illustrated in Fig. 3 by comparing BIC and AIC for selecting optimal number of components of a GMM using a 2-dimensional synthetic data set. Compared to BIC and AIC, the computational load of cross-validation for the same data set is increased by roughly a factor F, which is the number of partitions of the data set into the training and test sets (Smyth, 2000). The value of F is normally between 20 to 50 in practice. Besides the higher computational cost, another shortcoming of cross-validation is that it tends to over-estimate the model order, as can be seen in Fig. 3. It is not surprising to note that both AIC and cross-validation have the tendency of over-estimation, considering the theoretical similarity between these two criteria. The superiority of BIC over AIC and cross-validation on event classification is further demonstrated through our real-data experiments presented in Section 4.

3. Activity Modelling Using Dynamic Probabilistic Networks

For modelling group or interactive activities of multiple objects, we consider that an activity consists of a group of dynamically correlated discrete events representing significant scene changes over time. We propose to model a group of events as the observational input to a Dynamic Probabilistic Network (DPN).

3.1. Dynamic Probabilistic Networks

Static causal relationships represented by a conventional BBN[2] are limited for modelling correlations

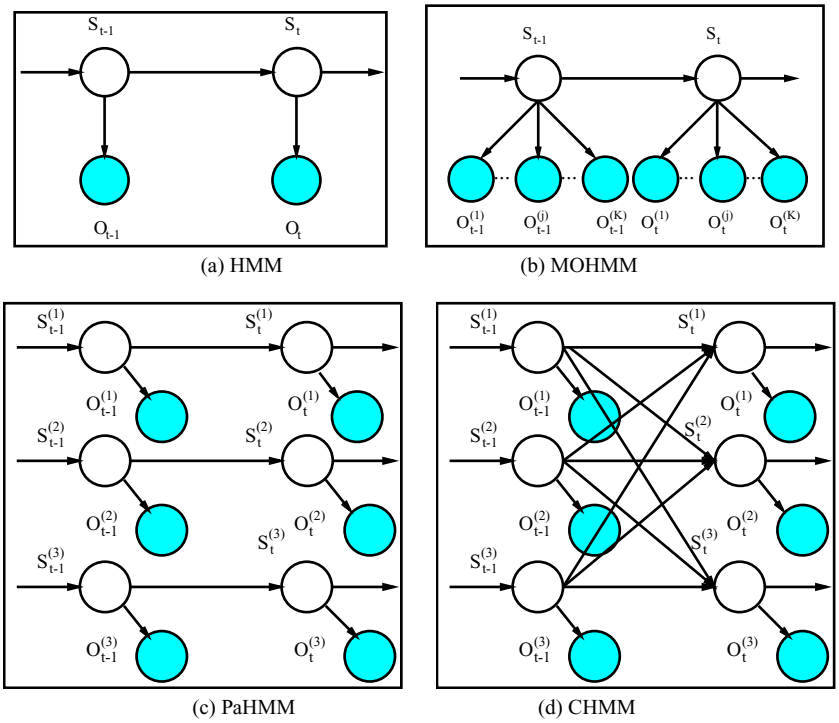


Figure 4. Three different types of Dynamic Bayesian Networks (DBNs) as extensions of a standard HMM. Observation nodes are shown as shaded circles and hidden nodes as clear circles.

among temporal states of multiple processes. Dynamic probabilistic networks and in particular Dynamic Bayesian Networks (DBNs) are BBNs that have been extended to model time series data (Ghahramani, 1998; Heckerman, 1995). More specifically, hidden nodes have been introduced in the topology of DBNs to represent hidden temporal states. A DBN $\mathbf{B}$ is described by two sets of parameters $(\boldsymbol{\lambda}, \boldsymbol{\theta}(\boldsymbol{\lambda}))$. The first set $\boldsymbol{\lambda}$ represents the structure of a DBN which includes the number of hidden state variables and observation variables at time t, and the topology of the network (set of directed arcs connecting nodes). The ith hidden state variable and the jth observation variable at time t are denoted as ${S_t}^{(i)}$ and ${O_t}^{(j)}$ respectively where $i \in \{1, \ldots, N_h\}$ and $j \in \{1, \ldots, N_o\}$, N_h and N_o are the number of hidden state variables and observation variables at each time instance respectively. The second set of parameters $\boldsymbol{\theta}(\boldsymbol{\lambda})$ quantifies the state transition probabilities/distributions $P(S_t^{(i)} \mid itPa(S_t^{(i)})$, the observation probabilities/distributions $P(O_t^{(j)} \mid Pa(O_t^{(j)})$, and the initial state distributions $P(S_1^{(i)})$ where $Pa(S_t^{(i)})$ are the parents of ${S_t}^{(i)}$ at $t-1$ (assuming first-order Markov models) and similarly, $Pa(O_t^{(i)})$ for observations. In this paper, unless otherwise stated, $S_t^{(i)}$ are discrete and $O_t^{(j)}$ are continuous random variables. Each observation variable has only hidden state variables as parents and the Conditional Probability Distributions (CPDs) are Gaussian. Examples of Dynamic Probabilistic Networks (DPNs) are shown in Fig. 4.

3.2. *Discovering the Structure of a DBN for Activity Modelling: Model Selection*

Instead of being fully connected as in the case of a CHMM, a Dynamically Multi-Linked Hidden Markov Model (DML-HMM) is proposed which aims to *only* connect a subset of relevant hidden state variables across multiple temporal processes. This is achieved by factorising the state space using Schwarz's Bayesian Information Criterion (BIC). The factorisation reduces the number of unnecessary parameters and caters for better network structure discovery.

We wish to simultaneously learn the temporal and causal correlations among events by finding a DBN model B parameterised by $(\boldsymbol{\lambda}, \boldsymbol{\theta}(\boldsymbol{\lambda}))$ that can best explain the observed events $\mathbf{O}$. Such a best explanation is quantified by the minimisation of a cost function. For

a Maximum Likelihood Estimation (MLE), the cost function is $-\log P(\mathbf{O} \mid \boldsymbol{\lambda}, \hat{\boldsymbol{\theta}}(\boldsymbol{\lambda}))$, the negative logarithm of the probability of observing $\mathbf{O}$ by model $\mathbf{B}$ where $\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda})$ are the parameters for the candidate structure $\boldsymbol{\lambda}$ that maximise the likelihood of observing data $\mathbf{O}$. $\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda})$ are estimated using an extended forward-backward algorithm for each candidate model structure, which is detailed later. A MLE of the structure of $\mathbf{B}$ in the most general case results in a fully connected DBN, which implies that any class of events would possibly cause all classes of events in the future. Therefore adding a penalty factor in the cost function to count for the complexity of a network is essential for extracting meaningful and computationally tractable causal relationships. To this end, we adopt BIC to measure the goodness of one hypothesised network model against that of another in describing a given dataset. Let us consider K competing DBN models of an activity consisting of a group of events. These models are denoted as $\mathbf{B}_k$ and parameterised by $(\boldsymbol{\lambda}_k, \hat{\boldsymbol{\theta}}(\boldsymbol{\lambda}_k))$ where $k \in \{1, \ldots, K\}$ and $\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda}_k)$ is a D_k-dimensional vector. Let $\mathbf{O} = (O_1, \ldots O_T)$ be an observation sequence where $O_t = (O_t^{(1)}, \ldots, O_t^{(N_o)})$ are the observation variables at time t, the BIC model selection is formulated as:

$$\hat{\mathbf{B}}_k = \arg\min_{\mathbf{B}_k} \left\{ -P(\mathbf{O} \mid \boldsymbol{\lambda}_k, \hat{\boldsymbol{\theta}}(\boldsymbol{\lambda}_k)) + \frac{D_k}{2} \log T \right\} \quad (8)$$

Let us further consider that the number of hidden processes is the number of event classes (see Section 2 on Event Recognition). We also consider two states for each hidden state variable, i.e. a binary variable switching between the status of `True` and `False`. The observation variables are continuous and given by the 7-D feature vector representing events (Eq. (3)). Their distributions are mixtures of Gaussian with respect to the states of their discrete parent nodes. For model learning, the distributions of the detected events are used to initialise the distributions of the observation vectors. The priors and transition matrices of states are initialised randomly. We then have:

$$\begin{aligned} &P(\mathbf{O} \mid \boldsymbol{\lambda}_k, \hat{\boldsymbol{\theta}}(\boldsymbol{\lambda}_k)) \\ &= \sum_{S_t^{(i)}} \left\{ \prod_{i=1}^{N_h} P\left(S_1^{(i)}\right) \prod_{t=2}^{T} \prod_{i=1}^{N_h} P\left(S_t^{(i)} \mid Pa\left(S_t^{(i)}\right)\right) \right. \\ &\left. \times \prod_{t=1}^{T} \prod_{j=1}^{N_o} P\left(O_t^{(j)} | Pa\left(O_t^{(j)}\right)\right) \right\} \end{aligned} \quad (9)$$

To effectively evaluate $P(\mathbf{O} \mid \boldsymbol{\lambda}_k, \hat{\boldsymbol{\theta}}(\boldsymbol{\lambda}_k))$, we formulate the following extended forward-backward algorithm for a dynamically linked probabilistic network of multiple temporal processes.

3.2.1. Learning Parameters of a Multi-Process Dynamic Probabilistic Network. Let us consider a DBN with C temporal processes and one hidden variable and one observation variable respectively for each temporal process at each time instance. We thus have $N_h = N_o = C$. It is assumed that all the hidden state variables are discrete and all the observation variables are continuous whose probability density functions are Gaussian with respect to each state of their parent hidden state variables. The parameter space thus consists of the following components:

1. The initial state distribution $\pi = \{\pi_{i^{(c)}}\}$ where $\pi_{i^{(c)}} = P(S_1^{(c)} = q_{i^{(c)}}) 1 \leq i^{(}c) \leq N^{(}c)$, and $1 \leq c \leq C$.
2. The state transition probability distribution $A = \{a_{Pa(j^{(c)})j^{(c)}}\}$ where $a_{Pa(j^{(c)})j^{(c)}} = P(S_{t+1}^{(c)} = q_{j^{(c)}} \mid Pa(S_{t+1}^{(c)}) = q_{Pa(j^{(c)})})$, $Pa(S_{t+1}^{(c)})$ are the hidden variables at time t on which $S_{t+1}^{(c)}$ is conditionally dependent, $Pa(j^{(c)})$ are subscripts of those discrete values that $Pa(S_{t+1}^{(c)})$ can assume, $1 \leq j^{(c)} \leq N^{(c)}$ and $1 \leq c \leq C$.
3. The observation probability distribution $B = \{b_{i^{(c)}}(O_t^{(c)})\}$ where $b_{i^{(c)}}(O_t^{(c)}) = \mathcal{N}(O_t^{(c)}; \mu_{i^{(c)}}, \mathbf{U}_{i^{(c)}})$, $\mu_{i^{(c)}}$ and $\mathbf{U}_{i^{(c)}}$ are the mean vector and covariance matrix of the normal (Gaussian) distribution with respect to $S_t^{(c)} = q_{i^{(c)}}$, $1 \leq i^{(c)} \leq N^{(c)}$ and $1 \leq c \leq C$.

Given an observation sequence $\mathbf{O}$ and a model structure $\boldsymbol{\lambda}$, we need to determine the model parameters $\boldsymbol{\theta}(\boldsymbol{\lambda}) = \{A, B, \pi\}$ that maximise the probability of the observation sequence given the model structure $P(\mathbf{O} \mid \boldsymbol{\lambda}, \boldsymbol{\theta}(\boldsymbol{\lambda}))$. There is no analytical solution to determine the optimal parameters given a finite observation sequence. However, the parameters can be estimated iteratively using an extended forward-backward (Baum-Welch) algorithm (Baum and Petrie, 1966). Let us first define the following variables:

- The forward variable $\alpha_t\left(i^{(1)}, \ldots, i^{(C)}\right) = P(O_1, O_2, \ldots, O_t, S_t^{(1)} = q_{i^{(1)}}, \ldots, S_t^{(C)} = q_{i^{(C)}} \mid \boldsymbol{\lambda}, \boldsymbol{\theta}(\boldsymbol{\lambda}))$,

$$\xi_t(i^{(1)},\ldots,i^{(C)},j^{(1)},\ldots,j^{(C)}) = \frac{\beta_{t+1}(j^{(1)},\ldots,j^{(C)})\prod_{c=1}^{C}\alpha_t(i^{(1)},\ldots,i^{(C)})a_{Pa(j^{(c)})j^{(c)}}b_{j^{(c)}}(O_{t+1}^{(c)})}{P(\mathbf{O}|\boldsymbol{\lambda},\boldsymbol{\theta}(\boldsymbol{\lambda}))}$$

i.e., the probability of the partial observation sequence until time t and states for $S_t^{(1)},\ldots,S_t^{(C)}$ given the model $\boldsymbol{\lambda}$ and $\boldsymbol{\theta}(\boldsymbol{\lambda})$:

$$\alpha_t\big(i^{(1)},\ldots,i^{(C)}\big) = \begin{cases} \prod_{c=1}^{C}\pi_{i^{(c)}}b_{i^{(c)}}\big(O_t^{(c)}\big) & \text{if } t=1 \\ \prod_{c=1}^{C}\Big(\Big(\sum_{j^{(1)},\ldots,j^{(C)}}\alpha_{t-1}\big(j^{(1)},\ldots,j^{(C)}\big) \times a_{Pa(i^{(c)})i^{(c)}}\Big)b_{i^{(c)}}\big(O_t^{(c)}\big)\Big) & \text{if } 1<t\le T \end{cases}$$

- The backward variable $\beta_t(i^{(1)},\ldots,i^{(C)}) = P(O_t,\ldots,O_T,S_t^{(1)}=q_{i^{(1)}},\ldots,S_t^{(C)}=q_{i^{(C)}}\,|\,\boldsymbol{\lambda},\boldsymbol{\theta}(\boldsymbol{\lambda}))$, i.e., the probability of the partial observation sequence from $t+1$ to T, given the states for $S_t^{(1)},\ldots,S_t^{(C)}$ and the model $\boldsymbol{\lambda}$ and $\boldsymbol{\theta}(\boldsymbol{\lambda})$:

$$\beta_t\big(i^{(1)},\ldots,i^{(C)}\big) = \begin{cases} 1 & \text{if } t=T \\ \sum_{j^{(1)},\ldots,j^{(C)}}\Big(\prod_{c=1}^{C}\big(a_{Pa(j^{(c)})j^{(c)}}b_{j^{(c)}}\big(O_t^{(c)}\big)\big) \times \beta_{t+1}\big(j^{(1)},\ldots,j^{(C)}\big)\Big) & \text{if } 1\le t<T \end{cases}$$

where $P(\mathbf{O}\,|\,\boldsymbol{\lambda},\boldsymbol{\theta}(\boldsymbol{\lambda}))$ can be computed using the forward and backward variables:

$$P(\mathbf{O}\,|\,\boldsymbol{\lambda},\boldsymbol{\theta}(\boldsymbol{\lambda})) = \sum_{i^{(1)},\ldots,i^{(C)}}\alpha_t\big(i^{(1)},\ldots,i^{(C)}\big)\,\beta_t\big(i^{(1)},\ldots,i^{(C)}\big) \tag{10}$$

- $\gamma_t(i^{(1)},\ldots,i^{(C)}) = P\big(S_t^{(1)}=q_{i^{(1)}},\ldots,S_t^{(C)}=q_{i^{(C)}}\,|\,\mathbf{O},\boldsymbol{\lambda},\boldsymbol{\theta}(\boldsymbol{\lambda})\big)$, i.e., the probability of being at certain states at time t, given the model and observation sequence:

$$\gamma_t\big(i^{(1)},\ldots,i^{(C)}\big) = \frac{\alpha_t(i^{(1)},\ldots,i^{(C)})\beta_t(i^{(1)},\ldots,i^{(C)})}{\sum_{i^{(1)},\ldots,i^{(C)}}\alpha_t(i^{(1)},\ldots,i^{(C)})\beta_t(i^{(1)},\ldots,i^{(C)})}$$

Denote the current parameter estimates as $\boldsymbol{\theta}(\boldsymbol{\lambda}) = \{A,B,\pi\}$, the re-estimated parameters $\bar{\boldsymbol{\theta}}(\boldsymbol{\lambda}) = \{\bar{A},\bar{B},\bar{\pi}\}$ can be computed using the following re-estimation formula:

$$\bar{\pi}_{i^{(c)}} = \sum_{i^{(1)},\ldots,i^{(c-1)},i^{(c+1)},\ldots,i^{(C)}}\gamma_1\big(i^{(1)},\ldots,i^{(C)}\big) \tag{11}$$

$$\bar{a}_{Pa(j^{(c)})j^{(c)}} = \frac{\sum_{t=1}^{T-1}\sum_{j^{(1)},\ldots,j^{(c-1)},j^{(c+1)},\ldots,j^{(C)},i^{(c')}\neq Pa(j^{(c)})}\xi_t(i^{(1)},\ldots,i^{(C)},j^{(1)},\ldots,j^{(C)})}{\sum_{t=1}^{T}\sum_{i^{(c')}\neq Pa(j^{(c)})}\gamma_t(i^{(1)},\ldots,i^{(C)})} \tag{12}$$

- $\xi_t(i^{(1)},\ldots,i^{(C)},j^{(1)},\ldots,j^{(C)}) = P(S_t^{(1)}=q_{i^{(1)}},\ldots,S_t^{(C)}=q_{i^{(C)}},S_{t+1}^{(1)}=q_{j^{(1)}},\ldots,S_{t+1}^{(C)}=q_{j^{(C)}}\,|\,\boldsymbol{\lambda},\boldsymbol{\theta}(\boldsymbol{\lambda}))$, i.e., the probability of being at certain states at time t and $t\times 1$, given the model and observation sequence:

$$\bar{\mu}_{i^{(c)}} = \frac{\sum_{t=1}^{T}\Big(\sum_{i^{(1)},\ldots,i^{(c-1)},i^{(c+1)},\ldots,i^{(C)}}\gamma_t(i^{(1)},\ldots,i^{(C)})\Big)O_t^{(c)}}{\sum_{t=1}^{T-1}\sum_{i^{(1)},\ldots,i^{(c-1)},i^{(c+1)},\ldots,i^{(C)}}\gamma_t(i^{(1)},\ldots,i^{(C)})} \tag{13}$$

$$\bar{\mathbf{U}}_{i^{(c)}} = \frac{\sum_{t=1}^{T}\Big(\sum_{i^{(1)},\ldots,i^{(c-1)},i^{(c+1)},\ldots,i^{(C)}}\gamma_t(i^{(1)},\ldots,i^{(C)})\Big)\big(O_t^{(c)}-\mu_{i^{(c)}}\big)\big(O_t^{(c)}-\mu_{i^{(c)}}\big)^T}{\sum_{t=1}^{T}\sum_{i^{(1)},\ldots,i^{(c-1)},i^{(c+1)},\ldots,i^{(C)}}\gamma_t(i^{(1)},\ldots,i^{(C)})} \tag{14}$$

If we iteratively use $\bar{\boldsymbol{\theta}}(\boldsymbol{\lambda})$ to replace $\boldsymbol{\theta}(\boldsymbol{\lambda})$ and repeat the re-estimation calculation until some limiting point is reached, the final result is a maximum likelihood estimate of parameters $\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda})$. Notice that since the forward-backward algorithm is essentially a EM algorithm (Rabiner, 1989), $P(\mathbf{O} \mid \boldsymbol{\lambda}, \boldsymbol{\theta}(\boldsymbol{\lambda}))$ is only locally maximised by the estimated parameters. In general cases, the optimisation surface has multiple local maxima. The forward-backward algorithm is thus sensitive to initialisation.

In search for the optimal model $\hat{\mathbf{B}}_k$ with minimal BIC value, for each candidate model structure $\boldsymbol{\lambda}_k$, the corresponding Maximum Likelihood Estimation (MLE) of the parameters $\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda}_k)$ are estimated iteratively using the extended forward-backward algorithm. After parameter learning the BIC value can be computed using Eq. (8). Alternatively, parameter and structure learning can be performed within a single EM process using a structured EM algorithm (Friedman et al., 1998). It is worth mentioning that for our case, the structure search space is limited to only the inter-links among different temporal processes since the number of states for each hidden variable has been fixed.

3.2.2. Observations. Comparing DML-HMM with CHMM, it is clear that DML-HMM will always consist of more optimised factorisation of the state space and most likely have less connections. This allows for more tractable computation when reasoning about complex group activities. In addition, a more subtle but perhaps also more critical advantage of DML-HMM over CHMM is its ability to cope with noise. Given sufficiently noise-free data, it is possible for CHMM to learn the correct relationships between coupled hidden temporal processes. However, with noisy data, probability propagation travelling freely among all the hidden state variables during the EM parameter estimation, CHMM can be led to capture structures heavily biased by noise, especially when there are a large number of hidden processes. Similar problems should surface for MOHMM. Since no factorisation is performed in the state space, MOHMM needs far more parameters compared to DML-HMM and CHMM. As for PaHMM, although it may not be easily influenced by noise, it will pay the price for discarding any correlations between multiple temporal processes. This will be shown in our experiments in Section 4.

Model selection criteria other than BIC can also be used for determining the topology of a DML-HMM. An example of selecting the optimal topology of DML-HMM using a synthetic data set is shown in Fig. 5 to compare BIC with AIC. It is found that, similar to the case of determining the optimal number of components of a GMM (Section 2.3), AIC tends to over-estimate, i.e. select a DBN with more-than-necessary inter-links. Similar results are also obtained in our experiments presented in Section 4. It is worth mentioning that the number of candidate topologies of a DML-HMM grows exponentially with the number of temporal processes. The topology search space becomes huge even for a DML-HMM consisting of a small number of temporal processes. The much higher computational cost of cross-validation makes it unsuitable for determining the topology of a DML-HMM.

3.3. Understanding Behaviour Using a Learned Activity Model

Once learned, an activity model using a DML-HMM can be utilised to identify the key stages of the activity and characterise the temporal and causal correlations among them. These are semantic descriptions of the activity which are automatically generated through unsupervised learning. Activity recognition can also be perform given different models built for different activities. Moreover, the learned model can be used to improve the accuracy of event detection and classification through inference of hidden states, which leads to better understanding of the behaviours of each individual objects involved in the activity. Our approach thus provides both a bottom-up and top-down mechanism for understanding behaviours.

3.3.1. Extracting Semantics of Activity from Automatically Generated Activity Graphs. The temporal and causal correlations among events are quantified by the structure and parameters of DBNs learned using the training data. Once trained, the DBNs aim to encode the understanding of the dynamics of the scene. The parameters of the trained DBNs can thus be utilised to extract high level semantics from the scene. One of the important semantics we wish to extract is the structure of the activity interpreted by correlated events. To this end, we first automatically generate an activity transition matrix from the transition matrices of the trained DBNs. Important activity stages can then be identified from this activity transition matrix. The temporal and causal correlations among different activity stages are also encoded in this matrix. More specifically, we follow the following procedure after the structure and

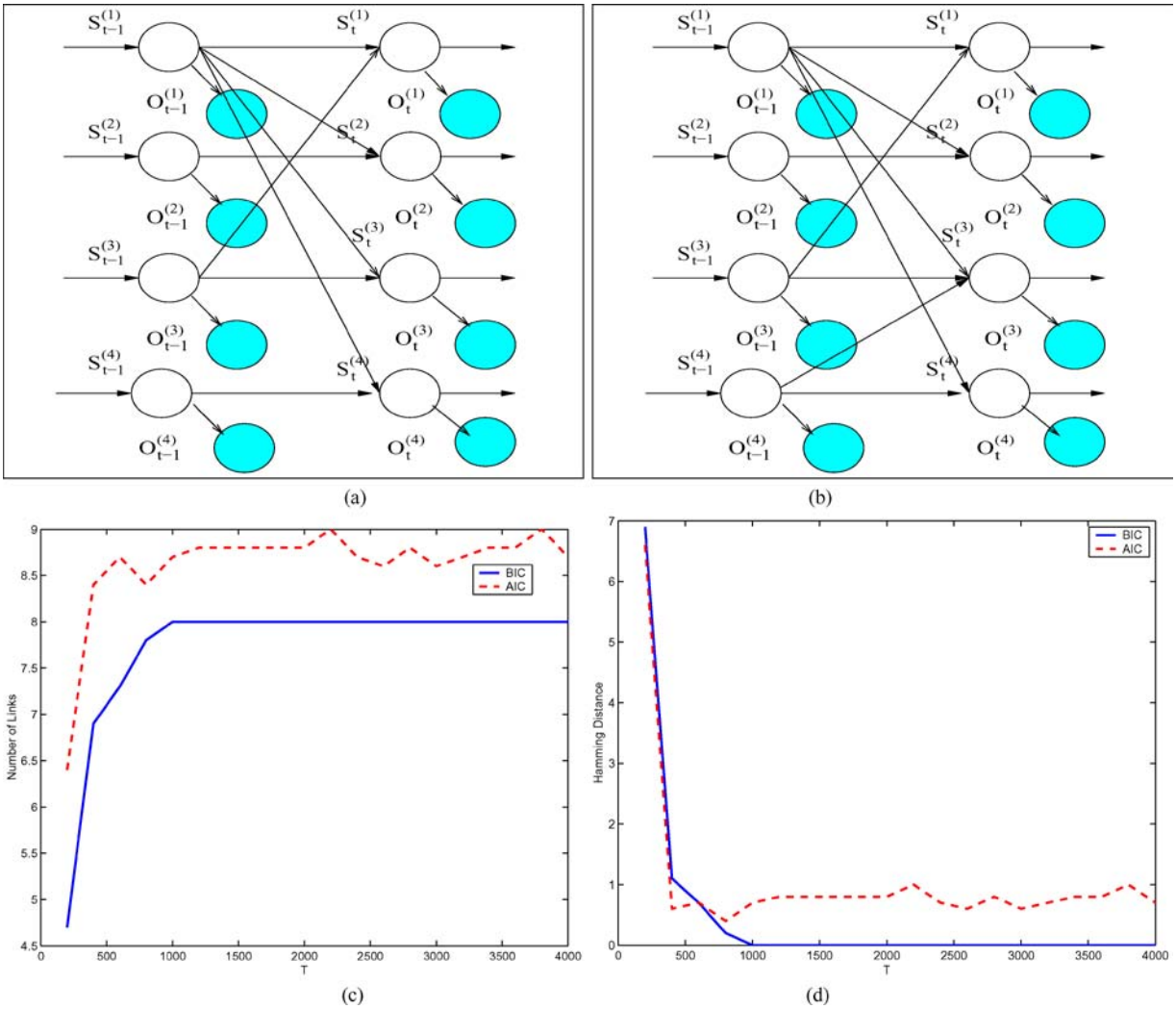


Figure 5. Comparing topology learning of DML-HMM using BIC and AIC. (a) shows the true topology of the DML-HMM. Each hidden state variable has two states; Each observation variable is a 7-D vector whose distribution is mixtures of Gaussian with respect to the states of its parent node. The training data sets were obtained by sampling the true DML-HMM with added uniformly distributed random noise. We also assume that 25% of the observations were labelled to wrong temporal processes. The average number of inter-links among different temporal processes (over 2 time instance) determined by BIC and AIC over 10 trails are plotted against the sample size in (c) (true number is 8). The average Hamming distance between the estimated topology and the true topology are shown in (d). It can be seen from (c) and (d) that when the sample size is large (e.g. $N > 3D_k$ where $D_k = 352$ in this case), correct topology was selected by BIC, while AIC over-estimated the number of inter-links. (b) shows an example of the topology selected by AIC when $N > 3D_k$.

parameter learning of a C-temporal process DBN for activity modelling:

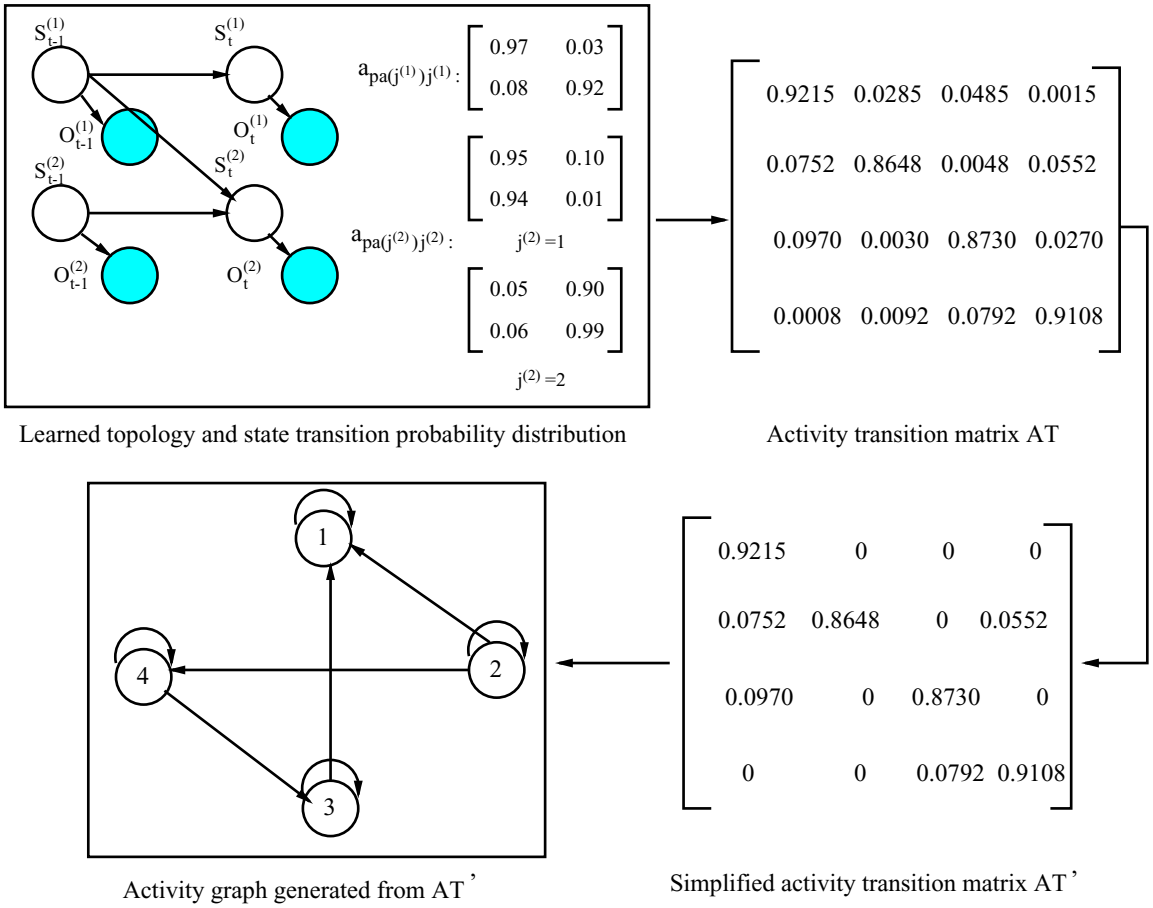
1. Compute the activity transition matrix $AT = \{at_{ij}\}$ where $1 \leq i, j \leq 2^C$. This is a $2^C \times 2^C$ matrix. Each entry of the matrix at_{ij} is computed as:

$$at_{ij} = \prod_{c=1}^{C} a_{Pa(j^{(c)})j^{(c)}} \tag{15}$$

where $j = \sum_{c=1}^{C} 2^{(c-1)}(j^{(c)} - 1)$ and $i = \sum_{c=1}^{C} 2^{(c-1)}(i^{(c)} - 1)$ for $i^{(c)}$ that satisfies: $i^{(c)} \in \{Pa(j^{(c)})\}$. at_{ij} represents the probability of transferring from activity stage i at time instance t to activity stage j at time instance $t+1$.

2. Obtain a simplified transition matrix $AT' = \{at'_{ij}\}$:

$$at'_{ij} = \begin{cases} at_{ij} & \text{if } at_{ij} > Th_{tr} \\ 0 & \text{otherwise} \end{cases}$$



Learned topology and state transition probability distribution

Activity transition matrix AT

Activity graph generated from AT'

Simplified activity transition matrix AT'

Figure 6. An example of automatically generating activity graph from a learned DML-HMM for activity modelling. Two classes of events, denoted as $e1$ and $e2$, are modelled using temporal processes 1 & 2 respectively. The learned topology indicates the $e2$ can be caused by $e1$ (reflected by the arc pointing from temporal processes 1 to 2). The simplified activity transition matrix AT' was obtained by thresholding the activity transition matrix AT. The number of non-zero diagonal elements of AT' corresponds to the number of key activity stages interpreted as the co-occurrence of the two classes of events. In this case, 4 key activity stages were detected. The causal relationships among these 4 stages are reflected by the directed arcs connecting the nodes in the activity graph.

By doing this, we remove the activity stage transition that is very unlikely to happen which may be cause by errors in event recognition. Th_{tr} is set to 0.05.

3. Generate an activity graph automatically from AT' where each state of the activity transition matrix is represented as a node and the temporal and causal correlations (corresponding to the non-zero entries of AT') are reflected by the directed arcs pointing from one node to another.

This procedure can be implemented to a CHMM, PaHMM and DML-HMM. For a DBN with single process such as a MOHMM, the transition matrix can be directly used as the activity transition matrix and step 1 of the procedure can be skipped. The process of generating an activity graph is illustrated using a simple DML-HMM with two temporal processes in Fig. 6.

3.3.2. Activity Recognition. Suppose we have learned K different DBNs $\mathbf{B}_k$ (where $1 \leq k \leq K$) for K different activities respectively. Now an unknown activity is captured in the image sequence and we wish to recognise it as one of the K candidate activities. Again, it is a model selection problem. Adopting the Bayesian model selection criterion, the model $\hat{\mathbf{B}}_k$ associated with the most likely activity is determined as:

$$\hat{\mathbf{B}}_k = \arg\max_{\mathbf{B}_k} P(\mathbf{O} \mid \mathbf{B}_k) P(\mathbf{B}_k) \tag{16}$$

where $P(\mathbf{O} \mid \mathbf{B}_k)$ is the probability of observing the new data which can be computed using Eq. (10) given the learned model structure and parameters, and $P(\mathbf{B}_k)$ is the *a priori* probability of observing the kth candidate activity. $P(\mathbf{B}_k)$ represents our priori knowledge about the unknown activity. If no such priori knowledge is available, the unknown activity is recognised as one of the candidate activities whose learned model can best explained the observation.

3.3.3. Improving Event Recognition through State Inference. Since each hidden state variable in the structure of a DML-HMM corresponds to whether a particular class of events are detected in the scene at a particular time instance, it can be utilised to improved the event detection and classification results. To this end, given a sequence of detected and recognised events, an extended Viteribi algorithm (Forney, 1973) is formulated to infer the hidden states using the learned model. Let us first define the following variables:

- $\delta_t(i^{(1)}, \ldots, i^{(C)}) = \max_{\{S_1^c\}, \ldots, \{S_{t-1}^c\}} P(\{S_1^{(c)}\}, \ldots, \{S_{t-1}^{(c)}\}, S_t^{(1)} = q_{i^{(1)}}, \ldots, S_t^{(C)} = q_{i^{(C)}}, O_1, \ldots, O_t \mid \boldsymbol{\lambda}, \boldsymbol{\theta}(\boldsymbol{\lambda}))$ where $\{S_t^c\} = \{S_t^{(1)}, \ldots, S_t^{(C)}\}$. $\delta_t(i^{(1)}, \ldots, i^{(C)})$ is the highest probability of the partial observation sequence until time t and a sequences of hidden states from time 1 to time t, given the model $\boldsymbol{\lambda}$ and $\boldsymbol{\theta}(\boldsymbol{\lambda})$.
- $\varphi_t(i^{(1)}, \ldots, i^{(C)})$, which is an array used to store the best state sequence. The best hidden states at time t for the C hidden state variables are denoted as $\{S_t^{*(c)}\}$.

The extended Viteribi algorithm has the following steps:

1. Initialisation:

$$\delta_1\left(i^{(1)}, \ldots, i^{(C)}\right) = \prod_{c=1}^{C} \pi_{i^{(c)}} b_{i^{(c)}}\left(O_t^{(c)}\right)$$
$$\varphi_1\left(i^{(1)}, \ldots, i^{(C)}\right) = \mathbf{0}$$

2. Recursion:

$$\delta_t\left(i^{(1)}, \ldots, i^{(C)}\right) = \max_{j^{(1)}, \ldots, j^{(C)}} \times \left\{ \prod_{c=1}^{C} \delta_{t-1}\left(j^{(1)}, \ldots, j^{(C)}\right) a_{Pa(i^{(c)})i^{(c)}} b_{i^{(c)}}\left(O_t^{(c)}\right) \right\}$$

$$\varphi_t\left(i^{(1)}, \ldots, i^{(C)}\right) = \arg \max_{j^{(1)}, \ldots, j^{(C)}} \times \left\{ \prod_{c=1}^{C} \delta_{t-1}(j^{(1)}, \ldots, j^{(C)}) \times a_{Pa(i^{(c)})i^{(c)}} b_{i^{(c)}}\left(O_t^{(c)}\right) \right\}$$

where $1 < t \leq T$.

3. Termination:

$$\left\{S_T^{*(c)}\right\} = \arg \max_{i^{(1)}, \ldots, i^{(C)}} \delta_t\left(i^{(1)}, \ldots, i^{(C)}\right)$$

4. Best state sequence backtracking:

$$\left\{S_t^{*(c)}\right\} = \varphi_{t+1}(\{S_{t+1}^{*(c)}\})$$

where $t = T - 1, T - 2, \ldots, 1$.

The inferred hidden state sequence represents the understanding of the learned model regarding to the occurrences of different events. As can be seen from the formulation of the extended Viteribi algorithm, the temporal and causal correlations among different events, which have been learned from training data, are utilised to explain away the errors in event detection and classification when the hidden states are inferred on the new observations. This leads to more accurate event detection and classification. A DML-HMM for activity modelling thus provides us with a top-down mechanism for the learned knowledge to be utilised for improving the activity representation, which is the input to the model.

4. Experiments

We have described in the previous sections an approach for modelling activities involving simultaneous movements of multiple objects. Using this approach, discrete scene events are detected and classified before being fed into a DML-HMM in order to reason about the temporal and causal correlations among different event classes. Our approach is fully unsupervised in the sense that both the parameters of the event classifier and the DML-HMM topology and parameters are learned without manual data labelling. In this section, we illustrate the effectiveness of our approach with two examples of modelling activities captured in an indoor and an outdoor cluttered scenes.

4.1. Modelling Shopping Activities

4.1.1. Data Set. A simulated 'shopping scenario' was captured on a 20 minutes video at 25 Hz. Some typical scenes can be seen in Fig. 7(a). The scene consists of a shopkeeper sat behind a table on the right side of the view. Drink cans were laid out on a display table. Shoppers entered from the left and either browsed without paying or took a can and paid for it. The data used for this experiment were sampled at 8 frames per second with total number of 5699 frames of images sized 320×240 pixels.

4.1.2. Event Recognition. We adopted the adaptive Gaussian mixture background model (Stauffer and Grimson, 2000) for dynamic background modelling. The parameters were set as: learning rate $\alpha = 0.002$, background model threshold $T = 0.7$, six Gaussian components were maintained and a diagonal co-variance matrix was adopted. The parameters for pixel grouping were: $\varsigma = 12$, $\tau = 10$, $T_M = 10$ and $T_B = 100$. Only those salient pixel groups whose sizes were larger than 40 were considered. The first 2700 frames of the sequence were used for training. In each frame, salient pixel groups were estimated in a 7-D feature space given by Eq. (3). Unsupervised clustering was performed on all the salient pixel groups from the 2700 frames in the training set where 2118 events were detected and classified into 5 different classes through model selection using BIC. The event clustering and classification results are illustrated in Fig. 7(b). Some examples of detected and classified events are shown in Fig. 7(a). The location and the temporal order of the events throughout the sequence are shown in Fig. 7(d). We can observe that the 5 classes of events corresponded correctly to 5 key constituents of the shopping activity. They were labelled as `canTaken`, `entering/leaving`, `shopkeeper`, `browsing` and `paying` respectively. For comparison, clustering and classification were also performed using AIC. Figure 7(c) shows that 7 clusters were formed using AIC and cross-validation. Compared with the result obtained using BIC shown in Fig. 7(b), the `shopkeeper` event class was split into two classes, so was the `paying` event class. The trained model was then used to recognise events detected in the rest of the 20 minutes video where 252016 events were detected and classified into the 5 event classes. These results reinforce the early results using synthetic data shown in Fig. 3.

It was noted that different classes of events occurred simultaneously. It is also true that our event recognition model made errors. Some of the errors were caused by the occlusion, closeness and visual similarity among different events. However, since they occurred in different contexts, semantically they should belong to different event classes. For example, when a shopper stands in front of the shopkeeper, it is impossible to tell whether he is going to pay unless one takes into consideration whether any drink can was taken a moment ago. The event classifier is therefore expected to make such errors without taking into account the temporal and causal correlations among different classes of events. Such temporal and causal contexts are modelled using the Dynamically Multi-Linked Hidden Markov Model (DML-HMM) as follows.

4.1.3. Activity Modelling Using DML-HMM. For modelling the shopping activity with 5 different classes of events, we employed a DML-HMM to model the temporal and causal correlations among different events. The topology of the DML-HMM were learned from training data using BIC (see Fig. 8(a)). For comparison, the topology learned using AIC using the same training data is shown in Fig. 8(b). Comparing Fig. 8(b) with (a), it is obvious that a more complex model was selected by AIC. The discovered dynamic correlations among different classes of events are embodied in the topology of the DML-HMM. Compared with the expected structure of the shopping activity as shown in Fig. 8(c), the causal relationships among different classes of events and the temporal structure of the activity were mostly discovered correctly by BIC.

4.1.4. Comparing Different DBNs for Activity Modelling. Experiments were carried out to compare the performance of our DML-HMM to that of a MOHMM, PaHMM and CHMM on modelling the shopping activity. The first 2700 frames of the image sequence were used as the training set and the rest 2999 frames were used as the testing set. The number of events detected in the training and testing set were 2118 and 2516 respectively. These events were labelled into 5 event classes, as described in Section 4.1.2. The DML-HMM shown in Fig. 8(a) was adopted. For the PaHMM and CHMM, there were also 5 temporal processes in their topologies. There were 5 observation variables at each time instance in the topology of the MOHMM. The number of parameters to be estimated for the MOHMM, PaHMM, CHMM and DML-HMM

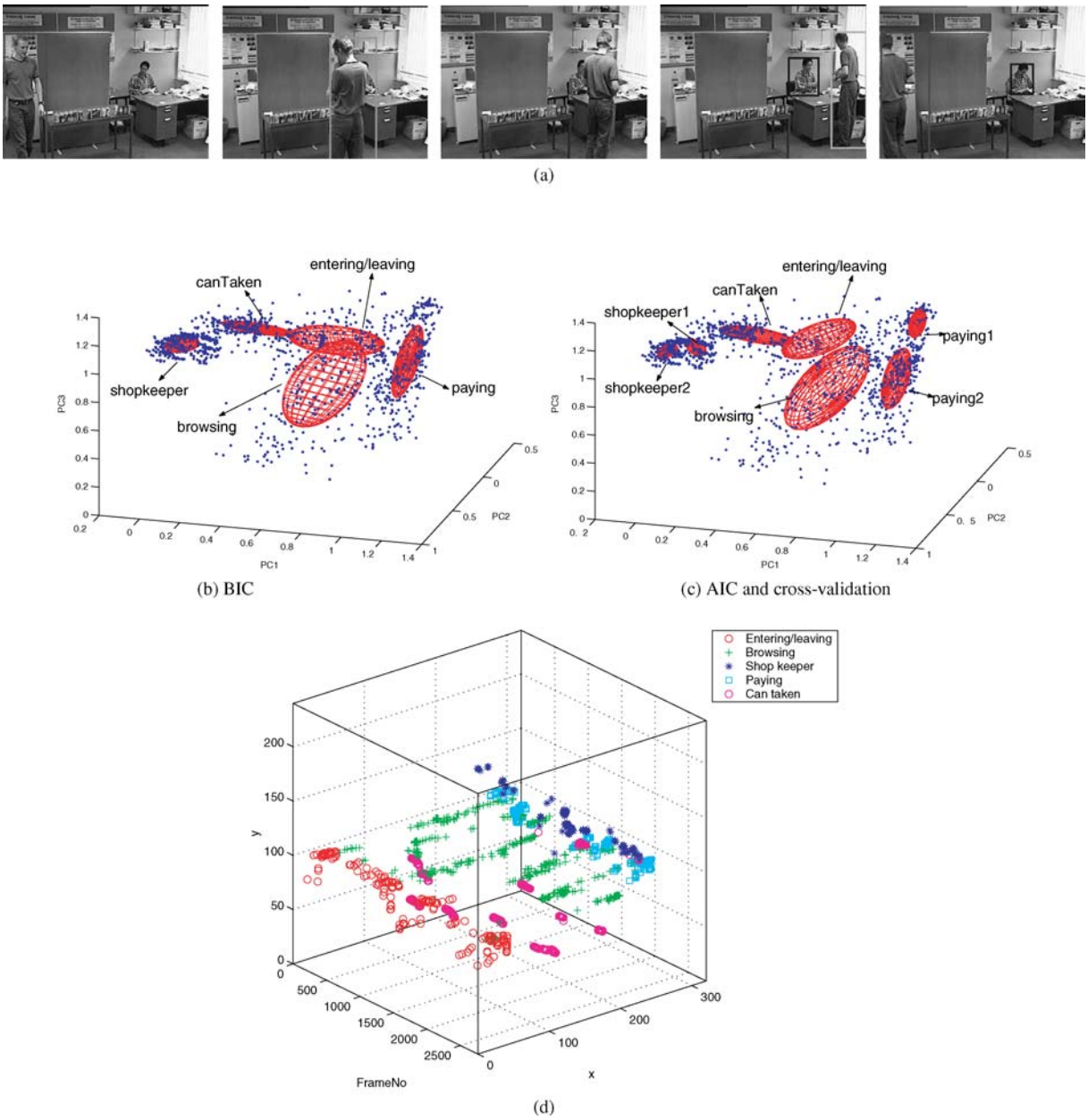


Figure 7. Event detection and classification in a shopping scene. (a) Examples of detected and classified events in the image frame. Events belonging to different classes are indicated with bounding boxes in different colours. (b) and (c) Unsupervised event clustering of the training set in the 7-dimensional feature space (only the first 3 principal components are shown for visualisation) by BIC, AIC and cross-validation respectively. (d) The where about and temporal order of the 5 classes of events being detected throughout the training sequence. Centroids of different classes of events are depicted using different symbols.

were 6655, 375, 675, and 419 respectively. In the following we present results on (1) extracting activity graphs using the training set, and (2) explaining away errors in event recognition, which were conducted on the testing set using the learned models.

Activity graphs—Four different activity graphs can be automatically generated using the procedure described in Section 3.3.1 from the trained model state transition matrices of a MOHMM, PaHMM, CHMM and DML-HMM respectively. Figure 9 shows the activity transi-

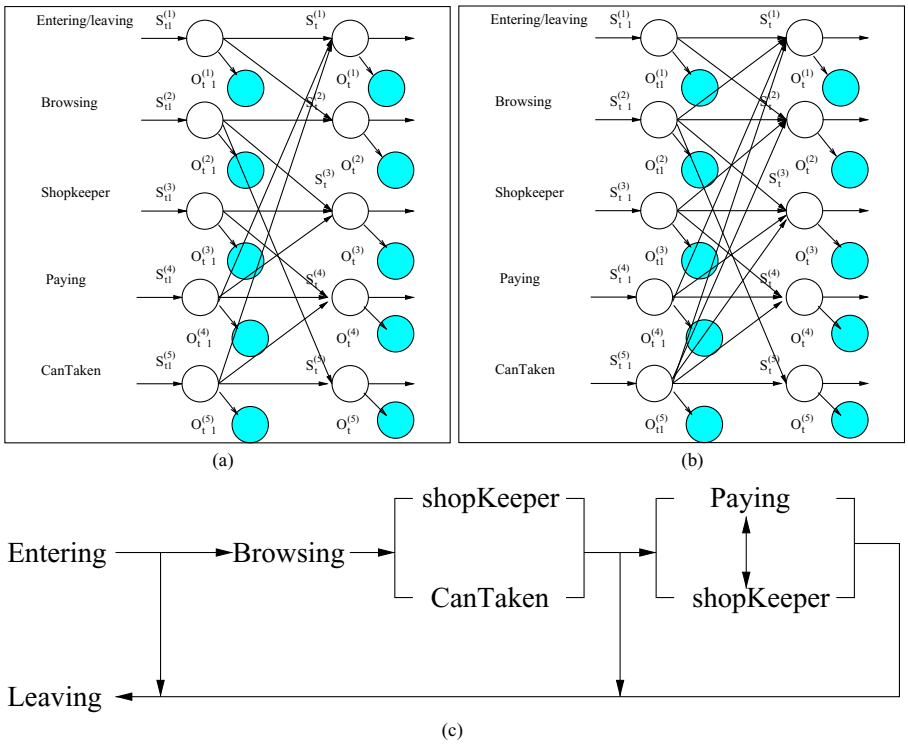


Figure 8. (a) and (b) Two DML-HMMs for modelling 5 temporal processes correlating to 5 different classes of events involved in the shopping activity. Their structure were determined by BIC, and AIC respectively using the same training set. (c) The expected causal and temporal structure of the shopping activity.

tion matrices and corresponding activity graphs for the four models. It can be seen from Fig. 9 that although the state transition matrices were initialised randomly with no constraint on their transitions, the learned activity transition matrices for the PaHMM, CHMM and DML-HMM have sparse structures, which were found to be insensitive to initialisation. On the contrary, the activity transition matrix for the trained MOHMM was very sensitive to initialisation and the generated activity graph revealed little about the true structure of the shopping activity. Taking into account the number of parameters to be estimated for the MOHMM and the size of the training set, it is obvious that the MOHMM suffers from severe over-fitting. Comparing the activity graphs extracted from the PaHMM, CHMM and DML-HMM, it is also clear that the activity graph generated from the DML-HMM was least affected by noise in the event recognition with the cleanest connections showing the most plausible structure of the shopping activity.

Explaining away errors in event recognition— Errors in event recognition can be explained away using the learned activity models. The values of the hidden state variables in the models can be inferred using the extended Viteribi algorithm formulated in Section 3.3.3., which correspond to the event recognition results explained by the learned model. Here we show an example of using different DBNs to explain away errors in event recognition. Figure 10(a) shows the ground truth of event occurrences during an activity from the test set which lasted 140 frames. The detected and classified events contained fair amount of errors as shown in Fig. 10(b). The hidden states of four different models were used to infer (generate) occurrences of events and their classes. Figure 10(d)– (f) show that the event recognition results were improved using the inferred hidden states of the PaHMM, CHMM and DML-HMM. However, more errors were introduced by the MOHMM compared to the event recognition in isolation using GMM (see Fig. 10(c) and (b)). It is expected because the MOHMM activity model was learned poorly due to the insufficient training data. It is also clear from Fig. 10(f) that the result obtained using DML-HMM was the nearest to the ground-truth shown in Fig. 10(a).

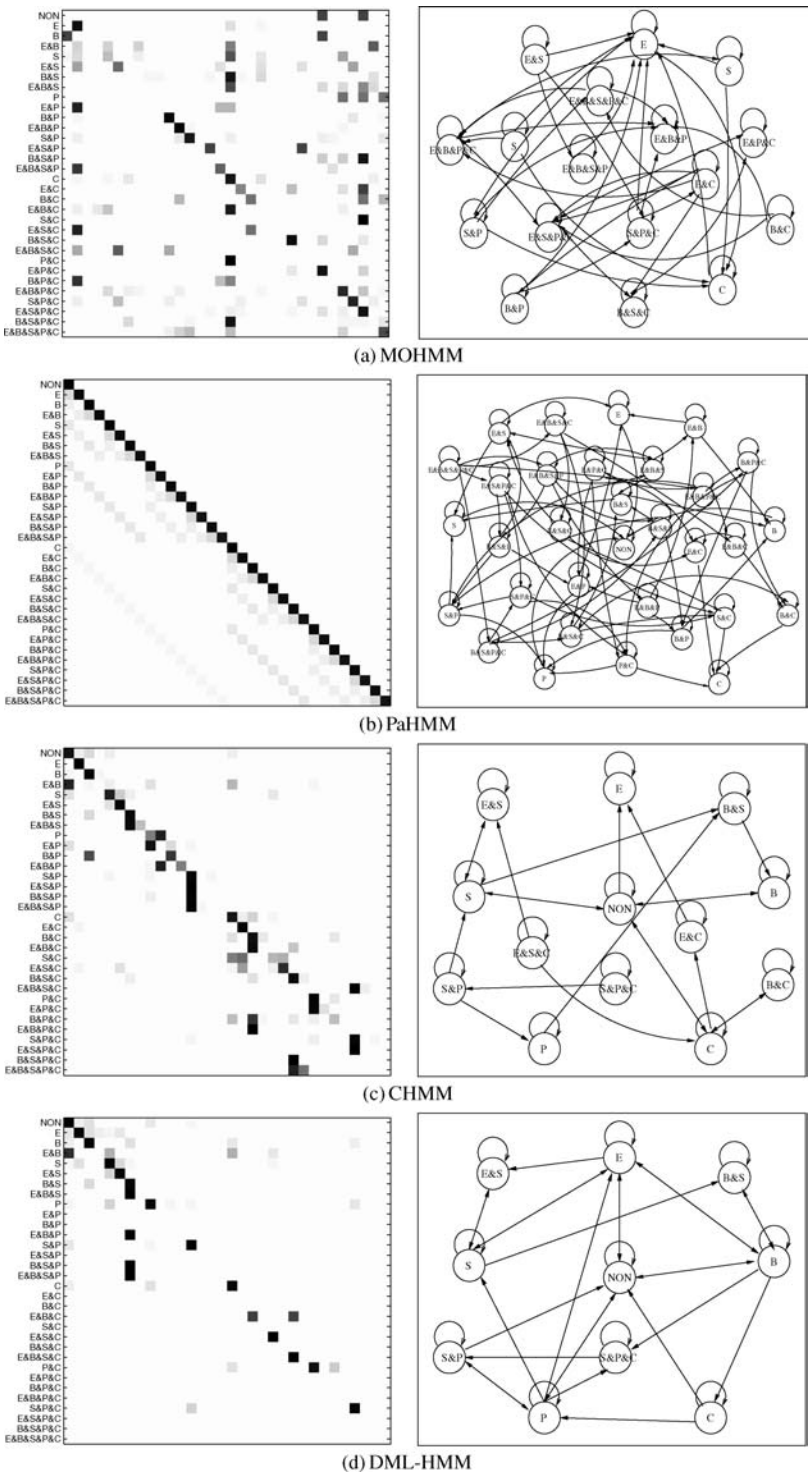
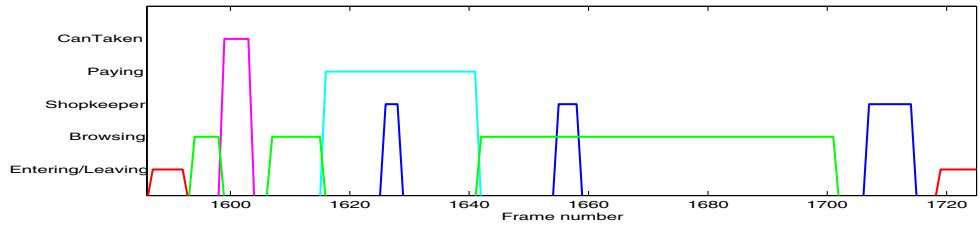
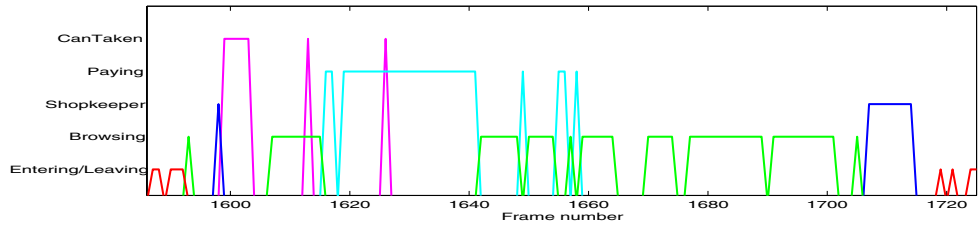


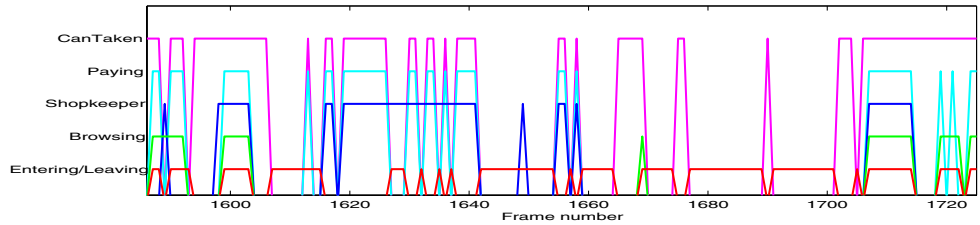
Figure 9. *Left*: Activity transition matrices learned from the training set using four different models. Each entry corresponds to the transition probabilities of two states (black for true and white for false) and each state corresponds to the occurrence of one or more different classes of events. States 'E', 'B', 'S', 'P', 'C' and 'NON' correspond to `entering/leaving`, `browsing`, `shopkeeper`, `paying`, `canTaken` and `no-activity` respectively. State 'B&S' refers to `browsing` and `shopkeeper` occurring simultaneously. *Right*: Activity graphs automatically generated from the activity transition matrices.



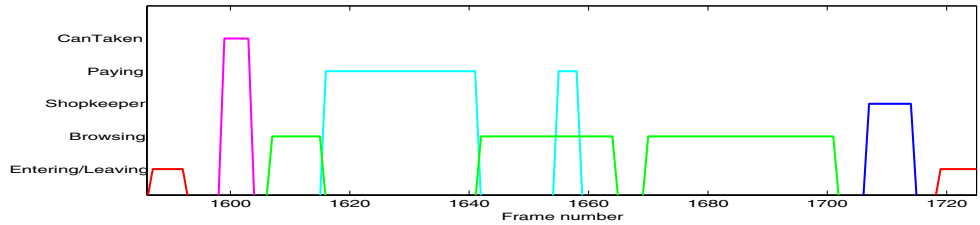
(a) Ground truth



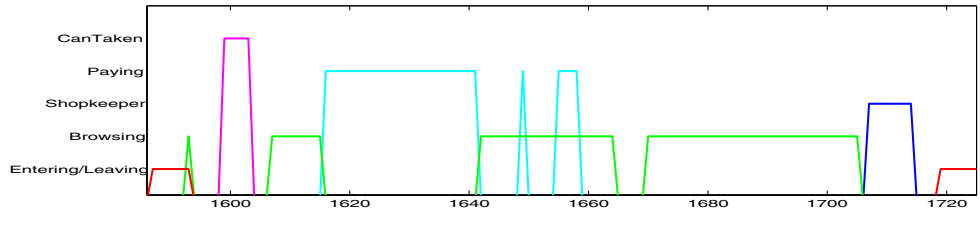
(b) Event recognition in isolation



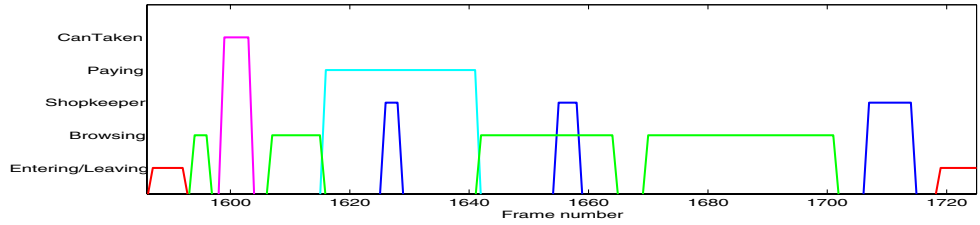
(c) MOHMM



(d) PaHMM



(e) CHMM



(f) DML-HMM

Figure 10. Improving event recognition accuracy in shopping activity modelling using different DBNs.

Figure 11. Typical scenes of aircraft cargo activities under different lighting conditions.

4.2. Modelling Aircraft Cargo Loading/Unloading Activities

4.2.1. Data Set. Let us now consider an aircraft docking scenario which is more challenging than the simulated shopping scenario because it is outdoor with very unstable lighting conditions (see Fig. 11). In particular, typically sequences taken in the early morning contained indistinct objects, reflecting poor lighting, whilst those taken during the midday had strong sunshine causing strong shadows in the scene. Fast moving clouds were common during the daytime, which resulted in very unstable lighting conditions. The camera was more than 50 meters away from where the activities took place, giving low resolution images of the objects concerned (see Fig. 11). Among various activities occurred in the scene, we are particularly interested in the aircraft cargo loading/unloading activities in which trucks, cargo lift and cargo container boxes are moving or being moved purposively to transfer cargoes to and from an docked aircraft on the ground (see Fig. 12(a) for typical scenes). A fixed CCTV analogue camera took continuous recordings over two weeks period. After digitisation, the final video sequences have a frame rate of 2Hz. Each image frame has a size of 320× 240 pixels. Note that it is common for CCTV surveillance videos to have such an extremely low frame rate.

4.2.2. Event Recognition. The parameters used for the GMM background model and salient pixel group estimation were the same as those used for the shop activity modelling (see Section 4.1.2). Four different classes of events were automatically detected using BIC as shown in Fig. 12(c). They were labelled as `movingTruck`, `movingCargo`, `movingCargoLift` and `movingTruckCargo` and illustrated using different colours in Fig. 12. As can be seen in Fig. 12(a) and (d), they corresponded correctly to four key constituents of frontal cargo service activities. The first three events correspond respectively to a truck, a cargo container and a cargo lift moving into a specific locations with particular directions of motion and occupancies in the image space. The last event corresponds to any occurrence of simultaneous movements of the truck and the cargo container when they are overlapped. For comparison, clustering was also performed using AIC and cross-validation when 6 event classes were detected (see 12(c)). Compared with the result obtained using BIC shown in Fig. 12(b), both `movingTruck` and `movingCargo` were clustered into two event classes. These results reinforce the early results shown in both Figs. 3 and 7.

It is noted that the event recognition model makes more mistakes for the aircraft cargo activities compared to that for the indoor shopping activities presented in the preceding section. This is due to the more challenging nature of the scenario in the sense that (1) the lighting condition in the aircraft scene was far less stable, (2) the image resolution of the moving objects in the aircraft scene were lower, and (3) the movements of different objects in the aircraft scene were overlapped a lot more. Similarly, a Dynamically Multi-Linked Hidden Markov Model (DML-HMM) was employed to interpret groups of events in correlation and as a result, to explain away the errors in event recognition.

4.2.3. Activity Modelling Using DML-HMM. For modelling the airport cargo loading/unloading activities with four different classes of events, we exploit a DML-HMM network topology as illustrated in Fig. 13(a). The topology of the DML-HMM were learned from training data using BIC (see Section 3.2). The discovered causal relationships among different classes of events are embodied in the topology of the DML-HMM. Figure 13(c) shows the expected structure for the airport cargo unloading activities. It can be seen that causal relationships among different classes of events and temporal structure of activity have been

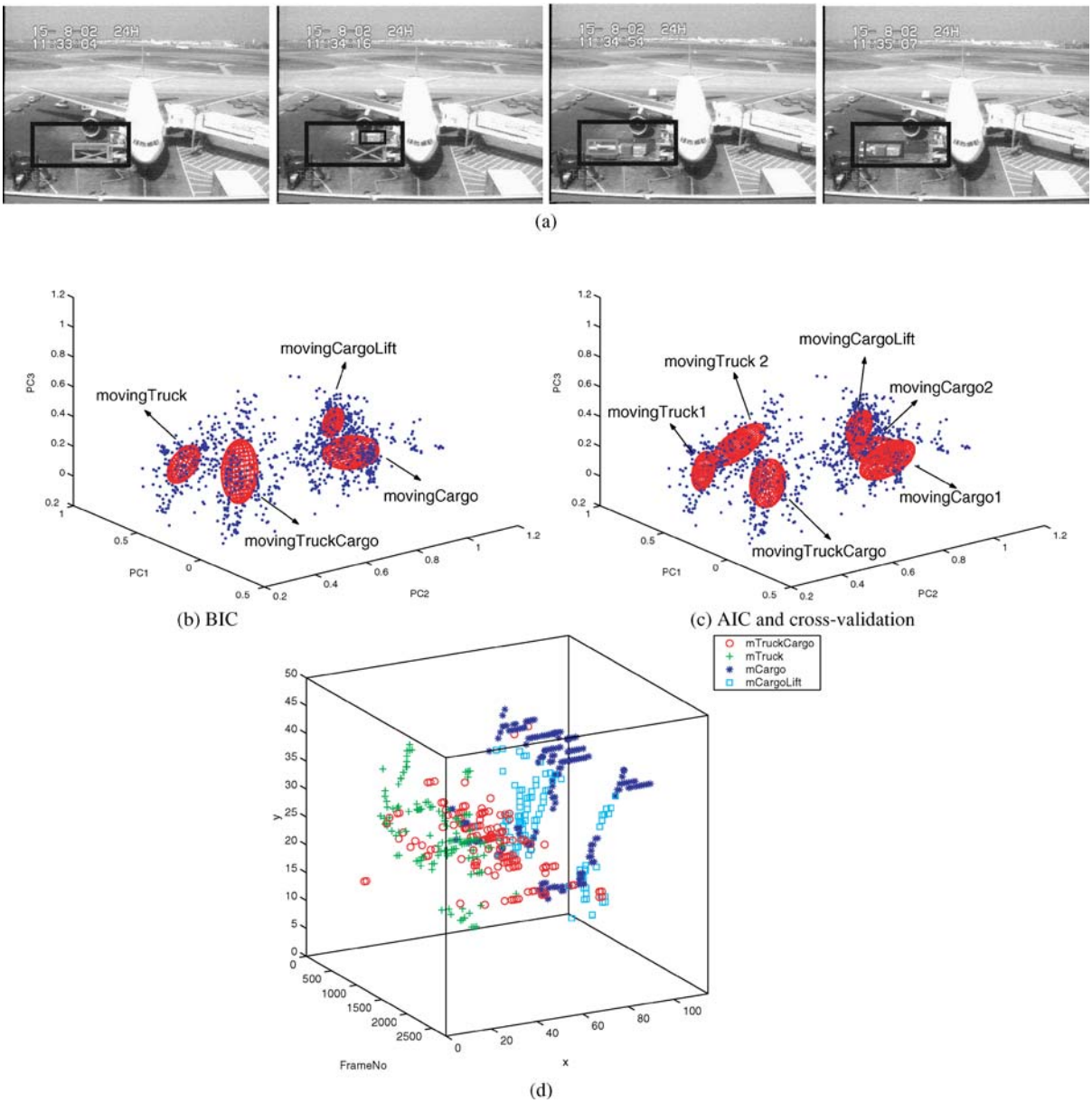


Figure 12. Event detection and classification during aircraft cargo loading/unloading activities. (a) Examples of recognised events in the image frame. Events belonging to different classes are indicated with bounding boxes in different colours. (b) and (c) Unsupervised event clustering of the training set in the 7-dimensional feature space (only the first 3 principal components are shown for illustration) using BIC, AIC and cross-validation respectively. (d) An example of the whereabout and temporal order of the four classes of events being detected throughout an example sequence. Centroids of different classes of events are depicted using different symbols.

discovered correctly. For comparison, Fig. 13(b) shows the topology learned by AIC using the same training data. A more complex topology was chosen and some wrong causal relationships among different event classes were also selected.

4.2.4. Comparing Different DBNs for Activity Modelling. Experiments were conducted on modelling aircraft cargo loading/unloading activities using MOHMM, PaHMM, CHMM, and DML-HMM and testing their comparative performances. Our database

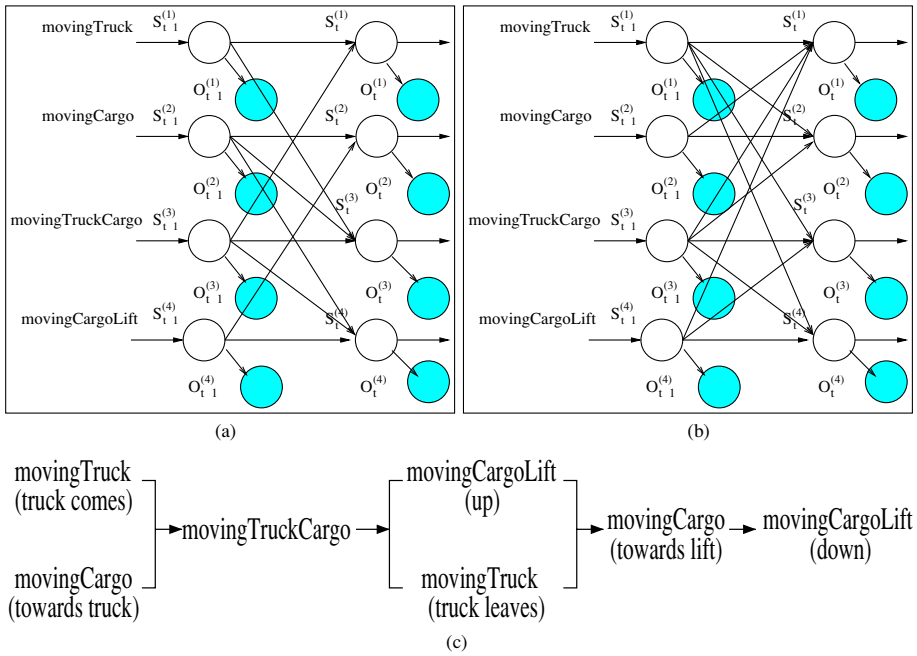


Figure 13. (a) and (b) Two DML-HMMs for modelling 4 temporal processes corresponding to 4 different classes of events detected in aircraft cargo unloading activities. Their structure are determined by BIC and AIC respectively using the same training set. (c) The expected causal and temporal structure of the activity.

for the experiments consists of 23 (9 loading and 14 unloading) continuous activity sequences selected from the 2 weeks recording giving in total 43275 frames of video data that covers different time of different days under changing lighting conditions, from early morning, midday to late afternoons. The length of each sequence was between 829 to 3449 frames at 2 Hz, covering 12–25 minutes video footage. The DML-HMM shown in Fig. 13(a) was adopted. For the PaHMM and CHMM, there were also 4 temporal processes in their topologies. There were 4 observation variables at each time instance in the topology of the MOHMM. The number of parameters to be estimated for the MOHMM, PaHMM, CHMM and DML-HMM were 2511, 300, 412, and 332 respectively. In the following we present results on (1) model training, (2) extracting activity graphs, (3) comparative performance evaluation on activity recognition, and (4) explaining away errors in event recognition.

Model training—Among the 23 sequences, there are 8 clean loading and 8 clean unloading, 1 noisy loading and 6 noisy unloading sequences. By 'clean' we imply that the lighting change in the duration of a sequence is tolerable with limited error in event recognition. We used different combinations of different subsets from the 23 sequences dataset to train the models in order to avoid any bias in the results. We used the remaining subsets for testing. Three different types of model training were conducted as follows.

Case I: Training by small clean sets. We randomly split the 16 clean sequences into 8 small sets for which each set, consisting of one loading and one unloading sequence, is used for training. The other 7 sets were used for testing. Each set has on average 3733 frames with the shortest 3117 and longest 4929. This was repeated 8 times with a different set. Event recognition was performed on each set using both sequences and there were on average 695 events of four different classes automatically detected per training set. These recognised events (represented by 7-D feature vectors) were then used as the observational input for training a DBN. The loading and unloading sequence in each set was used to train separately two sets of model structure and parameters in the training process.

Case II: Training by large clean sets. Each training set now consisted of randomly selected 4 clean loading and 4 clean unloading sequences from the 16 sequences. Each set has on average 14929 frames

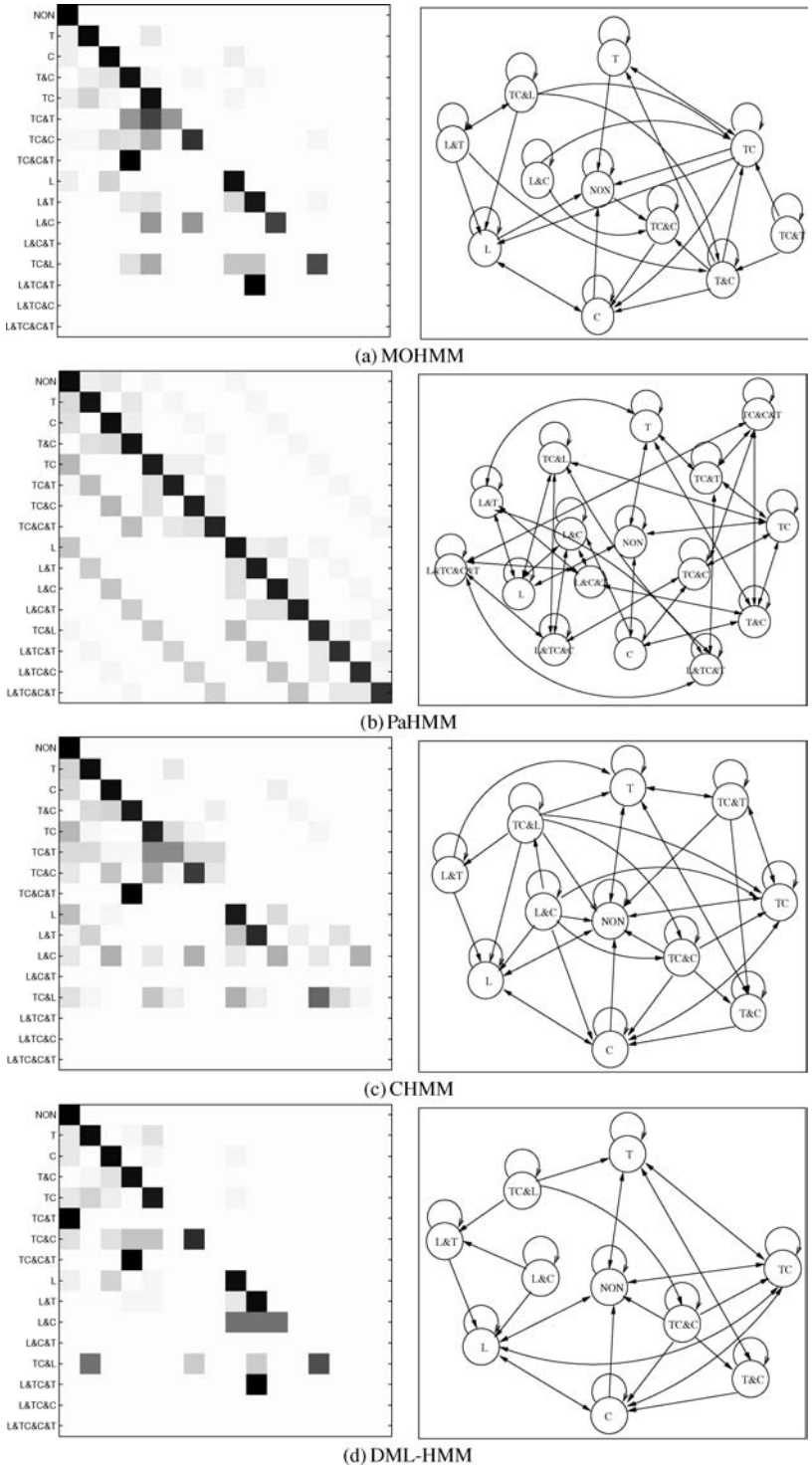
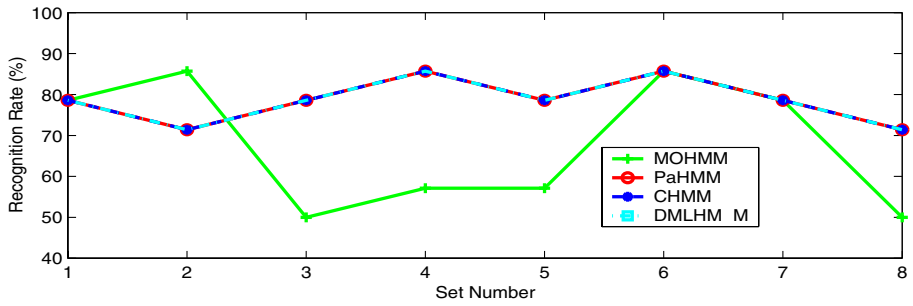
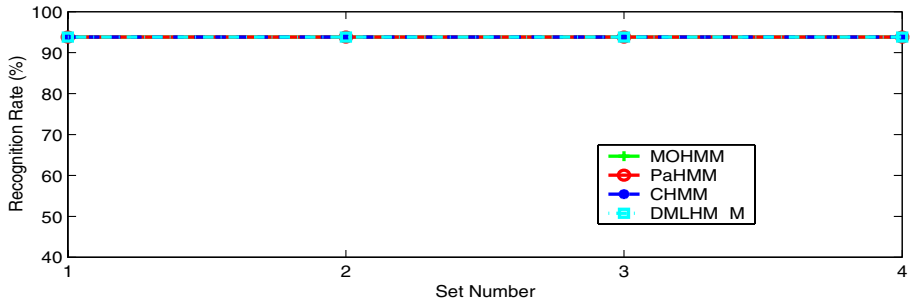


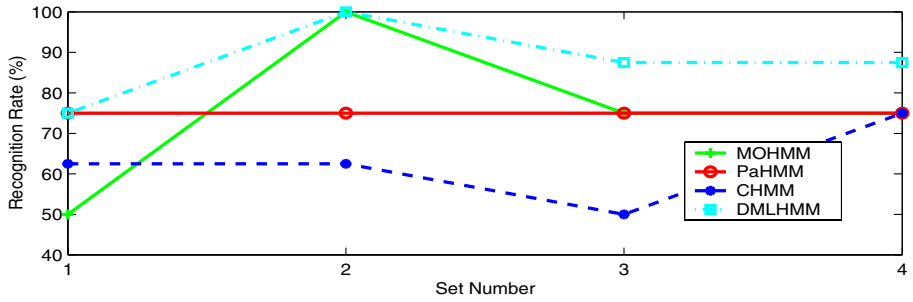
Figure 14. Left: Activity transition matrices learned from a noisy training set using four different models. Each entry corresponds to the transition probabilities of two states (black for true and white for false) and each state corresponds to the occurrence of one or more different classes of events. States 'T', 'C', 'TC', 'L' and 'NON' correspond to `movingTruck`, `movingCargo`, `movingTruckCargo`, `movingCargoLift` and `no-activity` respectively. State 'T&C' refers to `movingTruck` and `movingCargo` occurring simultaneously. Right: Activity graphs automatically generated from the activity transition matrices.



(a) Testing results from models trained by small clean sets



(b) Testing results from models trained by large clean sets



(c) Testing results from models trained by large noisy sets

Figure 15. Activity recognition rates from MOHMM, PaHMM, CHMM and DML-HMM.

with shortest 13637 and longest 16221. The training was repeated as above 4 times. These training and testing were repeated four times.

Case III: Training by large noisy sets. Four training sets were constructed using randomly selected 4 clean loading and 4 clean unloading sequences as above, but this time also included 1 noisy loading and 6 noisy unloading sequences in each set. Each set has on average 28346 frames with shortest 27054 and longest 29638. The training was repeated 4 times again.

Activity graphs—Figure 14 shows four different activity graphs automatically generated from the trained model state transition matrices of MOHMM, PaHMM, CHMM and DML-HMM. They were trained using a large noisy dataset from one of the *Case III* training sets above. From these activity graphs, important stages of activities are shown to be discovered by the models. Although the state transition matrices were initialised randomly with no constraint on their transitions, the learned activity transition matrices have sparse structures. It is also clear that among the four, the activity graph generated by the DML-HMM was least affected by noise with the cleanest connections showing the best factorised state space.

Activity recognition—The above trained four different types of models were tested for activity recognition. The models trained using each small clean set were tested for activity recognition on the remaining 7 sets. The models trained using each of the large clean sets and each of the noisy sets were tested on the

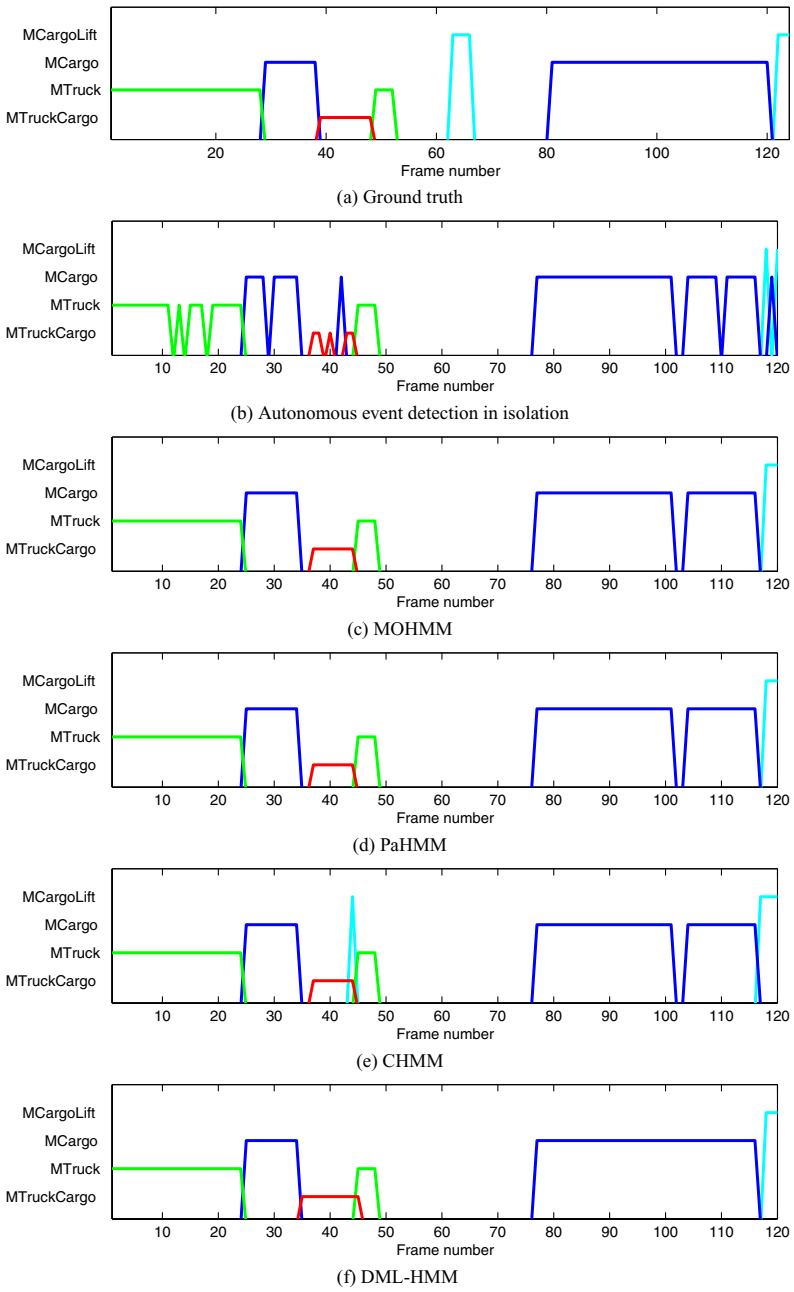


Figure 16. Improving event recognition accuracy for aircraft cargo activities using different DBNs.

remaining sets. Figure 15 shows comparative testing results. As expected when small data sets were used for training, PaHMM, CHMM and DML-HMM achieved higher average recognition rate over the 7 testing sets (79%) than that of MOHMM (68%) (Fig. 15(a)). This is due to that the latter's large number of parameters were poorly estimated without enough data. Given sufficiently large sets of clean data for training, all the models were able to give a fairly high and similar average recognition rate over the 4 testing sets at about 94% (Fig. 15(b)). However, if noisy data were used, the average recognition rate over the 4 testing sets of MOHMM (75%), PaHMM (75%) and in particular CHMM (62%) dropped sig-

nificantly compared to that of DML-HMM (88%) (Fig. 15(c)).

Explaining away errors in event recognition—Here we show an example of using different DBNs to explain away errors in event recognition. Figure 16(a) shows the ground truth of event occurrences for a cargo unloading activity unit from the test set which lasted 124 frames. The events recognised using GMM contained fair amount of errors as shown in Fig. 16(b). The hidden states of four different DBNs were used to infer (generate) occurrences of events and their classes. Figure16(c)–(f) show that the event recognition results were improved using the inferred hidden states of the DBNs. The result from the DML-HMM was the nearest to the ground-truth shown in Fig. 16(a).

5. Conclusion

In this paper, we have presented a unified automatic model selection based approach for modelling complex activities of multiple objects in cluttered scenes. Adopting a data-driven probabilistic model, both the structure and parameters of the model are learned in an unsupervised manner from data. In particular, object-independent events are detected and classified by unsupervised clustering using Expectation-Maximisation (EM) and classified using automatic model order selection based on Schwarz's Bayesian Information Criterion (BIC). We developed a DML-HMM model to discover the temporal and causal correlations among discrete events for robust and holistic scene-level behaviour interpretation. A Dynamically Multi-Linked Hidden Markov Model (DML-HMM) is built using BIC based factorisation resulting in its topology being intrinsically determined by the underlying causality and temporal order among different events. Extensive experiments were conducted on modelling activities captured in different scenarios. Our experimental results demonstrated that the performance of a DML-HMM on modelling group activities in a noisy and cluttered scene is superior compared to those of other comparable Dynamic Probabilistic Networks (DPNs) including a Multi-Observation Hidden Markov Model (MOHMM), a Parallel Hidden Markov Model (PaHMM) and a Coupled Hidden Markov Model (CHMM). Comparative results on using BIC, AIC and cross-validation for event recognition and DML-HMM topology discovery were also presented.

The main limitation of the proposed activity modelling method is that large amount of training data are required. The model thus may not be able to scale well for very complex activities. One possible solution is to utilise model prioris derived from learnt context knowledge to improve the learning efficiency of our activity model given limited data. It would also be worthwhile to further investigate whether the structure and parameters of a DML-HMM activity model can be adaptive to instant changes in the underlying behaviours of objects. This can be achieved by adopting an incremental learning and inference algorithm. Our future work will also be focused on developing a hierarchical DBN topology in order to model the underlying temporal processes of groups of different activities at the scene level.

Appendix A: Derivation of the Bayesian Information Criterion (BIC)

Bayesian Information Criterion (BIC) can be derived as an approximation of the Bayesian Model Selection (BMS). Given a data set $\mathbf{O}$ and a set of K competing model $\mathbf{m}_k$ where $k \in \{1, \ldots, K\}$, BMS chooses the best model according to the Maximum A Posteriori (MAP) principle. Specifically, the model chosen by BMS maximises $P(\mathbf{m}_k \mid \mathbf{O})$, the a posteriori probability of observing $\mathbf{O}$ given model m_k:

$$\hat{\mathbf{m}}_k = \arg\max_{\mathbf{m}_k} \{P(\mathbf{m}_k \mid \mathbf{O})\}$$

Using Bayes' rule, the posteriori probability can be written as:

$$P(\mathbf{m}_k \mid \mathbf{O}) = \frac{P(\mathbf{O} \mid \mathbf{m}_k) P(\mathbf{m}_k)}{\sum_{k=1}^{K} P(\mathbf{O} \mid \mathbf{m}_k) P(\mathbf{m}_k)} \tag{17}$$

where $P(\mathbf{O} \mid \mathbf{m}_k)$ is the marginal probability (likelihood) of the data and $P(\mathbf{m}_k)$ is the *a priori* probability of model $\mathbf{m}_k$. For a parameterised model $\mathbf{m}_k$, the marginal probability can be computed as:

$$P(\mathbf{O} \mid \mathbf{m}_k) = \int P(\mathbf{O} \mid \mathbf{m}_k, \boldsymbol{\theta}_{\mathbf{m}_k}) P(\boldsymbol{\theta}_{\mathbf{m}_k} \mid \mathbf{m}_k) d\boldsymbol{\theta}_{\mathbf{m}_k} \tag{18}$$

where $\theta \mathbf{m}_k$ is a vector of a dimensionality D_k describing the parameter under $\mathbf{m}_k$, $P(\boldsymbol{\theta}_{\mathbf{m}_k} \mid \mathbf{m}_k)$ is the *a priori* probabilistic density function of θ given $\mathbf{m}_k$ and $P(\mathbf{O} \mid \mathbf{m}_k, \boldsymbol{\theta}_{\mathbf{m}_k})$ is the probability density function of $\mathbf{O}$ given $\mathbf{m}_k$ and θm_k.

If there is no *a priori* knowledge that favours any of the candidate models, the Bayesian Model Selection method selects the model that yields the maximum marginal probability. The analytic evaluation of the integral in Eq. (18) is only possible for exponential family distributions. For more general cases, an asymptotic approximation method needs to be used. Here, the Laplace approximation is adopted to compute the marginal probability $P(\mathbf{O} \mid \mathbf{m}_k)$ (see Schwarz, 1978; for details), giving:

$$\begin{aligned}\log P(\mathbf{O} \mid \mathbf{m}_k) &= \log P(\mathbf{O} \mid \mathbf{m}_k, \hat{\boldsymbol{\theta}}_{\mathbf{m}_k}) \\ &+ \log P(\hat{\boldsymbol{\theta}}_{\mathbf{m}_k} \mid \mathbf{m}_k) + \frac{D_k}{2}\log(2\pi) - \frac{D_k}{2}\log N \\ &- \frac{1}{2}\log|\mathbf{i}| + O\left(N^{-\frac{1}{2}}\right) \end{aligned} \tag{19}$$

where D_k is the dimensionality of the parameter space, N is the sample size, $\hat{\boldsymbol{\theta}}_{\mathbf{m}_k}$ is the ML estimate of $\theta\mathbf{m}_k$, i is the expected Fisher information matrix for one observation (Raftery, 1995), and $O(N^{-\frac{1}{2}})$ represents any quantity such that $N^{-\frac{1}{2}}O(N^{-\frac{1}{2}})$ approaches a constant value as N approaches infinity. The first term on the right-hand side of Eq. (19) is of order $O(\log N)$, the fourth term is of order $O(N)$, while all the other terms are of order $O(1)$ or less. BIC is derived as the negative of $\log P(\mathbf{O} \mid \mathbf{m}_k)$ with those order $O(1)$ or less terms being eliminated:

$$BIC = -\log P(\mathbf{O} \mid \mathbf{m}_k, \hat{\boldsymbol{\theta}}_{\mathbf{m}_k}) + \frac{D_k}{2}\log N \tag{20}$$

Acknowledgements

We shall thank Huw Farmer and Mark Ealing at BAA for providing us with the aircraft cargo activity data under the DTI/EPSRC MI LINK project ICONS.

Notes

1. Similar to the MHI (see Bobick and Davis, 2001), PCH implicitly represents the direction of movement. First order moments based on PCH value distribution within the bounding box is thus capable of measuring the direction of movement quantitively.
2. BBNs are also known as Bayesian Networks, Belief Networks or Directed Acyclic Graphical (DAG) Models. They are special cases of graphical models which combine probability theory and graph theory to address two important issues in data modelling: uncertainty and complexity.

References

Akaike, H. 1973. Information theory and an extension of the maximum likelihood principle. In *2nd International Symposium on Information Theory*, pp. 267–281.

Babaguchi, N., Kawai, Y., and Kitahashi, T. 2002. Event based indexing of broadcasting sports video by intermodal collaboration. *IEEE Transactions on Multimedia*, 4(1):68–75.

Baum, L.E. and Petrie, T. 1996. Statistical inference for probabilistic functions of finite state markov chains. *Ann. Math. Stat.*, 37:1554–1563.

Biernacki, C., Celeux, G., and Govaert, G. 2000. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):719–725.

Bishop, C. 1995. *Neural Networks for Pattern Recognition*. Cambridge University Press.

Bobick, A. and Wilson, A. 1997. A state-based approach to the representation and recognition of gesture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(12):1325–1337.

Bobick, A.F. and Davis, J.W. 2001. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267.

Brand, M. and Kettnaker, V. 2000. Discovery and segmentation of activities in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):844–851.

Brand, M., Oliver, N., and Pentland, A. 1996. Coupled hidden markov models for complex action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, pp. 994–999.

Bregler, C. 1997. Learning and recognizing human dynamics in video sequences. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 568–575.

Buxton, H. and Gong, S. 1995. Visual surveillance in a dynamic and uncertain world. *Artificial Intelligence*, 78:431–459.

Chomat, O., Martin, J., and Crowley, J. 2000. A probabilistic sensor for the perception and the recognition of activities. In *European Conference on Computer Vision*, pp. 487–503.

Figueiredo, M. and Jain, A.K. 2002. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):381–396.

Forney, G.D. 1973. The viterbi algorithm. *Proceedings of the IEEE*, 61:268–278.

Friedman, N., Murphy, K., and Russell, S. 1998. Learning the structure of dynamic probabilistic networks. In *Uncertainty in AI*, pp. 139–147.

Gath, I. and Geva, B. 1989. Unsupervised optimal fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):773–781.

Ghahramani, Z. 1998. Learning dynamic bayesian networks. In *Adaptive Processing of Sequences and Data Structures. Lecture Notes in AI*, pp. 168–197.

Gong, S. and Buxton, H. 1992. On the visual expectations of moving objects: A probabilistic approach with augmented hidden markov models. In *European Conference on Artificial Intelligence*, Vienna, pp. 781–786.

Gong, S., Ng, J., and Sherrah, J. 2002. On the semantics of visual behaviour, structured events and trajectories of human action. *Image Vision Computing*, 20(12):873–888.

Gong, S., Walter, M., and Psarrou, A. 1999. Recognition of temporal structures: Learning prior and propagating observation augmented densities via hidden markov states. In *IEEE International Conference on Computer Vision*, Corfu, pp. 157–162.

Gong, S. and Xiang, T. 2003. Recognition of group activities using dynamic probabilistic networks. In *IEEE International Conference on Computer Vision*, pp. 742–749.

Greenspan, H., Goldberger, J., and Mayer, A. 2004. Probabilistic space-time video modelling via piecewise GMM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):384–396.

Haritaoglu, I., Harwood, D., and Davis, L.S. 2000. w^4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):809–830.

Heckerman, D. 1995. A tutorial on learning with Bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research.

Hongeng, S. and Nevatia, R. 2001. Multi-agent event recognition. In *IEEE International Conference on Computer Vision*, pp. 80–86.

Hung, H. and Gong, S. 2004. Quantifying temporal saliency. In *British Machine Vision Conference*, pp. 727–736.

Intille, S. and Bobick, A. 1998. Representation and visual recognition of complex multi-agent actions using Belief networks. In *ECCV Workshop on Perception of Human Action*, Freiburg, Germany.

Intille, S., Davis, J., and Bobick, A. 1997. Real-time closed-world tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 697–703.

Johnson, N., Galata, A., and Hogg, D. 1998. The acquisition and use of interaction behaviour models. In *IEEE Conference on Computer Vision and Pattern Recognition*, Santa Barbara, USA, pp. 866–871.

Kass, R. and Raftery, A. 1995. Bayes factors. *Journal of the American Statistical Association*, 90:377–395.

Rabiner, L.R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

McKenna, S., Jabri, S., Duric, Z., Rosenfeld, A., and Wechsler, H. 2000. Tracking group of people. *Computer Vision and Image Understanding*, 80:42–56.

Mclachlan, G. and Peel, D. 1997. *Finite Mixture Models*. John Wiley & Sons.

Medioni, G., Cohen, I., Bremond, F., Hongeng, S., and Nevatia, R. 2001. Event detection and analysis from video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):873–889.

Ng, J. and Gong, S. 2001. Learning pixel-wise signal energy for understanding semantics. In *British Machine Vision Conference*, pp. 695–704.

Oliver, N., Rosario, B., and Pentland, A. 2000. A bayesian computer vision system for modelling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):831–843.

Pavlovic, V., Rehg, J.M., Cham, T., and Murphy, K.P. 1999. A dynamic bayesian network approach to figure tracking using learned dynamic models. In *IEEE International Conference on Computer Vision*, pp. 94–101.

Piater, J.H. and Crowley, J.L. 2001. Multi-modal tracking of interacting targets using gaussian approximation. In *Proceedings of 2nd IEEE Workshop on Performance Evaluation of Tracking and Surveillance*, pp. 141–147.

Raftery, A. 1995. Bayes model selection in social research. *Sociological Methodology*, 90:181–196.

Rao, C., Yilmaz, A., and Shah, M. 2002. View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 50:203–226.

Rissanen, J. 1989. *Stochastic Complexity in Statistical Inquiry*. World Scentific.

Roberts, S. 1997. Parametric and non-parametric unsupervised cluster analysis. *Pattern Recognition*, 30(2):261–272.

Roberts, S., Husmeier, D., Rezek, I., and Penny, W. 1998. Bayesian approaches to Gaussian mixture modelling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1133–1142.

Schwarz, G. 1978. Estimating the dimension of a model. *Annals of Statistics*, 6:461–464.

Sherrah, J. and Gong, S. 2000. VIGOUR: A system for tracking and recognition of multiple people and their activities. In *International Conference on Pattern Recognition*, Barcelona, pp. 179–182.

Sherrah, J. and Gong, S. 2001. Automated detection of localised visual events over varying temporal scales. In *Proc. European Workshop on Advanced Video-based Surveillance System*.

Smyth, P. 2000. Model selection for probabilistic clustering using cross-validated likelihood. *Statistics and Computing*, 10:63–72.

Stauffer, C. and Grimson, W. 2000. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–758.

Vogler, C. and Metaxas, D. 2001. A framework for recognizing the simultaneous aspects of american sign language. *Computer Vision and Image Understanding*, 81:358–384.

Wada, T. and Matsuyama, T. 2000. Multiobject behavior recognition by event driven selective attention method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):873–887.

Xiang, T. and Gong, S. 2003. Discovering bayesian causality among visual events in a complex outdoor scene. In *IEEE International Conference on Advanced Video- and Signal-based Surveillance*, pp. 177–182.

Xiang, T., Gong, S., and Parkinson, D. 2002. Autonomous visual events detection and classification without explicit object-centred segmentation and tracking. In *British Machine Vision Conference*, pp. 233–242.