Activity understanding and unusual event detection in surveillance videos

Loy, Chen Change

Activity Understanding and Unusual Event Detection in Surveillance Videos

Chen Change Loy

Submitted to the University of London in partial fulfilment of the requirements for the degree of Doctor of Philosophy

Queen Mary University of London

2010

Abstract

Computer scientists have made ceaseless efforts to replicate cognitive video understanding abilities of human brains onto autonomous vision systems. As video surveillance cameras become ubiquitous, there is a surge in studies on automated activity understanding and unusual event detection in surveillance videos. Nevertheless, video content analysis in public scenes remained a formidable challenge due to intrinsic difficulties such as severe inter-object occlusion in crowded scene and poor quality of recorded surveillance footage. Moreover, it is nontrivial to achieve robust detection of unusual events, which are rare, ambiguous, and easily confused with noise. This thesis proposes solutions for resolving ambiguous visual observations and overcoming unreliability of conventional activity analysis methods by exploiting multi-camera visual context and human feedback.

The thesis first demonstrates the importance of learning visual context for establishing reliable reasoning on observed activity in a camera network. In the proposed approach, a new Cross Canonical Correlation Analysis (xCCA) is formulated to discover and quantify time delayed pairwise correlations of regional activities observed within and across multiple camera views. This thesis shows that learning time delayed pairwise activity correlations offers valuable contextual information for (1) spatial and temporal topology inference of a camera network, (2) robust person re-identification, and (3) accurate activity-based video temporal segmentation. Crucially, in contrast to conventional methods, the proposed approach does not rely on either intra-camera or inter-camera object tracking; it can thus be applied to low-quality surveillance videos featuring severe inter-object occlusions.

Second, to detect global unusual event across multiple disjoint cameras, this thesis extends visual context learning from pairwise relationship to global time delayed dependency between regional activities. Specifically, a Time Delayed Probabilistic Graphical Model (TD-PGM) is proposed to model the multi-camera activities and their dependencies. Subtle global unusual events are detected and localised using the model as context-incoherent patterns across multiple camera views. In the model, different nodes represent activities in different decomposed re-

gions from different camera views, and the directed links between nodes encoding time delayed dependencies between activities observed within and across camera views. In order to learn optimised time delayed dependencies in a TD-PGM, a novel two-stage structure learning approach is formulated by combining both constraint-based and scored-searching based structure learning methods.

Third, to cope with visual context changes over time, this two-stage structure learning approach is extended to permit tractable incremental update of both TD-PGM parameters and its structure. As opposed to most existing studies that assume static model once learned, the proposed incremental learning allows a model to adapt itself to reflect the changes in the current visual context, such as subtle behaviour drift over time or removal/addition of cameras. Importantly, the incremental structure learning is achieved without either exhaustive search in a large graph structure space or storing all past observations in memory, making the proposed solution memory and time efficient.

Forth, an active learning approach is presented to incorporate human feedback for on-line unusual event detection. Contrary to most existing unsupervised methods that perform passive mining for unusual events, the proposed approach automatically requests supervision for critical points to resolve ambiguities of interest, leading to more robust detection of subtle unusual events. The active learning strategy is formulated as a stream-based solution, *i.e*. it makes decision on-the-fly on whether to request label for each unlabelled sample observed in sequence. It selects adaptively two active learning criteria, namely likelihood criterion and uncertainty criterion to achieve (1) discovery of unknown event classes and (2) refinement of classification boundary.

The effectiveness of the proposed approaches is validated using videos captured from busy public scenes such as underground stations and traffic intersections.

Declaration

I hereby declare that this thesis has been composed by myself and that it describes my own work. It has not been submitted, either in the same or different form, to this or any other university for a degree. All verbatim extracts are distinguished by quotation marks, and all sources of information have been acknowledged.

Some parts of the work have previously been published as:

- **Chapter 2**
  - C. C. Loy, T. Xiang and S. Gong, "Detecting and Discriminating Behavioural Anomalies," in Pattern Recognition, vol. 44(1), pp. 117-132, 2011
  - C. C. Loy, T. Xiang and S. Gong, "From Local Temporal Correlation to Global Anomaly Detection," in International Workshop on Machine Learning for Vision-based Motion Analysis (MLVMA), 2008
- **Chapter 3**
  - C. C. Loy, T. Xiang and S. Gong, "Time-Delayed Correlation Analysis for Multi-Camera Activity Understanding," in International Journal of Computer Vision (IJCV), vol. 90(1), pp. 106-129, 2010
  - C. C. Loy, T. Xiang and S. Gong, "Multi-Camera Activity Correlation Analysis," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1988-1995, 2009
- **Chapter 4**
  - C. C. Loy, T. Xiang and S. Gong, "Modelling Activity Global Temporal Dependencies using Time Delayed Probabilistic Graphical Model," in International Conference on Computer Vision (ICCV), pp. 120-127 ,2009

- **Chapter 5**
  - C. C. Loy, T. Xiang and S. Gong, "Incremental Activity Modelling in Multiple Disjoint Cameras," submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2010
- **Chapter 6**
  - C. C. Loy, T. Xiang and S. Gong, "Stream-based Active Unusual Event Detection," in Asian Conference on Computer Vision (ACCV), 2010
  - C. C. Loy, T. Xiang and S. Gong, "Modelling Multi-object Activity by Gaussian Processes," in Proc. of the British Machine Vision Conference (BMVC), 2009

Acknowledgements

Firstly, I would like to thank my supervisor Dr. Tao Xiang for his perpetual patience, encouragement and guidance. I doubt I could have had a more capable or accommodating advisor, and he has my deepest gratitude. Besides, I am deeply grateful for invaluable advice and consistent support from my co-supervisor Professor Shaogang Gong throughout the project.

I would like to thank Dr. Ioannis (Yiannis) Patras for being my internal examiner throughout my PhD project. My warm appreciation goes to various students and associates at Vision Group for their friendship and support, in particular Jian Li, Tim Hospedales, Emanuel Zelniker, Wei-Shi Zheng, Somboon Hongeng, Tom Haines, Samuel Pachoud, Matteo Bregonzio, Marco Paladini, Khalid Bashir, Bryan Prosser, Javier Orozco, Parthipan Siva, Ke Chen, Yogesh Raja, Lukasz Zalewski, David Russell and Milan Verma.

I am grateful to all friendly and highly competent administrative and systems support staff in the department for enabling things to run smoothly and efficiently.

I also would like to convey my special thanks to Professor Lim Chee Peng and Dr. Lai Weng Kin who encourage me to take up the challenge of PhD study.

I am indebted to my family in particular my mother Lee Kum Mui, my loving wife Wong Yuen Fei, and my sister Loy Cheng Ying for their enduring love, support and understanding. Without them, I would never have completed my study. Finally, I would like to dedicate this work to my father Loy Yoke Soo, though he could not witness the completion of this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Human eyes are highly efficient devices for scanning through large quantity of low-level visual sensory data and delivering selective information to one's brain for high-level semantic interpretation and gaining situational awareness. Over the last few decades, the computer vision community has endeavoured to bring about similar perceptual capabilities to artificial visual sensors. Substantial efforts have been made towards understanding static images of individual objects and the corresponding processes in the human visual system. This endeavour is intensified further by the need for understanding massive quantity of video data, with the aim to comprehend multiple entities not only within a single image but also over time across multiple video frames for understanding their spatiotemporal relations. A significant application of video analysis and understanding is intelligent surveillance, which aims to automatically interpret human activity and detect unusual events that could pose a threat to public security and safety.

1.1 Activity Understanding and Unusual Event Detection

There has been an accelerated expansion of Closed-Circuit TeleVision (CCTV) surveillance in recent years, largely in response to rising anxieties about crime and its threat to security and safety [168]. Substantial numbers of surveillance cameras have been deployed in public spaces ranging from transport infrastructures (*e.g.* airports, underground stations), shopping centres, sport arenas to residential streets, serving as a tool for crime reduction and risk management. Conventional video surveillance systems rely heavily on human operators to monitor activities and determine the actions to be taken upon occurrence of an incident, *e.g.* tracking suspicious

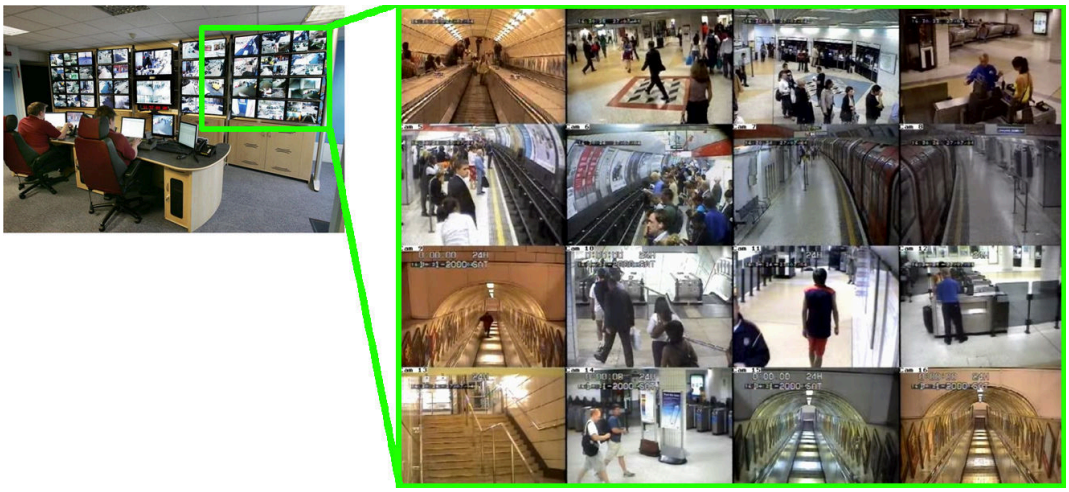


Figure 1.1: A typical CCTV control room monitoring multiple camera views. A report by Home Office [85] reveals that an operator may be required to oversee 90 cameras or more at one time. Consequently, surveillance systems are used primarily in a reactive mode, *i.e.* operators often rely on indication from external agencies to direct their surveillance, or view the recorded footage retrospectively.

target from one camera to another camera or alerting relevant agencies to areas of concern.

Unfortunately, many actionable incidents are simply miss-detected in such a manual system due to inherent limitations from deploying solely human operators eyeballing CCTV screens [85]. Miss-detections could be caused by (1) excessive number of video screens to monitor (see Figure 1.1), (2) boredom and tiredness due to prolonged monitoring, (3) lack of *a priori* and readily accessible knowledge for what to look for, and (4) distraction by additional responsibilities (*e.g.* tape management). In fact, many studies have shown the limits of human-based surveillance. For instance, a study by the United States Sandia National Laboratories shows that the attention of most people falls below an acceptable level after only 20 minutes of monitoring video surveillance screens [90]. This could be linked to the short attention span of humans [156]. Recent studies in cognitive psychology [234–236] also suggest that human tends to make mistake in detection when performing visual search on rare targets with low prevalence (*e.g.* searching a knife during baggage screening) due to the inherent difficulty in the task as well as the immense emotional pressure. Due to the aforementioned factors, surveillance footage is often used merely as passive records or as evidence for post-event investigation.

Miss-detection of important events can be perilous in critical surveillance tasks such as border control or airport surveillance. Technology providers and end-users recognise that manual process alone is inadequate to meet the need for screening timely and searching exhaustively colossal amount of video data, which are generated from the growing number of cameras in pub-

lic spaces. To fulfil such a need, video content analysis paradigm is shifting from a fully human operator model to a machine-assisted and automated model.

1.1.1 Automated Video Content Analysis

There is a surge in demand in the last few years for automated video analysis technologies, also known as *video analytics*. This trend is persisting[1,2], mainly driven by government initiatives with strong demands from retail and transportation sectors[3]. Increasing number of CCTV solutions are made available with some degree of automated analytic capabilities by suppliers from large-scale system integrators to subject matter expert (SME) software developers including IBM, Bosch, Pelco, GE Security, Honeywell, Siemens, ObjectVideo, DVTel, Aimetis, Sony, Panasonic, Nice, Vidient, March Network, Mate, and BRS Labs [77, 88].

Security has been the dominant driver for the development and deployment of video analytics solutions. Some common applications are intruder detection, unattended object detection, loitering detection, tailgating detection, and crowd management. These applications provide some practical and useful solutions. Nevertheless, their effectiveness and success depends largely on rather stringent operational conditions in carefully controlled environments. In fact, there are growing concerns on the viability of using such analytics in real-world scenarios especially in unconstrained public spaces. In particular, existing technologies for video content analysis largely rely on Video Motion Detection (VMD), hard-wired rules, and object-centred reasoning in isolation (*i.e.* object segmentation and tracking) with little if any context modelling. Such systems often suffer considerably high false alarm rate due to changes in visual context, such as weather conditions and gradual behaviour drift over time. Fully automated analysis of video data captured from public spaces is often intrinsically ill-conditioned due to large (and unknown) variations in video image quality, resolution, imaging noise, diversity of pose and appearance, and severe occlusion in crowded public scenes. As a result, those systems that rely on hard-wired hypotheses and location-specific rules are likely to break down unexpectedly giving frequent false alarms, requiring elaborative re-configuration and careful parameter tuning by specialists, mak-

[1]Frost and Sullivan estimates that the video surveillance software market will reach $670.7 million annually by 2011 [76].

[2]The growing interest on video analytics is also evident from various industrial focus conferences such as the IMS Video Content Analysis Conferences (`http://www.imsconferences.com`).

[3]Research conducted by the British Industry Security Association demonstrated that video analytics technologies are deployed by the transport and retail sectors most frequently - `http://www.bsia.co.uk/aboutbsia/cctv/O5E926740891`.

ing system deployment non-scalable and hugely expensive. In the worst-case scenario, installed expensive video analytics systems are abandoned or otherwise infrequently used due to excessive operational burden and intolerable level of false alarms.

Addressing the aforementioned problems demands more sophisticated computer vision algorithms. Various automated visual surveillance solutions [59, 107, 157] that are based on more advanced computer vision techniques are being explored recently, such as multi-object context-aware activity understanding [140], intent profiling [18, 172], crowded scene analysis [4, 231, 239], and multi-camera surveillance and cooperative monitoring [68, 116, 149, 217]. One of the most critical functionalities of such an automation solution is to monitor and *understand activity* captured in the video and *detect unusual events* that could pose a threat to public security and safety.

1.1.2 What is Activity Understanding?

Different terms including *action* and *event* are used in the literature when referring to *activity*. To disambiguate these terms, a uniform terminology is proposed, which will be used consistently throughout the remainder of this thesis.

This thesis follows similar taxonomies proposed by Xiang & Gong [240], Poppe [182], and Moeslund *et al*. [157]. In particular, *action* refers to a sequence of primitive movements carried out by a single object, such as human walking or vehicle turning right. *Activity*, on the other hand, contains a number of sequential actions, most likely involving multiple objects that interact or co-exist in a shared common space monitored by single or multiple cameras. Examples of *activity* include 'passengers walking on a train platform and sitting down on a bench' and 'vehicle stopping after turning right at a traffic intersection'.

Given multiple camera views, activity can be further categorised into global activity and regional activity. A regional activity refers to an activity that takes place locally in a single region of a camera view. For instance, in a tube station (Figure 1.1), passengers walking next to a train track or sitting on benches on a platform are regional activities. A global activity, on the other hand, is defined as an activity that involves correlated partial observations of multiple regional activities across multiple cameras. For example, a global activity of train departure may involve co-existing activities taking place at different regions such as the movements of a train at a track area of a platform, passengers moving towards the exits of the platform, and passengers leaving the station via escalators. This thesis does not attempt to explicitly differentiate *activity*

from *event*, as these terms are often used in the literature interchangeably.

Consequently, *activity understanding* is defined as the establishment of high-level interpretation from visual information, not only on single-object activity but also multi-object activities in a scene possibly covered by multiple cameras. This thesis refers activity understanding as general activity analysis tasks that do not involve detecting unusual event in a scene. Examples of such a task include inference of spatial and temporal connection across camera views from visual information or segmentation of activity patterns in videos.

1.1.3 What is an Unusual Event?

Unusual event is another term that has been causing much confusion in the literature. The notion of *unusual event* appears in different names including abnormal, rare, atypical, interesting, suspicious, surprising event, or simply anomaly, abnormality, irregularity and outlier.

Several studies have attempted to define or distinguish these terms. Xiang and Gong [244] define an unusual event as an atypical/abnormal behaviour pattern that is not represented by sufficient samples in a training dataset but satisfies constraint of abnormal pattern. Similarly, Hamid *et al*. [93] refer unusual events to those that are rare and dissimilar from regular instances. Breitenstein *et al*.'s definition [29] of unusual event is more clear-cut - only events that have not been observed before are considered unusual events; those that have been seen at least once are considered rare but not necessarily unusual.

As can be seen, no consensus on defining the term 'unusual event' has been reached so far due to subjective viewpoints of different studies, but there are commonalities. In particular, an event is considered worth further examination by a human operator or should trigger an alarm if:

1. The event seldom occurs or has not been observed before, *i.e*. having low statistical representation in a dataset.

2. The event is unknown or unpredictable.

Consequently, this thesis defines an 'unusual event' as an activity that possesses these common attributes.

1.1.4 Objectives

The principal goal of this work is to devise computer vision algorithms for *activity understanding and unusual event detection in public scene surveillance videos*. Specifically, the work focuses

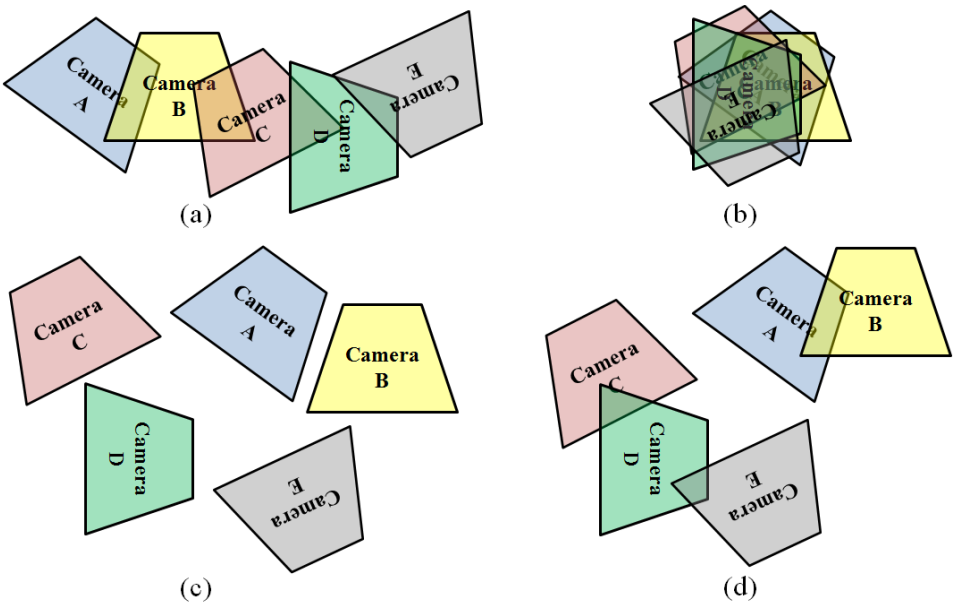


Figure 1.2: An illustration of different degrees of overlap between the field of view (FOV) of five cameras: (a) cameras partially overlap with adjacent cameras, (b) all cameras partially overlap with each other, (c) non-overlapping camera network, (d) the most common case involving different types of overlapping. The gap between two camera views is commonly known as 'blind area', in which a target object may be temporarily out of view of any camera.

on two problems: (1) activity analysis in public scenes monitored by multiple disjoint cameras with non-overlapping views and (2) detection of unusual events in crowded public scene. The following section discusses the underlying challenges to achieve the aforementioned objectives, which serve as motivations of this work.

1.2 Challenges and Motivations

1.2.1 Multiple Camera Activity Analysis in Public Scene

A multiple-camera system is commonly deployed compared to a single-camera system in complex public scenes. In comparison to a single-camera system, a multiple-camera system is capable of covering a wider area of a complex scene. Importantly, it has the potential of providing a complete record of an object's activity in a complex scene, allowing a global interpretation of the object's underlying behaviour. In most multiple-camera surveillance systems, disjoint cameras with non-overlapping field of view (FOV) are more prevalent, due to the desire to maximise spatial coverage in a wide-area scene whilst minimising the deployment cost (see Figure 1.2 for an illustration of different degrees of overlap between FOV). In general, global activity analysis across multiple disjoint cameras in the public scene is often hampered by a number of issues as follows.

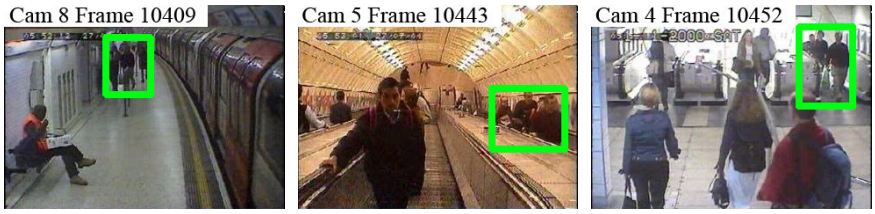


Figure 1.3: Partial observations of activities observed from different camera views: a group of people (highlighted in green boxes) get off a train [Cam 8, frame 10409] and subsequently take an upward escalator [Cam 5, frame 10443], which leads them to the escalator exit view [Cam 4, frame 10452]. Note that the same objects exhibit drastic appearance variations due to changes in illumination (both intra and inter-camera), camera orientation, and pose changes.

Inter-camera visual variations

Objects moving across camera views often experience drastic variations in their visual appearances owing to different illumination conditions, camera orientations and changes in object pose (see Figure 1.3). All these factors increase the uncertainties in activity understanding.

Unknown and arbitrary inter-camera gaps

Apart from inevitable visual variations across views, unknown and arbitrary inter-camera gap between disjoint cameras is another inextricable factor that leads to uncertainty in activity understanding. In particular, the unknown and often large separation of cameras in space causes temporal discontinuity in visual observations, *i.e.* a global activity can only be observed partially in different views whilst portions of the activity sequence may be unobserved due to the inter-camera gaps. To further complicate the matter, two widely separated camera views may include arbitrary number of entry/exit locations in the gap, where existing objects can disappear and new objects can appear, causing uncertainty in understanding and correlating activities in both camera views.

Owing to the temporal discontinuity in visual observations, a global unusual event, of which partial observations can span across multiple views, may look perfectly normal in isolated camera views. For instance, consider two cameras that monitor road junctions A and B that are one mile apart - a vehicle passing A will typically appear at B in 2 minutes, and it is normal to observe either large or small volume of traffics in either views. However, if a large volume of traffic is observed at junction A, but two minutes later only few vehicles can be seen in junction B, it is possible that a road accident has occurred between junctions A and B. Individual inspection on each view would fail to detect such an unusual event since local activities appear perfectly normal when observed in individual camera views. This problem renders a direct implementation

of single-view approaches [104, 131, 134, 244] infeasible since global behaviour interpretation in disjoint cameras is not achievable based solely on visual evidence captured within a single view.

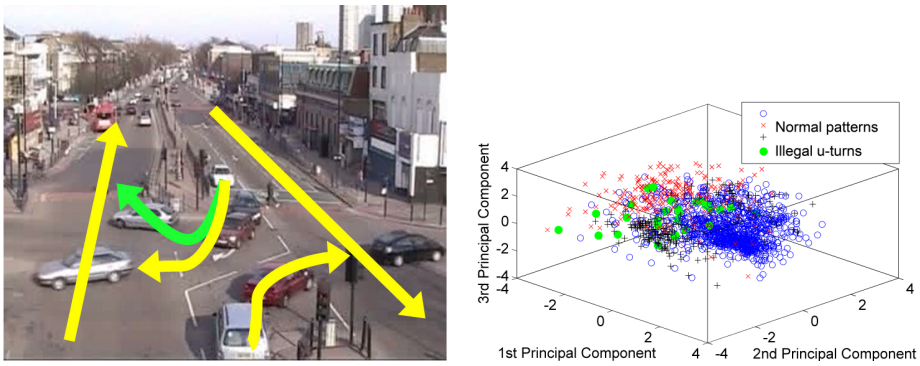
Figure 1.4: Object tracking in low-quality video and crowded scenes is very challenging due to significant visual ambiguities: these figures show three consecutive frames captured from two different cameras at 0.7 fps. An object can pass through the whole view in just three frames. In addition, severe inter-object occlusion and low-quality video are among the key factors that render object tracking infeasible.

Low-quality videos captured in crowded scene

To understand activity among disjoint cameras with non-overlapping FOV, an obvious solution seems to be tracking objects within and across camera views. Indeed, most previous methods rely on either intra-camera (within camera) tracking to detect entry and exit events for modelling inter-camera (between cameras) transition time distribution, or inter-camera tracking for object/trajectory association [149, 217, 233, 251]. These methods generally assume reliable object localisation and small spatial displacement in moving objects between consecutive frames. However, these assumptions are often invalid in real-world surveillance settings characterised by severe occlusions due to an excessive number of objects in the scene. In particular, in a typical public scene as shown in Figure 1.4, the sheer number of objects cause severe and continuous inter-object occlusions, leading to temporal discontinuity of trajectories. Tracking is further compounded by the typically low temporal and spatial resolutions of surveillance video[4], where large spatial displacement is observed in moving objects between consecutive frames.

Due to the aforementioned challenges, visual observations are inevitably noisy and partial, making the meaning of activity ambiguous. This motivates the use of visual context. Activities in a public space are inherently *context-aware*, exhibited through constraints imposed by

[4]Many surveillance systems record videos at less than 5 frames per second (fps) or compromise image resolution to optimise data bandwidth and storage space [48, 133].



(a) (b)

Figure 1.5: An example of illegal u-turn event (Figure 1.5(a)). It is subtle due to its visual similarity with a large number of co-occurring normal patterns in a scene. This can be observed from the plot in a principal component analysis space (Fig .1.5(b)), where similar u-turn cases (plotted as green dots) are partially overlapped with other normal patterns.

scene layout and the correlated activities of other objects both in the same camera view and other views. Given multiple disjoint cameras and low-quality public scene surveillance videos, the key to activity understanding and unusual event detection lie on how well one can learn a global *visual context* to associate partial observations of activities observed across camera views. The global visual context encompasses spatial and temporal context defining where and when a partial observation occurs, as well as correlation context specifying the expectation inferred from the correlated behaviours of other objects in the camera network. The issues caused by visual variations, temporal discontinuity, and low-quality video can be remedied with the contextual knowledge, leading to more robust and accurate video content analysis.

In fact, strong psychophysical evidence [11,20,174,195] suggests that visual contexts, which encompass relations of an object with its surroundings, are crucial for establishing a clear comprehension of a scene. Motivated by these studies, a number of computer vision problems have been addressed by exploiting visual context clues, such as object detection [98,220,257], action recognition [151], and tracking [5,28,164,169,238,246,256]. However, there have been limited studies that utilise visual context information for activity analysis in multiple non-overlapping crowded views captured by low-quality video data (see Section 2.3 for a more detailed discussion).

1.2.2 Unusual Event Detection in Public Scene

One of the foremost challenges in unusual event detection is the highly imbalanced distribution between normal event and unusual event classes, *i.e.* most of the samples corresponding to normal

event classes whilst the remaining unusual event classes only constitute a small percentage of the entire dataset. In addition, normal patterns are often known a-priori, whilst the unusual events are unforeseeable. The rarity and unpredictability of unusual events implies that collecting sufficient training data for supervised learning will often be impractical.

Consequently, most unusual event detection methods [104, 131, 155, 231] employ an outlier detection strategy, in which a model is trained using normal events through unsupervised one-class learning. Events that deviate statistically from the resulting normal profile are deemed unusual. This strategy offers a practical way of bypassing the problems of imbalanced class distribution and inadequate unusual event training samples. However, the outlier detection strategy is subject to a few inextricable limitations when it is used for unusual event detection in unconstrained public scenes:

1. Difficulty in detecting subtle unusual events. Subtle unusual events often refer to events that involve small and local changes [25] in behaviour or those that are embedded in temporal correlations among multiple objects [244]. In a busy public scene, feature distribution of an unusual event may partially overlap with normal events, making it similar visually to a large number of normally behaving objects co-existing in a scene (see Figure 1.5 for example). As a result, it is hard to spot these subtle unusual events using pure data-driven approaches.

2. No subsequent exploitation of flagged unusual events. Surveillance video often contains noise that is visually similar to possible genuine unusual events of interest. If there is no subsequent exploitation of flagged unusual events, it is computationally difficult to distinguish a genuine unusual event from a vast amount of noisy observations.

3. Large amount of uninteresting outliers causing false alarms. Normal behaviour patterns in a public scene are complicated and highly diverse. Hence, preparation of a complete set of well-defined normal data for off-line learning becomes infeasible. Defining a model that encompasses all normal events is inherently difficult if not impossible given limited data. Hence, some outlying regions of a normal class may be falsely and consistently detected as being unusual if no human feedback is taken into account for arbitrating such false alarms.

To overcome the aforementioned issues of unsupervised learning, other sources of information need to be exploited. *Human feedback* is a rich source of information that can be utilised

to assist in resolving ambiguities during class decision boundary formation. In essence, computers lack many of the fundamental visual capabilities that humans have, especially in learning and abstracting from examples [33], or detecting salient patterns or unseen classes robustly from noisy sensory data [112]. Consequently, occasional human inputs would be extremely useful for guiding the creation of an activity model for unusual event detection, by assigning an exact event class label or confirming a detection result, when the activity model encounters difficulty in distinguishing an equivocal or subtle unusual activity. Such a feedback could strengthen the decision boundary of activity classes on what is normal/unusual, leading to more robust detection of inconspicuous and unknown unusual events.

Although human feedback has been exploited in various computer vision tasks[5], there has been limited usage on unusual event detection problem (see Section 2.4.4 for discussion). It is worth noting that a random query for feedback is inefficient since fundamentally not all samples are critical for fine-tuning the section of the decision boundary that is "confused" by visual ambiguities and lack of training data [121]. In fact, a method is required to choose selectively the most critical and informative point for feedback to improve the activity model without overwhelming a human annotator with a large numbers of uninteresting queries. This problem is nontrivial and has not been addressed before for real-time unusual event detection in surveillance videos.

1.3 Approaches

1.3.1 Multi-Camera Activity Analysis with Visual Context Learning

As discussed in Section 1.2.1, to establish reliable reasoning on observed activity in a distributed camera network, it is critical and necessary to learn a global *visual context* that defines the spatial, temporal, and dependency relationships between partial observations across camera views. The partial observations of a global activity observed in disjoint cameras are correlated and interdependent in that the observations take place following a certain temporal order with unknown temporal gaps caused by the spatial distances between camera views. In other words, these partial observations often form a chain of inter-correlated spatiotemporal patterns, spanning across different regions in a global space (see Figure 1.3). Given consistent observations on these pat-

[5]The usefulness of human involvement has been proven by a multitude of different computer vision tasks that interleave visual-based learning with human inputs, including object detection [136] that exploits human-specified high-level description of target objects for training; relevance feedback in interactive image retrieval [260]; visual object recognition [125] and activity recognition [227] that rely on human feedback to disambiguate confusing instances between image classes.

terns over time, meaningful visual context can be extracted and represented as temporal delays and dependency (or correlation) strengths among activity patterns observed from multiple camera views.

Learning pairwise correlation

To this end, a new approach is proposed to model time delayed correlations between multi-camera activities without relying on either intra-camera or inter-camera tracking. Specifically, a novel Cross Canonical Correlation Analysis (xCCA) framework is formulated to discover and quantify correlation and temporal relationships between multi-camera regional activities. This thesis shows that pairwise correlation can be employed as a valuable contextual cue to address three fundamental problems in multi-camera activity understanding:

1. Estimating the spatial topology (*i.e.* between-camera spatial relationships) and importantly the temporal topology of a camera network, that is, the temporal relationships (*e.g.* the unknown delay time) between inter-correlated partial observations taking place in different camera views.

2. Facilitating more robust and accurate person re-identification between different camera views, by resolving ambiguities and uncertainties that arise due to large and unknown separation between cameras both spatially and temporally.

3. Performing activity-based temporal segmentation by linking visual evidence collected from different camera views.

Discovering global activity dependency

Although the xCCA is effective in the aforementioned tasks, it is limited to discover linear pairwise relationships without considering multiple dependencies in a global context. The xCCA is thus not suitable for building a graphical model for global unusual event detection in a complex network, where activity can have multiple causes from different views. To address this limitation, this thesis formulates a novel Time Delayed Probabilistic Graphical Model (TD-PGM) to model globally-optimised and nonlinear dependencies between regional activities. Specifically, each node in the TD-PGM represents activities in a decomposed region from each of the camera views, and the directed links between the nodes encode the time delayed dependencies between the regional activities. The time delayed dependencies are discovered and optimised globally

using a novel two-stage structure learning method, which combines constraint-based and scored-searching based methods in a principled manner. With the learned model, context-incoherent global events can be detected effectively through inspecting the consistency between node observation and graph propagation.

Incremental learning on activity dependency

In an unconstrained environment, visual context changes are ineluctable and may occur gradually or abruptly. Specifically, gradual context change may involve subtle behaviour drift over time, *e.g*. different volumes of crowd or traffic flows during daytime and evening. Abrupt changes, on the other hand, occur if there are significant alterations to the physical camera network including removal/addition of camera from/to a camera network or changes in camera angles. Both gradual and abrupt changes cause transitions and modifications of inter-camera relationships over time, which most existing methods would fail to adapt to since they generally perform off-line training and learn an activity profile that remain static once a training phase is completed.

To cater for changes of visual context in dynamic environments, the two-stage structure learning is extended to permit incremental refinement of the model structure for learning new relationships or removing outmoded dependency links to adapt to the current visual context over time. The proposed incremental learning approach is devised to be memory and time tractable, achieved through replacing an obsolete structure with a set of potential candidate structures that are deemed most promising at the current time, and also keeping a handful of sufficient statistics without storing all past observations in memory.

1.3.2 Learning from Human Feedback via Active Learning

As discussed in Section 1.2.2, human feedback is a rich source of information that can be employed to resolve ambiguities and uncertainties that arise during unusual event detection in a public scene. A potential approach to learn from human feedback is by employing an *active learning strategy* [197]. Active learning method is capable of following a set of predefined query criteria to select the most critical and informative point for labelling requests. This active selection capability allows a model to achieve a comparable classification rate with fewer samples compared to passive random labelling strategy. Importantly, it helps in resolving ambiguities of interest, leading to more robust and accurate detection of subtle unusual events.

In this thesis, a new stream-based active learning strategy with adaptive selection of multiple

query criteria is formulated. Below are several key features of the proposed active learning approach:

1. It is formulated as a stream-based approach to ensure real-time response during on-line unusual event detection. Specifically, in contrast to conventional batch mode learning, the model makes immediate decision on whether to request human verification as new video data are streamed in.

2. The proposed approach is capable of performing joint discovery of unknown events (exploration) and classifier learning (exploitation) using multiple query criteria. In particular, some classes, especially unusual event classes have to be discovered since they are not available in the early stage of training. At the same time, it is necessary to improve the model gradually by refining the decision boundary. Thus, different criteria are needed to achieve these goals.

3. Different criteria are selected adaptively for a reliable detection performance. This is important because good active learning criteria are dataset dependent [197]. Importantly, one typically does not know the best suited criterion for a dataset at different phases of learning.

1.3.3 Hypotheses

In summary, the aforementioned approaches are formulated based on these hypotheses:

1. Learning the time delayed correlation and dependency between regional activities could benefit activity understanding and unusual event detection in multiple disjoint cameras with non-overlapping views.

2. Learning from human feedback via active learning could achieve better unusual event detection performance compared to unsupervised learning strategy. It could also yield a comparable event classification performance with fewer labelled samples compared to random sampling strategy. In addition, adaptive selection of different active query criteria could facilitate the joint discovery of unknown events (exploration) and classification boundary refinement (exploitation).

3. Non object-centred representation (*i.e.* without explicit object segmentation and tracking) is useful for activity understanding and unusual event detection in crowded spaces captured by low-quality video data.

1.4 Contributions

The contributions of this thesis to video-based activity understanding and unusual event detection are as follows:

1. A new xCCA framework is proposed to discover and quantify pairwise correlation and temporal relationships among multi-camera regional activities without relying on top-down rules or predefined scene models. As opposed to existing object-centred approaches, the proposed xCCA learns activity correlations by exploiting the underlying spatial and temporal correlations of regional activities in a holistic manner, and it avoids object tracking in crowded spaces captured by low-quality video data [215, 216].

2. The xCCA is limited to pairwise optimisation but not global dependency discovery that is needed for unusual event detection in multiple disjoint cameras. To this end, a novel TD-PGM is formulated to model the global activity dependencies. The dependencies are optimised using a two-stage structure learning method that combines constraint-based and scored-searching based methods. Contrary to single-stage structure learning approach, the two-stage structure learning method allows tractable and reliable dependency learning given a large graph (> 50 nodes). Importantly, this is achieved without any prior knowledge and assumptions on the camera topology [144, 214]. To date there is no reported study on modelling time delayed activity dependencies for real-time detection of subtle and ambiguous global unusual events across distributed multi-camera views of busy public scenes.

3. The two-stage structure learning method is extended to allow on-line refinement of the TD-PGM structure for accommodating changes in the definition of normality/abnormality caused by visual context changes and adapt to new dependency relationships between cameras over time. The learning process is achieved by keeping a handful of sufficient statistics without storing all past observations in memory. This makes the proposed approach computationally and memory efficient, it is therefore suitable for real-time applications [144].

4. A new active learning approach is proposed to incorporate crucial human supervision to resolve ambiguities for more robust and accurate unusual event detection over conventional unsupervised approaches. Unlike existing methods, the proposed approach is capable of selecting the most critical and informative point automatically for labelling request. In

addition, a new adaptive weighting scheme suitable for combining multiple query criteria in a stream-based setting is introduced to provide joint unknown event discovery and classification boundary refinement in real-time [145].

1.5 Outline

This thesis is organised into seven chapters:

Chapter 2 presents a review on various existing strategies and approaches for activity understanding and unusual event detection, while providing further motivations for the proposed approaches of this thesis.

Chapter 3 provides detailed explanations on the xCCA framework. It shows that time delayed pairwise correlations are not only useful for inferring the spatial and temporal topology of a camera network, but also important as contextual information to facilitate more robust and accurate person re-identification and video temporal segmentation.

Chapter 4 presents the TD-PGM for detecting global unusual events in multiple disjoint cameras with unknown and arbitrary time gaps. In particular, the chapter describes a new two-stage structure learning algorithm for learning globally-optimised time delayed dependencies between activities observed across camera views.

Chapter 5 explains the extension of the two-stage structure learning approach for incremental adaptation to visual context changes. Specifically, the chapter describes a way to achieve memory and time tractable incremental update on both parameters and structure of the TD-PGM. Experiments are conducted to evaluate the performance of the algorithm in adapting to gradual and abrupt visual context changes, as well as correcting potential errors in an initial model.

Chapter 6 presents the mechanism of learning from human feedback for on-line unusual event detection through stream-based active learning. Experiments are carried out to demonstrate the effectiveness of the proposed approach in balancing different active query criteria for joint unknown event discovery and decision boundary refinement, and how such an approach can lead to more robust and accurate detection of subtle unusual events in public space.

Chapter 7 provides conclusions and suggests a number of areas to be pursued as further work.

Chapter 2

Literature Review

In general, a system for activity understanding and unusual event detection in surveillance videos involves several key components for processing videos:

1. Low-level components for background modelling, feature extraction, and object tracking.
2. Middle-level components for object and action description, *e.g*. object classification and action classification.
3. High-level components for semantic interpretation, *e.g*. activity understanding and unusual event detection.

Considerable effort has been spent and significant improvements have been achieved in each component over the years [107, 137, 224]. Given the broad and expanding nature of this research area, this review narrows down the scope by focusing only on specific techniques within low and high-level components that are widely adopted in visual surveillance systems. Specifically, this review is structured into four subsections: low-level component on activity representation (Section 2.1) and high-level components on single camera view analysis (Section 2.2), multiple camera view analysis (Section 2.3), and activity learning strategies (Section 2.4).

Some specific techniques such as background subtraction and object tracking are not described thoroughly in this review. A survey written by Hu *et al*. [107] provides detailed descriptions on each of the aforementioned techniques. Whilst Turaga *et al*. [224] and Lavee *et al*. [137] summarise state of the art approaches focusing on modelling methods for understanding action

and activities in videos, Dee and Velastin [59] highlight the challenges in visual surveillance and present a list of different strategies developed for visual surveillance. For multiple camera activity analysis, a complete treatment on the most commonly used techniques can be found in Javed and Shah [117], as well as in Aghajan and Cavallaro [2]. For a general review on active learning strategy beyond visual surveillance, Settles [197] provides a comprehensive coverage on common techniques and the most recent developments in the domain.

2.1 Activity Representation

Before semantic interpretation of activity can be established, the question of activity representation or abstraction [137] must be addressed. It concerns the extraction, selection, and transformation of low-level visual properties in video to construct intermediate input to an activity model. As pointed out by Hongeng *et al*. [102] and Lavee *et al*. [137], activity representation should be expressive enough to describe a variety of activities yet sufficiently discriminative in distinguishing different individual activities.

The selection of activity representation is governed by several factors such as video frame rate, distance of objects from the camera, and scene crowdedness. In the following sections, the ways of selecting an appropriate activity representation are discussed. Advantages and weaknesses of various representations commonly used for activity understanding are also highlighted, including object-based features, pixel-based features, and other feature representations.

2.1.1 Object-based Representation

Object-based representation constructs a set of features for each object in a video based on an assumption that individual objects can be segmented reasonably well for event reasoning. These features include trajectory or blob-level descriptors such as bounding box and shape. Among these features, a trajectory-based feature is commonly employed to represent the motion history of a target in a scene [30, 78, 120, 161, 162, 165, 173, 179, 194, 232, 255].

A trajectory is typically obtained by associating a set of attributes of detected object, such as appearance features and velocity over consecutive frames using motion tracking algorithms [248]. Since these attributes are highly dynamic and vary over time, probabilistic frameworks such as Kalman Filter [124] and Particle Filter [111] are commonly adopted. In addition, processing steps such as moving average smoothing [120, 173], or trajectory merging [78] are commonly

employed to address noise problem or trajectory discontinuity problem to a certain extent.

A discrete vector representation of trajectory can include information such as spatial location, velocity [120], velocity change [173], direction, and temporal curvature. They can be expressed in two-dimensional image space (Figure 2.1), ground plane or three-dimensional space if camera models are available. A single-point trajectory only carries the spatial and motion information of the object as a whole but not the transition of the object's body parts. If video with higher resolution is available, one can extract bounding box or complete contour to describe the shape patterns of a moving object. With the extra information on shape variations, anomalous body-level actions can be detected with a near-field camera [62, 80, 127].

(a) (b)

Figure 2.1: Object-based representation with trajectory feature: (a) Normal trajectory (b) atypical trajectory. ([173])

Earlier studies relying on object-based representation focus on point-to-point-based modelling, *e.g.* learning the distribution of trajectory data [120] and comparing similarity at trajectory point-level [173]. More recent approaches [254] cluster complete trajectories for activity understanding, *e.g.* by using hierarchical clustering [118] and spectral clustering [78, 232].

Object trajectories provide rich spatiotemporal information about an object's activity. Therefore, trajectory information is typically employed to understand object's behaviour in the scene over time. For example, a normal behaviour of passing through a parking lot (Figure 2.1(a)) and an unusual behaviour of wandering about among the parked cars (Figure 2.1(b)), which signify a potential car theft, can be easily distinguished based on the trajectory trace of an object over time.

Nevertheless, employing object-based representation in real-world surveillance settings can be challenging. In general, object tracking assumes reliable object localisation and small spatial displacement between consecutive frames. These assumptions are often invalid due to severe occlusions and low-frame rate surveillance videos. In particular, the sheer number of objects with

(a) (b) (c)

Figure 2.2: Different pixel-based activity representations: (a) pixel change history, (b) optical flow, and (c) particle advection using average optical flow field. ([240], [134], [155])

complex activities causes severe and continuous inter-object occlusions (also known as dynamic occlusions). Simultaneous tracking of multiple objects in this environment is challenging since dynamic occlusions can cause ambiguities on the number and identities of targets, leading to temporal discontinuity in trajectories. Given low-frame rate video, large spatial displacements of the object are observed between consecutive frames, causing severe fragmentation of object trajectories.

Various strategies have been proposed to address the aforementioned problems. For instance, tracking-by-detection approaches [28,169,238] employ object or object part detectors with tracking module to achieve more reliable tracking. Brostow and Cipolla [32] exploit space-time proximity and trajectory coherence of promising image features to perform tracking in a dense crowd. There are also studies that exploit contextual information around a target object to facilitate more robust long-term tracking [246] or tackle the occlusion problem by learning trajectory transition likelihoods in a scene using nonparametric modelling approach [194]. However, reliable solutions for combining or filtering unreliable information sources (false positives of detector), identifying auxiliary contextual cues, or learning trajectory distribution in crowded and unconstrained environments remain elusive.

2.1.2 Pixel-based Representation

Pixel-based representation embodies schemes that extract pixel-level features such as colour, texture, and gradient. Unlike object-based representation, pixel-based abstraction does not group features into blobs or objects.

The most basic pixel-based representation is foreground pixels estimation through background subtraction. Despite its simplicity, Benezeth *et al*. [17] has demonstrated encouraging results in the unusual event detection task by representing activity using both spatial and tempo-

ral distribution of foreground pixels. Evidently, previous studies have already shown the feasibility of this simple representation in human motion recognition [24] using motion history image (MHI) and unusual event detection using pixel change history (PCH) [87] (Figure 2.2(a)) or average behaviour image [119]. Foreground pixel-based representation is attractive due to its computation feasibility. Importantly, it avoids explicit object segmentation and tracking. It is thus applicable for representing activity in crowded scenes where tracking is severely limited intrinsically.

Another common pixel-based representation in activity understanding is optical flow (Figure 2.2(b)) [1, 4, 6, 7, 104, 134, 155, 230, 231, 239, 247]. This pixel-based abstraction typically involves the extraction of motion (direction and speed) of individual pixels between consecutive frames [15]. An image space is usually divided into cells of a specific size (*e.g.*, 10×10), of which the average or median flow field is computed. In some cases, optical flow computed in each cell is quantized into different directions (*e.g.*,4 or 8 directions) [104, 134, 231]. To reduce potential observation noise, flow vectors are normally filtered based on a predefined threshold level. In addition, optical flow information extracted is usually combined with a foreground mask, after which only the vectors caused by foreground objects are considered, whilst all the flow vectors outside the foreground mask are set to zero [6].

Similar to foreground pixel-based representation, optical flow based representation avoids explicit tracking of individual objects. Therefore, it is widely used in highly crowded scenes with extensive clutter and dynamic occlusions. For instance, Andrade *et al.* [6,7] learn the variation in optical flow patterns from human crowds using an hidden Markov model (HMM) to distinguish between normal and unusual behaviours. Ali and Shah [4] overlay a cloud of particles over optical flow fields in crowded scenes, and employ Lagrangian Coherent Structures (LCS) [92] to segment coherent crowd flows. Their subsequent studies [155,239] capture dynamics of the interaction forces in the crowd in addition to optical flow for unusual event detection (Figure 2.2(c)).

In contrast to foreground pixel-based representation, optical flow permits straightforward extraction of motion direction and speed, which are essential for understanding certain types of activity, *e.g.* driving behaviour at traffic intersection. Nevertheless, most optical flow methods assume small object displacement and constant brightness for the computation of velocity field [210], which is invalid for videos with very low frame rate and poor image quality.

Apart from foreground pixel and optical flow based features, promising results have also been

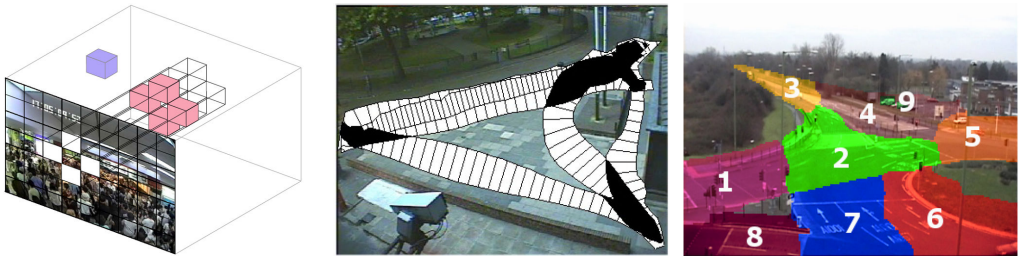
demonstrated in recent studies that utilise image appearance-based features. For instance, Breitenstein *et al*. [29] utilise histogram of oriented gradient (HOG) features [56] to detect unusual events in web-camera videos. In a study by Zelnik-Manor and Irani [249], space time intensity gradients are applied as salient features to a nonparametric distance measure for learning the disparity between activities. In another study, Kratz and Nishino [132] extract spatiotemporal gradients of pixel intensities from video to characterise activities in extremely crowded scene. Mahadevan *et al*. [147] apply mixture of dynamic texture to represent activity patterns. Whilst the aforementioned features offer additional information to represent the salient properties in video, calculating such mixtures of texture or space-time gradients may be computationally expensive. In addition, spatiotemporal gradient extraction would undoubtedly fail given a low frame-rate video due to motion discontinuities.

2.1.3 Other Feature Representations

Instead of using low-level image features, such as location, shape, and motion directly for activity modelling, some studies transform the low-level features to a more compact representation for efficient modelling of complex behaviours. For instance, Xiang and Gong [244] propose an event-based abstraction that represents a behaviour pattern using the probabilities of different classes of event occurring in each video clip. Different types of behaviour patterns are either composed by different classes of events, or having different order of event occurrence. In a relatively close spirit, Kim and Grauman [131] apply a mixture of probabilistic principal component analysers (MPPCA) algorithm [218] to learn a generative model for local optical flow patterns, which offers a compact representation by encoding the optical flow patterns as probabilistic words.

Following the paradigm of object-based representation, there are also attempts to exploit multiple abstract levels of features in a unified framework. For example, Park and Trivedi [175] introduce a framework that switches between trajectory-based features (*e.g*. velocity and position) and blob-based features (*e.g*. aspect ratio of bounding box and height) based on the visual quality of detected objects. Specifically, the proposed algorithm switches to the body-level analysis whenever possible, and switches back to the trajectory-based representation whenever the body-part appearance quality degrades. In another study, Du *et al*. [62] propose to model both trajectory-based features and blob-based features as two stochastic processes with a single directional link between them.

Given object-based, pixel-based, and alternative representations, intuitively when it comes



(a) (b) (c)

Figure 2.3: Different activity decomposition methods: (a) decomposition into local fixed-size spatiotemporal volumes, (b) decomposition into paths (white areas) and junctions (black areas), and (c) decomposition into semantic regions, each of which encapsulates identical activities and differs from those observed in other regions. ([132], [148], [141])

to activity reasoning, a more direct representation such as trajectory and optical flow would appear to offer more expressive power compared to simpler counterparts such as temporal changes of foreground pixels. However, such more elaborated representations would become unreliable under challenging surveillance scenarios with multiple compounding factors, *e.g*. crowded scene and low frame-rate video. Moreover, the computational cost of such representation may be prohibitive for real-time processing [137]. Consequently, quality of CCTV footage and computation requirement becomes the foremost consideration in selecting an appropriate activity representation. In this thesis, foreground pixel-based representation is adopted in Chapters 3 to 5 since the employed datasets are captured from low-quality surveillance videos that feature severe inter-object occlusions.

2.2 Single Camera View Analysis

Most of the studies on activity understanding and unusual event detection are devoted to single camera view. This section discusses common modelling strategies and techniques for single view analysis before we move to multiple camera view analysis in Section 2.3.

2.2.1 Activity Decomposition in Single Camera View Analysis

In single camera view analysis, many existing studies adopt *object-based decomposition* (utilising the object-based activity representation discussed in Section 2.1.1), which consider objects individually in unusual event detection [58, 103, 120, 173]. However, object-based decomposition relies on explicit object segmentation and tracking. Therefore, it is prone to problems associated with occlusion and trajectory discontinuities when applied to a crowded scene, as pointed out by many recent studies [131, 132, 141, 147, 155, 239]. Moreover, as argued in Li *et al*.'s study [141],

due to complexity and enormous uncertainties in a scene, object-based decomposition may not be sufficient to detect unusual events that are visually subtle and ill-defined semantically. In many cases, an event can only be interpreted meaningfully by reasoning both spatially and temporally with other objects in the shared space.

Consequently, more recent studies [104, 131, 132, 155, 231] tackle the activity understanding problem by using *region-based decomposition*, whereby a wide-area scene is segmented into local regions, followed by a global activity modelling to discover and quantify the space-time dependencies between local activities. Without explicit object segmentation, this holistic approach is applicable to dense occluded crowds in contrast to methods relying on object-based decomposition. Different techniques have been proposed following this idea. The most common strategy is to divide a video into local fixed-size spatiotemporal volumes, within which local activity patterns are extracted (Figure 2.3(a)). There are also attempts for exploiting more rigid, top-down structural knowledge in decomposing a scene such as exit/entry zones, routes, junctions and paths [31, 148, 232] (Figure 2.3(b)).

Li *et al*. [141] describe a region-based decomposition approach that decomposes a scene into regions so that each region semantically encapsulates coherent activities, which differ from those observed in other regions (Figure 2.3(c)). They argue that behaviours are inherently context-aware, exhibited through constraints imposed by scene layout and the temporal nature of activities in a given scene. Hence, the method segments a scene without relying on the visual appearance features such as colour, but based on the underlying activity structure. For example, as shown in Figure 2.3(c), different traffic lanes are segmented into different regions given different activity patterns observed over time. The method is shown to be effective in giving semantically meaningful representation on traffic scenes. Nonetheless, it may not be suitable for region decomposition in an extremely crowded scene captured on low-frame rate video, since this method requires feature inputs such as bounding box and optical flow that can hardly be extracted reliably. In addition, it employs a bag of words representation on these features, which ignores the temporal order of activity occurrences that may be important for distinguishing different types of local activity patterns.

The activity decomposition method proposed in this thesis (see Section 3.1.1) is similar to that of Li *et al*. [141] but with clear differences on how activities are represented. In particular, the proposed approach represents local activity patterns as time series of foreground pixel-based

features; it is thus applicable to low-quality surveillance videos featuring severe inter-object occlusions. It also takes into account the temporal correlation between activity patterns during segmentation, thus more discriminative than the bag of words representation proposed in Li *et al*. [141].

2.2.2 Activity Modelling and Unusual Event Detection

After selecting appropriate activity representation and decomposition strategies, the next step is activity modelling. There is a great diversity of approaches to activity modelling in single camera view including:

- Probabilistic graphical models (PGMs), *e.g*. Bayesian networks [35,110], dynamic Bayesian networks (DBNs) [63, 64, 86, 240, 244], propagation nets [202].
- Probabilistic topic models (PTMs), *e.g*. probabilistic latent semantic analysis (pLSA) model [140], latent Dirichlet allocation (LDA) model [104, 155], hierarchical Dirichlet processes (HDP) model [134, 231].
- Petri nets [3].
- Syntactic approaches, *e.g*. context-free grammars [26], stochastic context-free grammars [113].
- Rule-based approaches [58, 154, 201, 221].

In what follows, the scope of this discussion is limited to some widely-used graphical models that have been applied on complex activity modelling in crowded public scenes. Detailed reviews on other approaches such as Petri nets, neural networks, synthetic approaches, and rule-based approaches can be found in a survey by Turaga *et al*. [224].

Bayesian Networks (BN)

A Bayesian network or belief network is a directed acyclic graphical model with nodes representing variables of interest (*e.g*. the occurrence of an event) and the links encoding dependencies among the variables. The strength of a dependency is parameterised by conditional probabilities that are attached to each cluster of parent-child nodes in the network [176].

Bayesian network has been a popular tool for activity modelling due to its powerful capabilities in representing and reasoning uncertain visual observations, as well as its computational feasibility. For instance, Buxton and Gong [35] use BNs in a traffic surveillance application

to capture dependencies between the scene layout and low-level features obtained from motion segmentation and tracking. In another study on modelling multi agent interactions, Intille and Bobick [110] apply BNs for probabilistic representation and recognition of individual agent goals from visual evidence.

A dynamic Bayesian network (DBN) extends BN by incorporating temporal dependencies between random variables. Hidden Markov model (HMM), the simplest DBN with one hidden state variable and one observation variable at each time instance, has been extensively used for activity modelling and recognition. To model more complex activities, various topological extensions to the standard HMM have also been developed, which factorise the state space and/or observation space by introducing multiple hidden state variables and observation state variables, *e.g*. coupled hidden Markov model (CHMM) for modelling interactions between temporal processes [27], parallel hidden Markov model (PaHMM) for learning independent processes of sign language [228], and dynamically multi-linked hidden Markov model (DML-HMM) for interpreting group activities [86]. There are also several attempts to embed hierarchical behaviour structure in the model topology. Examples include hierarchical hidden Markov model (HHMM) [165] and switching hidden semi-Markov model (S-HSMM) [64], in which the state space is decomposed into multiple levels of states according to the hierarchical structure of behaviour.

The cascade structure of DBNs has been considered for activity analysis [170,253]. Oliver *et al*. propose a layered hidden Markov model (LHMM) to capture different levels of temporal details when recognising human activity [170]. The LHMM is essentially a cascade of HMMs, in which each HMM accepts observation vectors processed with different time scales. Zhang *et al*. [253] present a similar framework based on LHMM with each stage of the cascade being employed to learn different levels of actions exhibited from individual to group of people.

Dynamic Bayesian Networks have demonstrated good performance in modelling temporal and dependencies of complex behaviours. However, learning a DBN with multiple temporal processes is intractable and often require large numbers of training data or extensive hand-tuning by human experts. For instance, exact inference on a CHMM [27, 171] beyond two chains (each chain corresponds to one object) is likely to be computationally intractable [171]. The computational problem limits the applicability of DBNs in large scale settings such as activity modelling in a multi-camera network.

Probabilistic Topic Models (PTMs)

Probabilistic topic models (PTMs) such as probabilistic latent semantic analysis (pLSA) [101] and latent Dirichlet allocation (LDA) [23] have recently been applied to activity analysis and unusual event detection.Probabilistic topic models are traditionally used for text mining [22] to discover topics from text documents according to co-occurrence of words. The PTMs are essentially bag of words models that perform clustering by concurrency, *i.e.* a topic is a cluster of co-occurring words. In activity modelling, local visual events and video clips are often treated analogously as 'words' and 'documents' in document analysis. Each video clip may be viewed as a mixture of various 'topics' that represent events [231]. Typically, a video sequence is divided into a sequence of short clips (documents), each of which is represented by words that are constructed from extracted features accumulated over a temporal window. A trained PTM can then be applied to evaluate normality of each local visual event (*i.e.* word) whilst considering interactions (*i.e.* topic) between them.

Probabilistic topic models have found wide applications in single view analysis especially in crowded scene activity modelling. For instance, pLSA [101] is applied by Li *et al.* [140] for analysing events observed at busy traffic intersection. Mehran *et al.* [155] employ LDA [23] to model human interactions within crowds for unusual event detection. With pLSA and LDA, only co-occurring words can be found. Wang *et al.* [231] extend these models by formulating two hierarchical PTMs to model higher-level behaviour interactions by finding both co-occurring words (topics) and co-occurring topics (interactions).

In general, PTMs are less demanding computationally and less sensitive to noise in comparison to DBNs due to the bag of words representation. This advantage, however, requires a compromise of not imposing explicit temporal dependencies between local activities. To address this shortcoming, Hospedales *et al.* [104] propose a Markov clustering topic model (MCTM) to capture the temporal dependencies by introducing a Markov chain on top of a LDA model. Subsequently, Kuettel *et al.* [134] point out that utilising a single Markov chain to learn temporal characteristic of a complex scene is still insufficient. To that end, they employ dependent Dirichlet processes to learn an arbitrary number of infinite hidden Markov models (iHMMs) [14], which can model multiple global temporal rules.

In summary, PTMs have shown superior robustness to inevitable noise in visual features extracted from crowded scene. However, most of the approaches based on PTMs are infeasible

for incremental learning due to its high training cost. Moreover, as pointed out by Hospedales *et al*. [104], using the PTMs entails an appropriate choice of temporal window extent for collecting the bag of words, and a decision regarding whether to overlap the window, and by how much. In particular, a large window would risk in overwhelming behaviours of shorter duration, and small windows risk breaking up behaviours arbitrarily [104].

Other graphical models

Apart from BNs and PTMs, different graphical models have been proposed for activity modelling. For instance, Propagation nets [202, 203], a subset of DBNs with the ability to explicitly model temporal interval durations, have been employed to capture the duration of temporal sub-intervals of multiple parallel streams of events. In another study, Hamid *et al*. [94] attempt to obtain a better activity class discovery performance by using a suffix tree [153] to extract variable lengths of event-subsequence of an activity. Kim and Grauman [131] propose to detect unusual event in video by using a space-time Markov Random Field (MRF), an undirected graphical model with nodes corresponding to a grid of local regions, which neighbouring nodes in both space and time are associated with links.

Most of the aforementioned techniques measure abnormality of an activity based on statistical deviation with respect to a learned normal profile (learning strategies are discussed in Section 2.4), *e.g*. using log-likelihood $p(\mathbf{x}|\Theta)$ of a newly observed event $\mathbf{x}$ given a trained model with parameters Θ. Alternatively, if there exists a model trained with unusual events, one can perform a likelihood ratio test (LRT) [17, 64, 243], which examines the likelihood ratio between two hypotheses on whether an event $\mathbf{x}$ belongs to normal or unusual model. As is common in unusual event detection, it is often necessary to set a threshold in order to make a binary decision on the abnormality of an activity. To assess the performance of an algorithm, the threshold is varied to examine the false positive rate and true positive rate produced given a test set [244].

Single camera view analysis typically assumes that a meaningful interpretation of behaviour can be achieved within isolated camera view. In some cases, this assumption may not be true since only partial visual information can be observed in each camera view. To infer a coherent global understanding of behaviour, a complete record of an object's behaviour traced across multiple camera views is needed. Interpreting behaviour of an object in a global context is intrinsically more challenging than local analysis within a single camera view, due to greater uncertainties caused by visual discontinuity and visual variations (see Section 1.2.1). In the following

Figure 2.4: Multiple cameras are often used to monitor a wide area with complex scene structure. To perform global activity analysis and unusual event detection, it is often necessary to infer the camera topology, *i.e.* spatial and temporal connections between cameras. An example of camera topology is shown on the right-column of the figure, with the number following the camera order.

section, different solutions to the problem of multiple camera view analysis are discussed.

2.3 Multiple Camera View Analysis

The preceding section has been focused on activity understanding and unusual event detection in single-view scenario. In this section, existing approaches developed for distributed camera networks are discussed, including methods for inter-camera relationships learning, global activity modelling, and unusual event detection. Existing graphical model learning methods applicable to multiple camera view activity learning are reviewed at the end of this section.

2.3.1 Learning Inter-Camera Relationships

To perform multi-camera surveillance, it is necessary to first establish the relationship between different camera views. In the context of visual surveillance, the problem of learning inter-camera relationship manifests itself most frequently in the literature as *observation correspondence problem* [115, 116, 208] and *topology inference* [149, 217] (Figure 2.4). The former emphasises more on associating cross-camera trajectory correspondences. Topology inference on the other hand aims to infer a scene model or camera topology of a network. Camera topology typically refers to a graph representation whose nodes describe the object's activities in the field of view (FOV) of cameras, and connections between nodes represent transition probabilities (*i.e.* how likely an

object exiting a camera view would reappear in another camera view) or inter-camera time delay distribution (*i.e.* travel time needed to cross a blind area).

The rest of this subsection begins with a brief discussion on strategies for learning relationship between overlapping camera views. This discussion then concentrates on strategies for learning topology of non-overlapping camera views.

Overlapping camera views

Correspondences between objects and the relationship between cameras can be determined through careful reconstruction of three-dimensional camera models [44, 91, 222], estimation of homographies that relate the ground plane of a scene [139], and feature matching between multiple camera views [36, 39]. Although accurate observation correspondences can be established and precise spatial relationships can be discovered, the aforementioned approaches have several drawbacks that prohibit their use in surveillance scenarios with non-overlapping views:

1. *Most approaches can only work on overlapping FOV* - For instance, three-dimensional-based approaches require common salient feature points to establish correspondences. On the other hand, some approaches such as the method proposed by Caspi and Irani [37] requires significant common motion between the cameras, which can only be achieved given small disparity between cameras. Other works [8, 130, 200, 208] also assume significant overlapping camera views.

2. *Inter-camera time delay is ignored* - These approaches generally ignore the inter-camera transition time factor across views since given overlapping cameras, objects can be assumed to be close in space, time and appearance in successive camera views [115]. This assumption, however, is invalid given disjoint cameras with non-overlapping views, of which the observations are often widely separated in space and time. To establish correspondence across views, temporal constraint across views must be taken into account.

3. *Discovered relationship does not reflect the underlying activity* - Despite providing accurate spatial information between camera views, most of these techniques are not capable of reflecting the inter-camera relationships caused by underlying activities [66], *e.g.* actual paths taken by objects and the associated travel time between camera views.

4. *Tedious camera calibration is needed* - Some aforementioned methods require careful specification and manual camera calibration to establish camera correspondence. This

is infeasible for most surveillance scenarios where cameras may occasionally be removed or added to the network, or adjusted to face different angle.

Non-overlapping camera views

To determine inter-camera relationship from overlapping views to non-overlapping views, one shall take into account inter-camera transition probability and transition time distribution. The simplest strategy is to have a target walking in constant speed through all possible paths in a camera network in order to obtain the connectivity and transition time statistics. Whilst such a calibration method is attractive in its simplicity as pointed out by Ellis *et al*. [66], it is not feasible for real-world surveillance applications since an explicit calibration phase is needed and re-calibration might be required in the case of physical removal/addition of cameras. The fact that only a single object is considered also prevents the method from providing a realistic distribution of average transition statistics for activity analysis in a public scene.

1. *Feature matching approaches* - A more practical solution for establishing object correspondence is by matching individual object's visual appearance or motion trends such as movement speed across views. Once object correspondence is established using a large number of observations, it would be straightforward to infer the paths and transition time distributions between cameras.

   An earlier work of matching objects in disjoint views is undertaken by Huang and Russell [109]. In particular, they combine a set of spatial, temporal and appearance features to form an appearance probability for matching vehicles between two disjoint views along a motor way (Figure 2.5). By establishing correspondences between objects, the approach extracts and models the inter-camera transition times as Gaussian distributions. The method demonstrates high accuracy in estimating inter-camera transition time over a considerably long distance. However, it requires a training stage that assumes known correspondence. In addition, the cameras are placed alongside specific lanes where location and moving direction of objects are constrained.

   Javed *et al*. [115] carry out a more practical test that allows unconstrained object movements within a distributed camera network. In particular, they propose a supervised learning method based on Parzen window density estimator to jointly model object velocities, inter-camera transition times and spatial location for learning the inter-camera transition

Figure 2.5: Feature matching is employed to correspond objects for estimating transition time distribution between disjoint views. ([109])

probability and transition time distribution. Correspondences are assigned using a maximum *a-posteriori* (MAP) estimation. This method is later extended to handle illumination changes across views using brightness transfer function (BTF) [116], which map the colour distribution from one camera to another camera. Nonetheless, similar to Huang and Russell's method [109], the key drawback of this approach is that it requires a training stage with the assumption of known object correspondences, which is costly in manual annotation and not always available in most surveillance scenarios.

This drawback is later addressed in two more recent studies by Chen *et al*. [40] and Gilbert *et al*. [83]. Both methods relax the requirement of known correspondence during training stage by adopting an incremental learning strategy. The key difference between [83] and [40] is that the latter is capable of handling illumination changes with adaptive brightness transfer function (ABTF). Prosser *et al*. [186] also employ a BTF-based approach but the colour mapping function is only computed when sufficient visual evidence is accumulated. This cumulative brightness transfer function (CBTF) ensures an accurate estimation of the mapping function even if a sparse set of observations is given. In their later work [185], the CBTF is extended to allow adaptive update over time by combining past colour mapping function with background illumination changes in each camera view.

Another work is presented by Zou *et al*. [263], who use face recognition to better estimate the camera topology. The similarity scores of departure/arrival identities are modelled using mixture of Gaussian distributions. This method, however, is impractical in most surveillance scenarios since accurate face recognition requires high-resolution cameras and

subjects may not be facing the cameras all the time.

On a different note, extensive effort is channelled towards viewpoint invariant *person recognition or person re-identification* that aims to associate individuals observed under different camera views [69, 82, 89, 106, 186, 187, 256]. As opposed to the feature matching approaches discussed above, methods devised for person re-identification generally ignore the temporal constraints across views and match objects based solely on appearance features. Although these studies are not specifically formulated to learn inter-camera relationship, they can be easily extended to do so once correspondences between objects are determined. Conversely, more robust and accurate person re-identification can be achieved if inter-camera relationships are formulated as a contextual constraint on matching [40].

Overall, most of the aforementioned feature-based matching approaches depend on the availability of reliable visual and motion features from a target to achieve inter-camera object association. In practice, these strategies suffer significant feature variations across camera views due to changes in illumination (both intra-camera and inter-camera), camera orientation, and person appearance caused by pose change. Although various strategies have been proposed [82, 89, 116, 256] to adapt and rectify feature variation, object feature matching remains a notoriously difficult problem under real-world surveillance conditions. This is owing to the fact that even if feature variation were rectified or reduced, reliable features for matching may still not be available due to severe inter-object occlusions. Furthermore, ambiguities arising due to similar appearances between objects would lead to large numbers of false correspondences.

As pointed out by Wang *et al*. [233], even if the similarity of an object can be computed reliably, solving the correspondence problem using feature matching is still a nontrivial problem due to the enormous search space caused by large numbers of cameras and objects. Specifically, the search is carried out in a solution space of m-partile graphs, where m is the number of cameras. Unless camera topology is known where polynomial-time solution can be derived, the problem is NP-hard in the number of trajectories.

2. *Transition time distribution modelling* - It is possible to infer inter-camera relationships without solving the correspondence problem explicitly. Specifically, recent works [149, 150, 217] have suggested that camera topology can be inferred through searching for a consistent temporal correlation from population activity patterns (rather than individual

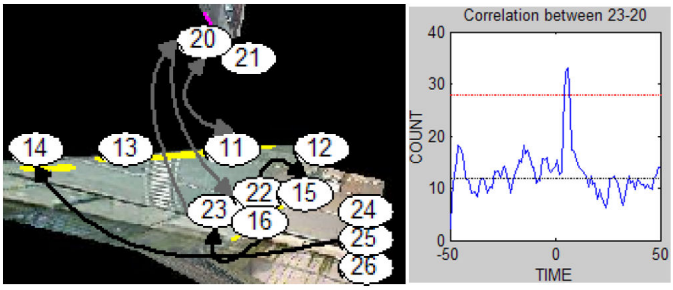


Figure 2.6: The figure shows the camera topology inferred by using a method proposed by Makris *et al*. [149]. (Left) Entry/exit zones are labelled on the ground plane. (Right) A peak on the cross correlation function suggests a connection between two zones with the corresponding time index indicating the most popular transition time. ([149])

whereabouts) across views.

With this idea in mind, Makris *et al*. [66, 149] present an unsupervised method to infer association of disjoint cameras by accumulating evidence from a large set of cross-camera trajectory observations. Specifically, they first identify the main entry and exit zones associated with different camera views by clustering starting and ending points of object trajectories using a Gaussian mixture model (GMM). All entry/exit events observed from any pair of entry/exit zones are assumed to be implicitly corresponding within a chosen time window. A transition time distribution of a set of these correspondences is then established by using cross-correlation analysis, followed by peak detection on the distribution using a thresholding strategy. A peak in the transition time distribution essentially implies a connection between the two camera views (Figure 2.6). Favourable results are demonstrated on a six-camera network. In particular, it is shown that the method is not only capable of learning the topological structure of a camera network, but also able to reveal the average inter-camera transition time, which can be used as a contextual cue for more robust inter-camera object tracking.

With a similar idea on exploiting transition distribution across views, Marinakis and Dudek [150] perform stochastic sampling on agent trajectories based on a delay model using Markov Chain Monte Carlo (MCMC) sampling. Nonetheless, the method assumes knowledge of the number of agents in the scene, which is often impractical in real-world surveillance scenarios.

An obvious way to improve the estimation of camera topology is to exploit appearance information on top of the cross-correlation between exit/entry events. Extending the idea

proposed in Makris *et al*. [149], Niu and Grimson [167] present an approach for a far-field traffic scene that combines both appearance information and transition distribution resulting from cross-correlation analysis. Specifically, they first extract appearance information (normalised colour and overall model size) of vehicles disappearing and reappearing in two views. Subsequently, similarity of appearance is exploited to filter out spurious cross-correlation values between views. The most popular transition time delay is then extracted, and the corresponding correlation coefficient is calculated to estimate the final mutual information to quantify the transition probability between views.

Tieu *et al*. [217] argue that a single-mode Gaussian transition distribution applied in Makris *et al*.'s study [149] (also approaches that adopt similar idea [150, 167]) cannot accurately recover multi-modal transitions between camera, *e.g*. when both car and pedestrian with different speeds are part of the observations. In addition, previous study [149] assumes implicit correspondences within a fixed time window, which is hard to define. Moreover, increasing the time window would result in more false correspondences that eventually hamper the accuracy of transition time estimation. To overcome the drawbacks, Tieu *et al*. [217] relate statistical dependence between observations in two cameras without a time window by computing mutual information (MI) using nonparametric density estimation and eliminate uncertain correspondences through MCMC sampling. Results on both synthetic data and real traffic data demonstrate the effectiveness of their method in handling multi-mode transition distributions.

It is worth noting that all the aforementioned methods rely on intra-camera (within camera) tracking to detect entry and exit events for modelling transition time distribution. As discussed in Section 2.1.1, explicit object segmentation and tracking is nontrivial in a crowded scene, especially given video captured in low temporal and spatial resolutions. Even if entry and exit events were reliably detected, the constantly busy scene would result in excessive number of false association of entry/exit pairs, leading to incorrect estimation of transition time. To address this problem, Chapter 3 will present an approach to modelling arbitrary time delayed correlations among multi-camera activities without relying on either intra-camera or inter-camera tracking; it is therefore more scalable to crowded scenes and videos with low temporal and spatial resolutions.

3. *Co-occurrence based approaches* - It is worth pointing out that Hengel *et al*. [225] also

attempts to infer camera topology without relying on object tracking. Specifically, they start with full connectivity among camera views and gradually eliminate linkages among image regions that exhibit simultaneous object occupancy and vacancy. The method is appealing in its simplicity and is scalable to large number of cameras. However, it only examines static object co-occurrence over time among overlapping camera views without considering the temporal relationships at all.

Wren *et al*. [237] takes a slightly different tracking-free approach. In particular, they quantify inter-camera connectivity by accumulating the co-occurrence statistic given a predefined temporal offset between two streams of events observed from two separate views. Although a small temporal offset is fixed, the method is still unable to capture arbitrary time delay between camera views. It is thus limited to learning the connectivity between cameras with very short gap or overlapping views. On the contrary, the proposed approach in Chapter 3 takes temporal relationships among activities into account. In addition, although this thesis focuses on disjoint cameras, the proposed approach can be readily used for camera views with different degrees of overlapping.

2.3.2 Global Activity Modelling and Unusual Event Detection

There has been considerable amount of work for activity understanding and unusual event detection, but mostly devoted to single camera scenario (Section 2.2). These methods are not directly applicable to scenarios involving multiple disjoint cameras since there is no mechanism to discover and quantify the time delays among activities observed across non-overlapping views.

An obvious and equally intuitive approach to activity understanding and unusual event detection in multiple cameras is to reconstruct the global path taken by an object by merging its trajectories observed in different views, followed by a standard single-view trajectory classification approach to identify atypical global behaviour of the object [251]. With this approach, one must solve the camera topology inference problem [149, 217] and the trajectory correspondence problem [116] as discussed in Section 2.3.1. Both problems can be difficult given a large number of disjoint cameras and video captured in low spatial and temporal resolutions from crowded public scenes, *i.e*. one may never be able to visually construct object global motion trajectories consistently for all objects.

Wang *et al*. [233] propose an alternative trajectory-based method that bypasses the topology

inference and correspondence problems. In particular, trajectories observed in different views are first connected based on their temporal proximity, *i.e*. two trajectories are connected if their temporal distance is less than a fixed threshold, T. Then a generative model based on LDA is used to cluster the trajectories, forming a set of activity categories. These categories are then used to infer the transition probability between cameras. A trajectory is flagged as being unusual if it has a low likelihood given the learned model. Note that their trajectory co-clustering method is based on LDA and is thus limited to capturing only co-occurrence relationships among activity patterns. Any temporal relationship is not discovered and quantified automatically but simply determined by the predefined temporal threshold, T. Importantly, the method cannot cope with busy scenes as there will be too many false inter-camera trajectory connections.

In contrast to the aforementioned trajectory-based approaches [233,251], Chapter 4 proposes a method that automatically infers time delayed dependencies between local activities across views without relying on explicit object-centred segmentation and tracking. Therefore, it can be applied to low-quality public scene surveillance videos featuring severe inter-object occlusions for robust multi-camera unusual event detection.

A tracking-free method has been proposed before for multi-camera unusual event detection. In particular, Zhou and Kimber [259] propose an event-based approach by detecting blob events in each camera view and model them as a first-order Markov chain in a CHMM. The chain's connectivity is manually defined and labelled to reflect neighbouring relationships of cameras. The model becomes intractable even with a small number of camera views. Moreover, the model is restricted to capturing first-order temporal dependency, which is not suitable for modelling cross-camera activity dependencies with arbitrary time delays. In comparison to the method proposed by Zhou and Kimber [259], the multi-camera unusual event detection approach presented in Chapter 4 learns activity dependencies without the need for any prior knowledge on camera topology or top-down rules for labelling. In addition, it is scalable and computationally tractable by adopting a probabilistic graphical model with less complex structure, which is capable of handling arbitrary time delays by performing cross correlation among distributed activity patterns.

2.3.3 Graphical Model in Multi-Camera Activity Analysis

Section 2.3.1 highlighted several approaches that utilise graphical model to represent camera topology [149, 217] with different nodes representing entry/exit zones from different views, and the probabilistic links between nodes encoding the transition probabilities and travel time be-

tween the zones. A similar graphical model representation is employed in Chapters 4 and 5.

To use the model for global unusual event detection, it is a prerequisite to learn the graphical model using local activity observations extracted from different camera views. Learning a graphical model from observation data is an important and extensively studied problem. One of the main reasons is that the learned model can be used as a decision support system. Moreover, the dependency link of a model can potentially reveal causal relationships among different nodes, which is useful for probabilistic reasoning, *e.g*. interpretation of gene regulatory pathways [74].

Learning a graphical model such as Bayesian network involves the learning of both structure (graph topology) and parameters of each conditional probability distribution. The parameters are typically learned using the Expectation-Maximisation (EM) algorithm. Nevertheless, if the data is fully observable (each instance assigns values to all the variables of interest), a closed-form solution is available for Bayesian learning by keeping a density over possible parameter values [97].

As for structure learning, the aim is to search for a topology that best explains the observations. The problem of structure learning is known to be NP-hard, as the size of structure search space is super-exponential to the number of variables in a network [223]. Consequently, the structure learning problem is often addressed using heuristic methods. Previous heuristic methods can be categorised into either constraint-based methods [50, 123, 206], or scored-searching based methods [45, 51, 97].

Algorithms following the constraint-based methods attempt to find equivalent networks that are consistent with conditionally independent constraints derived from statistical or information theoretic measures. For instance, the PC algorithm[1] [206] performs a chi-square statistical test to decide conditional dependencies between variables and use the results to infer a network structure. Another example is the maximum weight spanning tree proposed by Chow and Liu [47] that is capable of approximating a tree structure with optimal joint probability based on pairwise mutual information (MI) among variables. The constraint-based methods are computationally tractable for a large network with hundreds of nodes, but they are sensitive to failures in conditional independence tests as pointed out by Friedman *et al*. [75]. Hence, it is more common to adopt a data-driven optimisation approach, known as scored-searching based method for learning a network structure.

[1] PC algorithm is named after Peter Spirtes and Clark Glymour, who invented it.

A scored-searching based method searches across the space of all possible structures to find an optimal network, which maximises a score that evaluates a given structure as a function of the data. A popular searching method is greedy hill-climbing (GHC) search [46]. In GHC search, different local operations[2] are performed in each searching step to find a structure that yields the highest score and conforms the acyclicity constraint. The searching step is repeated until no more positive changes can be made. There are approximately $O(n^2)$ possible local operations in each step where n is the number of variables. The cost of these evaluations become acute when a large network with hundreds of nodes are given [75]. Another scored-searching based method that is more tractable is the K2 algorithm[3] [51]. The method relies on a variable ordering list to greedily select parent sets for a given variable in order to maximise the overall network score. With the variable ordering list, the search space of the K2 algorithm is greatly reduced as compared to the conventional GHC search [10]. In addition, since the variable ordering list already ensures acyclicity of a structure, costly acyclicity checking is avoided.

Hybrid methods have also been proposed, which combine both constraint-based and scored-searching methods in order to improve computational efficiency and prediction accuracy in structure learning [42, 205, 223, 229]. Those approaches typically derive a set of initial network constraints by performing constraint-based learning. The constraints are then propagated to subsequent scored-searching based learning to reduce and constrain the network search space, by eliminating any candidate structures inconsistent with the constraints. This consequently leads to a significant computational speed-up in the searching process when scored-searching based learning takes place. In addition to the speed acceleration, accuracy of the structure may also improve since scored-searching based learning could discover highly interdependent sets of parents that might fail a conditional independence test during the constraint-based learning [75]. The existing hybrid methods, however, are not capable of learning graph dependencies among multiple time-series with unknown time delays. To overcome this problem, a new hybrid method that combines the K2 algorithm with an information theoretic based analysis for time delay estimation is proposed in Chapter 4.

The aforementioned structure learning methods for Bayesian networks are limited to batch mode learning. A model learned in batch mode may encounter deterioration in modelling accuracy due to context changes over time, such as changes of interdependency between variables or

[2]The usual choice of local operations are edge deletion, insertion, and reversal.

[3]The algorithm is named after an earlier version of the algorithm called Kutató [99]

shifts in the underlying distribution. In a realistic and unconstrained environment, it is necessary to adapt a model when new data is observed in order to cope with context changes. Several approaches for incremental structure learning have been proposed in the past [72, 135, 166] to sequentially update the parameters and structure of a model. A notable method is described by Friedman and Goldszmidt [72], whereby a structure is updated sequentially without having to store all the earlier observations. Specifically, the method keeps a set of network structures that are deemed most promising so far in a "frontier", together with the associated sufficient statistics of the structures. In each learning iteration, a search process is invoked and the obsolete structure is replaced by one of the structures in the frontier. The method remains memory and computationally efficient for a network of moderate size (*e.g*. ALARM network with 37 variables [16]), but may become intractable given a large network with hundreds of nodes [75] since a GHC search is employed. In addition, it does not consider temporal delay between multiple time series. It is thus not ideal for learning large camera topology characterised by activity patterns with arbitrary time delays.

The brief review of the graphical model learning in this subsection underscores the fact that no attempt has been made so far to exploit the time delayed dependencies present in underlying activities for graphical model learning. This is precisely where the main contribution of the proposed methods in Chapters 4 to 5 lies. In particular, it is shown that the time delay dependency estimated among partial observations monitored through multiple non-overlapping cameras can be utilised to facilitate graphical model learning for global unusual event detection.

2.4 Activity Learning Strategies

Learning strategies for activity understanding and unusual event detection vary in the amount of human supervision entailed, ranging from supervised learning, unsupervised learning, semi supervised learning, and active learning.

2.4.1 Supervised Learning

In a full supervised learning paradigm, it is assumed that both positive and negative instances are well-defined and available during training phase. Much earlier work on activity understanding and unusual event detection takes a supervised learning strategy. For example, Oliver *et al*. [171] describe a CHMM formulation to recognise human interactions, with the assumption that there

exist known *a priori* behaviour classes. In a car park application, Morris and Hogg [158] detect atypical trajectories by training a statistical model with both normal and atypical trajectories. In another study, Sacchi *et al*. [193] train a neural network with both normal and unusual behaviours to detect vandal acts. Gong and Xiang [86] recognise group activities by training a DML-HMM, of which the hidden states correspond to different event classes that are known beforehand. A slightly different supervised approach is taken in rule-based approaches [58, 201, 221], whereby human perception on what is normal/unusual is hand-crafted into a model for unusual event detection.

These approaches provide effective solutions for activity modelling and unusual event detection given simple and static scenes. Nevertheless, given a public scene with complex behaviours changing over time, it is no longer realistic to expect the existence of all possible configurations of positive and negative instances. In this case, a supervised approach generally requires a large amount of well-labelled data to model the complex behaviours, so as to prevent the generalisation problem on unseen instances. Without comprehensive training data, the model may be overfitting for an incomplete training set, leading to poor unusual event detection performance in practical settings. For rule-based approaches [58, 201, 221], the need for defining extensive rules becomes infeasible and may break down easily when the visual context and definitions of normality/abnormality drift over time [243].

2.4.2 Unsupervised Learning

Clustering strategy

In contrast to supervised learning, unsupervised learning does not assume any prior knowledge on the pattern classes. A common strategy is to perform clustering on unlabelled data in batch mode and label small clusters as being unusual. For instance, Zhong *et al*. [258] perform a bipartite co-clustering on video segments and identify isolated clusters as unusual events. With a similar idea, Lee *et al*. [138] use a N-cut clustering technique to separate unusual behaviours from normal behaviours based on distance measurement. Note that however, these approaches rely on batch clustering to compute internal cohesiveness of clusters for unusual event detection. Given new instances, a new round of clustering is required to re-compute the similarity between events. They are thus only suitable for post-mortem analysis but not for on-line unusual event detection.

This problem has later been addressed by Hamid *et al*. [93] whose method employs graph

theoretical clustering to cluster normal and abnormal events. Without the need to re-cluster the entire dataset, the abnormality of an unseen instance is determined by simply computing a weighted similarity between the new activity-instance and all the events in existing clusters. However, all previous instances have to be stored in memory for similarity comparison, limiting the scalability of this approach.

Non-parametric learning strategy

Another popular strategy that assumes no information on prior class labels relies on nonparametric modelling. For instance, Boiman and Irani [25] build a database of spatiotemporal patches using only regular/normal behaviour and detect those patterns that cannot be composed of the database as being unusual. This method, however, faces the same scalability problem in Hamid *et al.*'s approach [93], where all previous ensembles of spatiotemporal patches have to be stored.

Another nonparametric approach is described by Breitenstein *et al.* [29], whereby usual scenes are defined using nearest neighbours, and unusual event is identified by measuring its distance from the distribution of nearest neighbours. The method maintains a fixed number of clusters in their nearest neighbours model, thus avoids the memory problem encountered by Boiman and Irani's method [25], but at the expense of having an additional free parameter that specifies the trade-off between generalisation and discrimination of the model.

One-class unsupervised learning strategy

A popular activity learning strategy in this domain is to train a model using only normal class of instances or dataset dominated by normal events. Unusual events are subsequently detected as outlier, *i.e.* those that statistically deviate from the learnt normal profile.

This strategy is based on the observation that surveillance data is typically characterised by vast amount of uninteresting normal events and relatively scarce amount of interesting or unusual events. In addition, it is assumed that normal behaviours are well-defined and thus clustered better as compared to unusual instances with unpredictable variations [244, 258].

This strategy is often grouped into unsupervised learning category in the literature of activity understanding. Arguably, following the strict definition of unsupervised learning that receives no supervision at all [81], it may be more appropriate to refer the aforementioned strategy as one-class classification [213][4], which aims to draw a class boundary enclosing the normal patterns and reject any unusual patterns falling outside the boundary. Nevertheless, this thesis shall follow

[4] A class may consist of multiple clusters of activity patterns.

the notion of unsupervised learning typically used in the literature.

Many approaches have been proposed following one-class unsupervised learning strategy, by using models such as DBNs [64] and PTMs [104, 140, 155, 231]. For example, Duong *et al*. [64] exploit only the labelled normal duration behaviours to train a Switching Hidden semi-Markov Model (S-HSMM) to detect abnormality in the duration of activities. For PTMs based approaches, words and documents are typically constructed using normal instances. Topic models such as LDA [23] is then employed to discover the distribution of topics for the normal behaviour [155].

Incremental update for unsupervised learning strategy

Most of the methods mentioned above do not incorporate on-line model adaptation into the framework. In essence, due to the inevitable changes of visual context in realistic and unconstrained environments, the definition of normality/abnormality may change over time and the underlying distribution of the same behaviours may also undergo circumstantial changes [243]. As a result, most aforementioned methods may degrade in the long run since they remain static once trained without accommodating changes over time.

To cater for the visual context changes, Xiang and Gong propose an unsupervised method with incremental learning capability [241, 243]. Specifically, a small amount of training data represented as discrete events is used to construct a mixture of multi-observation hidden Markov models (MOHMMs). Incremental Expectation-Maximisation (EM) learning takes place when a new observed pattern is presented. A likelihood ratio test is introduced to determine if the new behaviour pattern is normal or unusual. Taking the classification output, the model is then adapted by either updating parameters or creating a new class if the pattern is totally unseen before. In the model adaptation process, a positive class might be switched to a negative class, or vice versa depending on the availability of new supporting observations. Normality/abnormality of classes is determined using a heuristic based on class proportion constraint, *i.e*. a fixed ratio of normal behaviours and abnormal behaviours is maintained. In Xiang and Gong's method [241, 243], however, only a clip-level measure of abnormality is considered without localising the unusual event explicitly in the scene.

Another unusual event detection approach based on graphical models is described by Kim and Grauman [131]. The method also features incremental learning but with additional capability in localising unusual events, unlike method proposed by Xiang and Gong [243]. In particular,

local optical flow patterns in decomposed regions are represented as mixture of probabilistic principal components (MPPCs) [218], and a space-time Markov Random Field (MRF) is used for event modelling in a global scale. Incremental updates of the MPPCs and associated MRF parameters are carried out to adapt the model as new video data streams in. Unusual events are localised by performing inference on each region node modelled by the MRF model. It is shown that incremental updates allow the algorithm to adapt to visual context changes over a long period of time, *e.g*. earlier events detected as unusual are later detected as normal after several occurrences.

2.4.3 Semi-Supervised Learning

For visual learning, unlabelled instances are often easy to collect, whilst labelled instances, either normal or unusual events are difficult or time-consuming to obtain, as they require exhaustive annotation from human experts. Falling between the two extremes of supervised learning and unsupervised learning, semi-supervised learning [261] appears to be a natural solution for activity modelling since it may be used to construct a predictive model by making use of a small amount of labelled data together with a large number of unlabelled instances. Typically, a model is first initialised with handful of labelled data, which include positive training instances and negative training instances. In the testing stage, the unlabelled data is either assigned to positive or negative class. Those unlabelled points that pass the confidence level, with the predicted labels are used to retrain the classifier sequentially.

One of the most prominent methods following a semi-supervised strategy is described in Zhang *et al*. [252]. The key idea is to train an HMM using a large amount of labelled usual behaviours. By performing iterations of likelihood test and Viterbi decoding on unlabelled video sequences, a number of unusual models is derived using the unlabelled sequences via Bayesian adaptation [188]. Subsequently, classification of events can be performed with the usual and unusual models. A drawback of this method is that its accuracy is very sensitive to the number of iterations during the semi-supervised adaptation process. It is observed that the false alarm rate increases rapidly along with the number of iterations.

A general concern of using semi-supervised learning strategy is that the use of unlabelled data may hurt instead of improving the classification performance. This problem occurs because semi-supervised learning relies fundamentally on what a model knows about based on a small amount of labelled data, and propagate the acquired knowledge and assumption to labelling unlabelled

instances for relearning. If the model assumption matches badly with the problem structure, poor classification performance may be obtained [54]. In a visual surveillance context where observations are uncertain and dynamic, it may be more appropriate to attack the problem from an opposite direction, *i.e.* explore the unknown and ambiguous aspects instead of exploit what a model thinks it knows about the unlabelled data. The following subsection discusses how a model could resolve the unknowns and ambiguities by learning from human feedback via an *active learning* strategy.

2.4.4 Active Learning

Existing approaches [104, 131, 155, 231] mostly address the unusual event detection problem using an outlier detection strategy, *i.e.* training a model with normal event patterns, and a flag newly observed pattern as an anomaly if it deviates statistically from the learned normal profile. As discussed in Chapter 1 (Section 1.2.2), these unsupervised approaches entail a number of limitations, such as the difficulty of detecting subtle anomalies or distinguishing specific anomalies of interest from noise and uninteresting outliers. In essence, such knowledge may simply not exist in the limited amount of training data at hand and may not be extracted through unsupervised mining [142].

To address the aforementioned issues, other sources of information need to be exploited. One of the non-visual sources is human input, which can be used to resolve ambiguities during class decision boundary formation. A fully supervised learning strategy, however, is time-consuming because exhaustive labelling is required to cover sufficient representation of different classes. Supervised strategy that performs exhaustive random labelling is inefficient since it treats all samples equally, whilst fundamentally not all samples are critical for fine-tuning the section of the decision boundary that is "confused" by visual ambiguities and lack of training data [121].

A potential solution to incorporate human input into an automated model building process is by using an active learning strategy [197]. In contrast to conventional unsupervised learning that fully relies on unlabelled data, and supervised learning that exhaustively labels all the available instances, active learning aims to reduce the amount of manual data annotation without compromising the performance of a classifier. This is achieved through selecting the most critical and informative data sample point for labelling request based on a set of predefined query criteria.

Pool-based setting vs. stream-based setting

In general, there are two different settings in active learning, namely a pool-based setting and stream-based setting [197]. A pool-based active learning method requires access to a fixed pool of unlabelled data for searching the most informative instance for querying. For a surveillance task since activity patterns are dynamic and unusual events are often unpredictable, preparing a pool of unlabelled data that encompasses complete event classes is impractical. Moreover, performing exhaustive search in the pool is expensive therefore unsuitable for surveillance tasks that demand real-time performance. Stream-based setting on the other hand does not assume the availability of unlabelled data pool; it requests human verification on-the-fly based on sequential observations. Whilst this setting appears more challenging since an immediate query decision has to be made without complete knowledge on the underlying data distribution, it is preferred in surveillance context that demands real-time response.

Query criteria

After deciding an appropriate active learning setting, one has to select a suitable query criterion for active selection of query samples. A number of criteria have been proposed in the past [197], such as uncertainty criterion [100], likelihood criterion [177], expected model change [198], and expected error reduction [190].

Different query criteria are needed for active learning in surveillance applications. The reason is that some classes, especially unusual event classes have to be discovered (*i.e.* exploration) since they are not available in the early stage of training. At the same time, it is necessary to improve the model gradually by refining the decision boundary (*i.e.* exploitation). Both exploration and exploitation can be satisfied by using *uncertainty criterion* and *likelihood criterion*. The uncertainty criterion prefers instances whose labels the learner is unsure of, *e.g.* ambiguous points closest to the decision boundary. Whilst the likelihood criterion selects points furthest away from the labelled points or with lowest likelihoods w.r.t. a model.

Most existing stream-based approaches are based on single query criterion [9, 100, 199], which are obviously not sufficient for exploration and exploitation that pursue different goals. Even though there are attempts in combining multi-criteria for active learning, they are either not adaptive [177, 209] or limited to pool-based setting [12, 38]. For instance, methods proposed by Stokes *et al.* [209] as well as Pelleg and Moore [177] iterate over different criteria to select a sample from an unlabelled set. Whilst Symons *et al.* [212] simply apply predefined

weights on individual query criteria throughout the active learning cycle. Non-adaptive methods (*e.g*. iterate over different criteria with constant weights) may fail in applying the right criteria at different phases of learning, *e.g*. the active learner may waste effort refining the boundary before discovering the right classes, or vice versa. In contrast, Baram *et al*. [12] propose an adaptive multi-criteria active learning approach, which adjusts weights on different criteria guided by classification entropy computed over a pool of unlabelled data. In another study by Cebron and Berthold [38], the weight adjustment is conducted based on uncertainty distribution defined by density estimates on an unlabelled set. Both aforementioned methods [12, 38], however, require access to a pool of unlabelled data, which are often unavailable to stream-based environments.

One of the most popular and theoretically motivated stream-based strategies is the Query-by-Committee (QBC) algorithm [9, 199]. The QBC approach maintains an ensemble of committee members that represent competing hypotheses but consistent with the training data seen so far. Query will be triggered if the class label of a sample is controversial among the members. Various measures of disagreement have been proposed such as vote entropy [9, 55] and average KL divergence [152]. These measures, however, only return the disagreement score among members without identifying conflicting classes, *i.e*. the classes closest to the uncertain point. This is undesirable since video surveillance data is typically characterised by highly imbalanced class distribution, *i.e*. most of the samples correspond to normal event classes whilst the remaining unusual event classes only constituent a small percentage of the entire dataset. For a more balanced sample selection for class imbalanced data, it is important to identify conflicting classes, so that uncertain samples surrounding unusual event classes can be favoured during query sample selection.

Active learning in activity analysis

There have been very few active learning approaches specifically proposed for activity understanding and unusual event detection. Zhou and Kimber [259] attempt learning from human feedback for more robust unusual event detection. Specifically, hierarchical clustering [114] is conducted to cluster unlabelled behaviour patterns. All events in the large clusters are labelled as being normal whilst those that are grouped into the small clusters are treated as unusual events. Manual selection is carried out to return normal events that are mistakenly grouped into smaller clusters back into the normal class. A model is then built using all normal instances for unusual event detection. It is claimed that the manual selection strategy could help in refining the clas-

sification boundary between normal and unusual events. However, no experimental results are shown to validate the effectiveness of the strategy. In addition, their method only exploits human feedback during batch mode learning but not exploited to incrementally enhance the model.

Sillito and Fisher [204] formulate a method to harnesses human feedback on-the-fly for improving unusual event detection performance. Using a model trained with normal instances, any instances classified as normal events in the testing stage will be used together with corresponding predicted labels to retrain the model. On the other hand, human approval is sought if a newly observed instance deviates statistically from the learned normal profile. If the suspicious instance is indeed normal, it will be included in the retraining process, or else it will be flagged as an anomaly. The framework is advantageous as compared to passive unsupervised learning in that it allows incremental incorporation of normal instances in an on-line manner to refine the model, whilst simultaneously it prevents anomalous behaviour being inadvertently incorporated into the model so avoiding corruption to the learned normal profile. However, their method is limited to learning the normal event class without exploiting the subsequent flagged anomalous behaviour. Therefore, it may still have a problem in distinguishing a genuine unusual event from noisy observations and normal patterns, which could be similar visually to the unusual instance as discussed in Section 1.2.2.

Unlike the method presented by Zhou and Kimber [259], the proposed active learning strategy in Chapter 6 exploits human feedback in an on-line manner. Specifically, the approach is formulated as a stream-based solution, *i.e.* it makes immediate decisions whether to request for labels for each unlabelled sample observed in the sequence. In contrast to the approach described by Sillito and Fisher [204], the method proposed in Chapter 6 exploits both normal and unusual events labelled by human during active learning to strengthen the classification boundary. Importantly, it selects adaptively from both likelihood criterion and uncertainty criterion for resolving ambiguities of interest. In addition, a new QBC scoring method is formulated to identify conflicting classes, thereby incorporating a constraint to favour uncertain samples surrounding unusual event classes, leading to a more balanced sample selection for class imbalanced data.

2.5 Summary

The preceding discussions have covered essential studies in the literature regarding activity understanding and unusual event detection in surveillance videos. In particular, various represen-

tations of activity, models for activity learning, and the associated learning strategies have been discussed. This chapter has also reviewed the state of the art approaches in multiple camera activity analysis.

Existing methods have shown promising results in various surveillance tasks. Nevertheless, there are several open problems and limitations that need to be solved. Firstly, most existing approaches for multi-camera activity understanding assume reliable intra-camera and inter-camera tracking for learning spatiotemporal relationships among non-overlapping camera views. These methods are likely to fail given crowded public scene video captured with low frame rate and poor resolution. Secondly, no tracking-free method has been proposed to model arbitrary time delays and relationships between multiple cameras for global unusual event detection. In addition, incremental learning in such a problem has never been attempted to date. Thirdly, unusual event detection approaches mostly rely on an outlier detection strategy that may be prone to inevitable false alarms due to ambiguity in feature space. Conventional active learning methods are still inadequate for surveillance applications since they either assume pool-based setting or employ single query criterion, without special treatment of the imbalanced class problem.

In subsequent chapters of this thesis, approaches are formulated to address these limitations following the two key concepts below:

1. *Learning visual context* - New approaches are formulated to discover and model the spatiotemporal and correlation context of a complex scene, which is monitored using multiple disjoint cameras separated by unknown and arbitrary time gaps. This is achieved without any prior knowledge on the camera topology or top-down rules. Importantly, contrary to the conventional methods discussed in Section 2.3.1, the proposed solution does not rely on either intra-camera or inter-camera object tracking; it thus can be applied to low-quality surveillance videos with severe inter-object occlusions.

2. *Learning from human feedback* - A stream-based active learning approach is formulated to incorporate human feedback for on-line unusual event detection. In contrast to most existing unsupervised learning-based methods (described in Section 2.4.2) that perform passive mining for unusual events, the proposed method automatically requests supervision for critical points to resolve ambiguities of interest, leading to more robust and accurate detection of subtle unusual events.

Chapter 3

Finding Pairwise Correlation

Performing global activity analysis in a public space through disjoint multi-cameras is nontrivial, especially with non-overlapping inter-camera views, in which global activities can only be observed partially with different views being separated by unknown time gaps. In particular, the unknown and often large separation of cameras in space increases the uncertainties in activity understanding due to temporal discontinuity in visual observations, as well as drastic visual appearance feature variations due to changes in illumination (both intra and inter-camera), camera orientation, and pose changes. Given multiple cameras with non-overlapping views, the key to activity understanding lies in how well one can associate partial observations of activities across camera views to infer a meaningful scene context prior.

To this end, this chapter proposes a new approach to understand multi-camera activities by modelling time delayed correlations among partial observations observed from disjoint cameras without either inter-camera or intra-camera tracking. Specifically, since a complex scene naturally consists of multiple local scene regions that encompass distinctive activities, each camera view is first decomposed automatically into regions, across which different spatiotemporal activity patterns are observed. A novel Cross Canonical Correlation Analysis (xCCA) framework is then formulated to discover and quantify pairwise correlation and temporal relationships of *arbitrary time delays* between these multi-camera regional activities. In contrast to existing methods [40, 115, 149, 150, 167, 217], the proposed approach does not rely on explicit object segmentation and tracking; it thus can be applied to crowded scenes as well as videos captured with low frame rate and poor resolution.

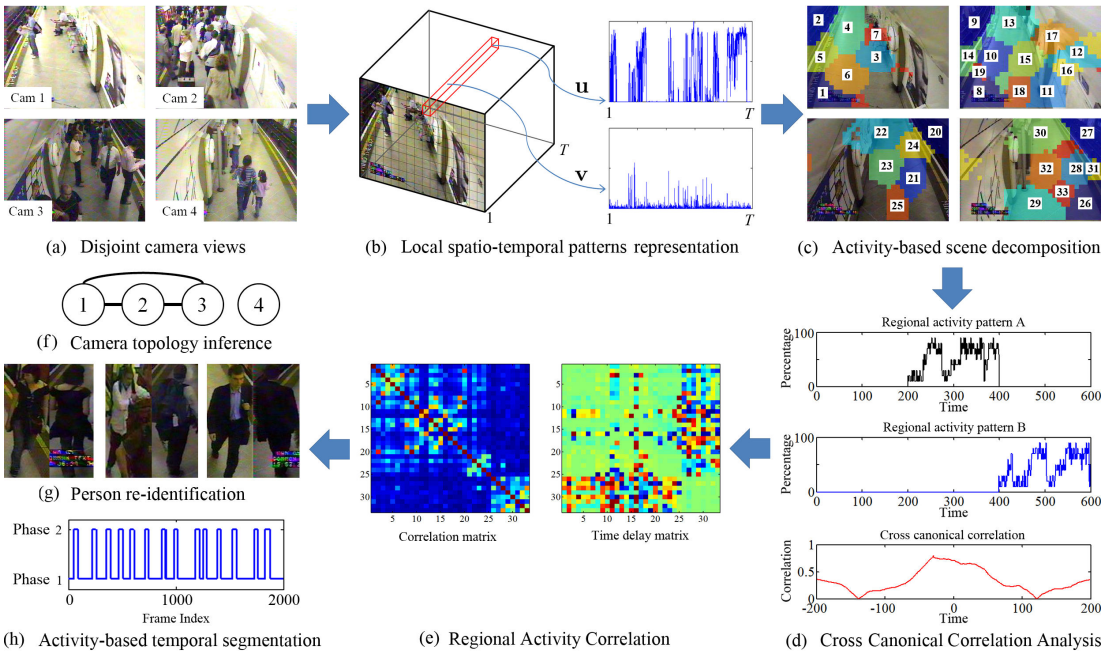


Figure 3.1: A diagram illustrating the multi-camera time delayed activity correlation approach.

This chapter is structured as follows: the proposed framework is explained in Section 3.1. Section 3.2 demonstrates that the learned time delayed correlations can be exploited as useful context cues to facilitate (1) spatial and temporal topology inference of a camera network, (2) robust person re-identification, and (3) accurate activity-based video temporal segmentation. Results are reported and discussed in Section 3.3. Specifically, the effectiveness of the proposed approach is evaluated using 330 hours of videos captured at 0.7 frame per second (fps) from two busy underground stations with eight and nine camera views, respectively, all of which feature crowded scenes and complex activities. Finally, conclusions are drawn in Section 3.4.

3.1 Pairwise Correlation Analysis

The key components of the proposed approach are illustrated in Figure 3.1. Given disjoint cameras in a camera network (Figure 3.1(a)), local spatiotemporal patterns are first extracted from each camera view and represented as time-series data (Figure 3.1(b)). The patterns are then used as input to activity-based scene decomposition method to segment the scenes into regions (Figure 3.1(c)), from which the regional activity patterns are extracted. Subsequently, the xCCA is performed to infer inter-region pairwise time delayed correlations (Figure 3.1(d)). Regional activity correlations are then discovered and quantified (Figure 3.1(e)). In this chapter, a training process refers to the process of activity-based scene decomposition and xCCA. Finally the

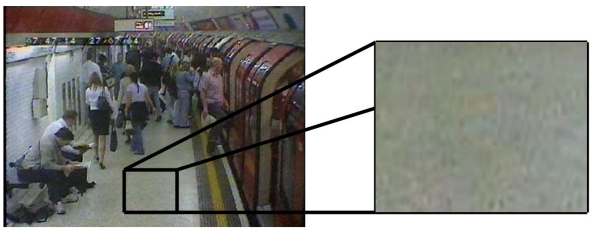
inferred regional activity correlations are exploited for camera topology inference, and used as contextual information for person re-identification and activity-based video temporal segmentation (Figure 3.1(f–h)).

3.1.1 Scene Decomposition and Activity Representation

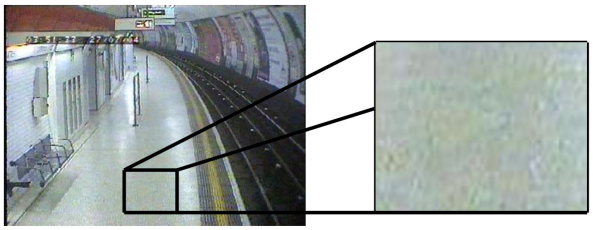
A complex public scene naturally consists of multiple local regions, each of which encapsulates a unique set of activity patterns correlated with each other. Given a set of training video sequences, the proposed approach aims to decompose m camera views into n regions $\boldsymbol{\mathcal{R}}$ according to the spatial-temporal distribution of activity patterns, where $\boldsymbol{\mathcal{R}}$ is given as:

$$\boldsymbol{\mathcal{R}} = \{\mathcal{R}_i | i = 1, \ldots, n\}. \tag{3.1}$$

Consequently, the j-th camera view in the network contains n_j regions with n_j being determined automatically, and $n = \sum_{j=1}^{m} n_j$. It is noted that a straightforward way would be to simply use fixed partitions of the scene. However, it offers no guarantee that a fixed-partitioned region will represent local scene region that encompass distinctive set of activities.



(a)



(b)

Figure 3.2: The figure depicts a frame with abrupt intensity level change (a) compared to its background model (b). The ratios of RGB channels between the extracted regions of (a) and (b) are 1.2380, 1.2829 and 1.3428 respectively.

Robust background modelling – The proposed approach represents activity patterns using foreground pixel-based representation (discussed in Section 2.1.2). A simple method to extract foreground pixels from current frame is by subtracting the current frame with a pre-estimated back-

ground image [178]. This frame differencing method, however, is susceptible to sudden and frequent intensity changes in real-world surveillance videos. Using this method may introduce noise into the representation and subsequently affect the accuracy of time delayed correlation analysis. To address this problem, a robust background modelling method is formulated[1]. The changes in intensity level are caused either by lighting condition changes (*e.g.* moving clouds outdoor or flashing advertising boards indoor) or camera response to different crowdedness in the scene. In the latter case, auto gain and white balancing functions of cameras yield different global intensity level on a particular video frame when crowd or large objects are present in the video. An example of the latter can be seen in Figure 3.2 where drastically different global intensity level are observed when an underground train platform changes from crowded (Figure 3.2(a)) to empty (Figure 3.2(b)).

The key idea of the proposed method is to adapt the background image to the intensity levels of the current frame prior to background subtraction. In particular, a static background image is first constructed by employing a method proposed by Russell and Gong [191]. Pixel-level intensity ratios $\mathbf{g}$ are then computed between all pixels of the stored background image and the current frame. Subsequently, the background image is adjusted by multiplying all its pixels with $\mathbf{g}$. However, not all the ratios reflect the true intensity level change as some of them belong to foreground regions. To eliminate the effect of incorrect ratios caused by foreground regions, a mean-shift procedure [79] is performed to find the stationary point of the distribution of $\mathbf{g}$. In particular, the centre of a Gaussian kernel denoted by $\{\mathbf{c}_j\}_{j=1,2,\ldots}$ is iteratively moved from the current point to the new point according to:

$$\mathbf{c}_{j+1} = \frac{\sum_{i=1}^{n_{\text{pixel}}} \mathbf{g}_i \exp\left(-\frac{\|\mathbf{c}_j - \mathbf{g}_i\|^2}{2h^2}\right)}{\sum_{i=1}^{n_{\text{pixel}}} \exp\left(-\frac{\|\mathbf{c}_j - \mathbf{g}_i\|^2}{2h^2}\right)}, \; j = 1, 2, \ldots \tag{3.2}$$

where n_{pixel} is the number of pixels and the bandwidth, h is set to 1 in this study. To obtain the initial point $\mathbf{c}_1$ of the kernel, a coarse background subtraction is first performed between the stored background image and the input image. The initial point is then computed as the mean of the intensity ratios between all the non-foreground pixels. The mean shift procedure terminates

[1]Matlab implementation of the proposed background subtraction method is available at `http://www.eecs.qmul.ac.uk/~ccloy/files/substractBackgroundMS.zip`

when the maximum iteration allowed is reached or:

$$\|\mathbf{c}_{j+1} - \mathbf{c}_j\| < \varepsilon, \tag{3.3}$$

where ε is set to a small value. The final centre of the kernel gives the most likely intensity ratios that account for the change of intensity level. These ratios are used to adjust the original background image in an on-line manner and a fine background subtraction is performed to obtain a foreground mask that is least affected by abrupt changes of intensity level.

Apart from implementing robust background modelling, colour correction is also carried out in YUV colour space by blending the chrominance components of previous and current frames to reduce the chroma noise commonly found in surveillance videos.

Discussion – Many studies have been done in robust background modelling. There are quite a few methods that can handle gradual lighting changes but are still vulnerable to sudden lighting changes [207, 262]. In particular, they are based on statistical background modelling, which are slow in model update, thus being less effective in handling rapid lighting changes. Recently, a number of methods have been proposed to cope with sudden lighting changes [180,211,245]. The background subtraction method presented in this chapter is similar to that reported by Sung *et al*. [211] but with a key difference on the intensity ratio estimation. In their approach, a set of recent frames are kept to estimate a background model; therefore $\mathbf{g}$ is estimated between the current frame and previous frame. However, given surveillance videos captured in crowded scenes, it is hard to maintain a reliable background model with limited number of recent frames. Therefore, the proposed approach chooses to generate a single background image, and adjusts it based on $\mathbf{g}$ estimated between the current frame and the background image itself.

Local block activity pattern representation – First, the image space of a camera view is divided into equal-sized blocks with 10×10 pixels each (Figure 3.1(b)). Foreground pixels are then detected using the aforementioned background subtraction method. The foreground pixels are categorised as either static or moving via frame differencing (*e.g*. sitting people are detected as static foreground whilst passing-by people are detected as moving foreground). Activity patterns of a block are then represented as a bivariate time-series:

$$\begin{aligned} \mathbf{u_b} &= (u_{\mathbf{b},1}, \ldots, u_{\mathbf{b},t}, \ldots, u_{\mathbf{b},T}) \\ \mathbf{v_b} &= (v_{\mathbf{b},1}, \ldots, v_{\mathbf{b},t}, \ldots, v_{\mathbf{b},T}) \end{aligned}, \tag{3.4}$$

where $\mathbf{b}$ representing the two-dimensional coordinates of a block in the image space and T is the total number of frames used in the training process, whilst $u_{\mathbf{b},t}$ and $v_{\mathbf{b},t}$ are the percentages of static and moving foreground pixels within the block at frame t, respectively. Note that T needs to be sufficiently large to cover enough repetitions of activity patterns depending on the complexity of a scene.

The low spatial and temporal resolutions of surveillance footage have imposed great challenges to the selection of appropriate features for local block activity pattern representation. As explained in Section 1.2.1, trajectory features [108, 194] are extremely unreliable under these restrictions. More sophisticated features such as optical flow [231, 247] are found to be unstable too. Importantly, optical flow computation assumes small object displacement and constant brightness for the computation of velocity fields; both assumptions are invalid for videos with very low frame rate and poor image quality. Similarly, spatiotemporal gradients proposed by Kratz and Nishino [132] would fail due to motion discontinuities in low-frame rate videos.

Consequently $\mathbf{u_b}$ and $\mathbf{v_b}$ are chosen as they are the only features that can be extracted reliably given videos of low spatial and temporal resolution such as the dataset used in this work (Section 3.3.1). Despite their simplicity, time-series features $\mathbf{u_b}$ and $\mathbf{v_b}$ are found to be effective in capturing the temporal characteristics of activity patterns including temporal persistence of different patterns and their temporal order.

Activity-based scene decomposition – After feature extraction, blocks are grouped into regions according to the similarity of local spatiotemporal activity patterns represented as $\mathbf{u_b}$ and $\mathbf{v_b}$. Specifically, two blocks are considered similar and grouped together if they are close to each other spatially, and exhibit high correlations in both static and moving foreground activities over time. The grouping process begins with computing correlation distances among local activity patterns of each pair of blocks. A correlation distance is defined as a dissimilarity metric derived from the Pearson's correlation coefficient [143], r, given as:

$$\bar{r} = 1 - |r|, \tag{3.5}$$

In particular, $\bar{r} = 0$ if two blocks have strongly correlated local activity patterns, or $\bar{r} = 1$ otherwise. Subsequently, an affinity matrix $\mathbf{A} = [A_{ij}] \in \mathbb{R}^{B \times B}$ is constructed, where B is the total

number of blocks in the camera view and A_{ij} is defined as:

$$A_{ij} = \begin{cases} \exp\left(-\frac{(\bar{r}^{\mathbf{u}}_{ij})^2}{2\sigma^{\mathbf{u}}_i \sigma^{\mathbf{u}}_j}\right) \exp\left(-\frac{(\bar{r}^{\mathbf{v}}_{ij})^2}{2\sigma^{\mathbf{v}}_i \sigma^{\mathbf{v}}_j}\right) \exp\left(-\frac{\|\mathbf{b}_i - \mathbf{b}_j\|^2}{2\sigma^2_{\mathbf{b}}}\right) \\ \quad \text{if } \|\mathbf{b}_i - \mathbf{b}_j\| \leq R \text{ and } i \neq j \\ 0 \quad \text{otherwise} \end{cases}, \tag{3.6}$$

where the correlation distances of $\mathbf{u_b}$ and $\mathbf{v_b}$ between block i and block j are given by $\bar{r}^{\mathbf{u}}_{ij}$ and $\bar{r}^{\mathbf{v}}_{ij}$ respectively, whilst $[\sigma^{\mathbf{u}}_i, \sigma^{\mathbf{u}}_j]$ and $[\sigma^{\mathbf{v}}_i, \sigma^{\mathbf{v}}_j]$ are the respective correlation scaling factors for $\bar{r}^{\mathbf{u}}_{ij}$ and $\bar{r}^{\mathbf{v}}_{ij}$. The correlation scaling factors are defined as the mean correlation distance between the current block and all blocks within a radius R. The coordinates of the two blocks are denoted as $\mathbf{b}_i$ and $\mathbf{b}_j$. Similar to the correlation scaling factors, the spatial scaling factor $\sigma_{\mathbf{b}}$ is defined as the mean spatial distance between the current block and all blocks within the radius R. The affinity matrix is then normalised according to:

$$\overline{\mathbf{A}} = \mathbf{L}^{-\frac{1}{2}} \mathbf{A} \mathbf{L}^{-\frac{1}{2}}, \tag{3.7}$$

where $\mathbf{L}$ is a diagonal matrix and $L_{ii} = \sum_{j=1}^{B} A_{ij}$. Upon obtaining the normalised affinity matrix $\overline{\mathbf{A}}$, the spectral clustering method proposed by Zelnik-Manor and Perona [250] is employed to decompose each camera view into regions with the optimal number of regions being determined automatically.

In the computation of the affinity matrix, similarities are computed within a fixed radius R following a strategy reported by Li *et al.* [141], which was shown to be capable of preventing the under-fitting problem during decomposition in comparison to a local scaling strategy proposed by Zelnik-Manor and Perona [250]. Note that the similarity in the Gaussian kernel affinity matrix is governed by the selection of scaling factors [163, 250]. In this study, the scaling factors are functions of the radius R; the scene decomposition results are therefore governed by the selection of R. It is observed from experiments that the cluster formations are generally stable when R is set within the range of 20-30. Consequently, $R = 20$ is selected in this study. Figure 3.1(c) shows some examples of scene decomposition. It is observed that each camera view is decomposed into semantically relevant regions such as train track areas and people sitting areas.

The proposed scene decomposition method is similar to that presented by Li *et al.* [141] but with a noticeable modification on how local activities are represented. Specifically, in the

proposed method local activities are represented as time-series and correlation distance between them are used as the dissimilarity measure. In comparison, a bag of words representation is adopted by Li *et al.* [141], which ignores the temporal order information of a local activity and is thus less discriminative than time-series representation.

Regional activity representation – Given the scene decomposition, regional activity patterns of a camera view are formed based on the local block activity patterns. In particular, the regional activity patterns at region $\mathcal{R}_i$ are represented as:

$$\begin{aligned} \hat{\mathbf{u}}_i &= \frac{1}{|\mathcal{R}_i|} \sum_{\mathbf{b} \in \mathcal{R}_i} \mathbf{u}_{\mathbf{b}} = (\hat{u}_{i,1}, \ldots, \hat{u}_{i,t}, \ldots, \hat{u}_{i,T}) \\ \hat{\mathbf{v}}_i &= \frac{1}{|\mathcal{R}_i|} \sum_{\mathbf{b} \in \mathcal{R}_i} \mathbf{v}_{\mathbf{b}} = (\hat{v}_{i,1}, \ldots, \hat{v}_{i,t}, \ldots, \hat{v}_{i,T}) \end{aligned}, \tag{3.8}$$

where $|\mathcal{R}_i|$ is the number of blocks belonging to region $\mathcal{R}_i$. To facilitate a more accurate time delayed correlation analysis, any region with half of its blocks exhibiting low activity is removed. In this study, a low-activity block is defined as a block with activity patterns having a standard deviation that is less than three.

3.1.2 Cross Canonical Correlation Analysis

For any pair of regions in a camera network, two questions are to be answered: (1) are activities in these regions correlated? (2) if yes, how strong are the correlations and what are the temporal relationships among them? It is nontrivial to discover and quantify correlations and temporal relationships between cameras. Different viewing angles of cameras may introduce pattern variations across camera views. Importantly, pairwise correlations between regional activities across non-overlapping camera views are complex in that there is often an unknown temporal gap/delay between the times when a causing activity in one region taking place and the correlated/caused activity in the other region being observed.

To this end, a new xCCA is formulated to measure the correlation of two regional activities as a function of an unknown time lag τ applied to one of the two regional activity time-series. The aim is to search for combinations of the regional activities (represented as time-series) having maximal correlation and model the temporal gap as a temporal dependency of arbitrary time delay between the two time-series.

The xCCA differs from Canonical Correlation Analysis (CCA) [105] in that CCA can only measure how strong two vector variables are correlated in a concurrent or zero-order sense. The

proposed xCCA extends CCA to measure correlations beyond zero order by including additional steps similar in nature to the standard Cross Correlation Analysis (xCA) [128]. This principally involves shifting of one time series and computes its canonical correlation with the other. An example is shown in Figure 3.1(d). The xCCA analysis compares favourably to the standard xCA. In comparison to xCA, xCCA is more capable of capturing the underlying mutual patterns of two regional activity time series. This is because by projecting them into an optimal subspace, it minimises the effect of pattern variations introduced by different camera viewing angles and the temporal delays between correlated activities across camera views.

Formally, let $\mathbf{x}_i(t)$ and $\mathbf{x}_j(t)$ denote the two regional activity time series observed in the ith and jth regions respectively. Note that $\mathbf{x}_i(t)$ and $\mathbf{x}_j(t)$ are time-series of n_f-dimensional variables. In this study, $n_f = 2$ since two features $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ are extracted from each region, *i.e.* $\mathbf{x}_i(t) = (\hat{u}_{i,t}, \hat{v}_{i,t})$. For clarity in the following equations, $\mathbf{x}_j(t+\tau)$ is denoted as $\mathbf{y}(t)$. The symbol t is also omitted for conciseness, *e.g.* time series $\mathbf{x}_i(t)$ becomes $\mathbf{x}_i$ and $\mathbf{y}(t)$ becomes $\mathbf{y}$.

At each time delay index τ or each shifting step, the xCCA finds two sets of optimal basis vectors $\mathbf{w}_i$ and $\mathbf{w}_j$ for $\mathbf{x}_i$ and $\mathbf{y}$ such that correlation of the projections of them onto the basis vectors are mutually maximised. Let linear combinations of canonical variates be $\tilde{x}_i = \mathbf{w}_i^{\mathsf{T}}\mathbf{x}_i$ and $\tilde{y} = \mathbf{w}_j^{\mathsf{T}}\mathbf{y}$, canonical correlation $\boldsymbol{\rho}_{ij}(\tau)$ is defined as:

$$\begin{aligned}
\boldsymbol{\rho}_{ij}(\tau) &= \frac{\mathrm{E}[\tilde{x}_i\tilde{y}]}{\sqrt{\mathrm{E}[\tilde{x}_i^2]\mathrm{E}[\tilde{y}^2]}} \\
&= \frac{\mathrm{E}[\mathbf{w}_i^{\mathsf{T}}\mathbf{x}_i\mathbf{y}^{\mathsf{T}}\mathbf{w}_j]}{\sqrt{\mathrm{E}[\mathbf{w}_i^{\mathsf{T}}\mathbf{x}_i\mathbf{x}_i{}^{\mathsf{T}}\mathbf{w}_i]}\sqrt{\mathrm{E}[\mathbf{w}_j^{\mathsf{T}}\mathbf{y}\mathbf{y}^{\mathsf{T}}\mathbf{w}_j]}} \\
&= \frac{\mathbf{w}_i^{\mathsf{T}}\mathbf{C}_{ij}\mathbf{w}_j}{\sqrt{\mathbf{w}_i^{\mathsf{T}}\mathbf{C}_{ii}\mathbf{w}_i}\sqrt{\mathbf{w}_j^{\mathsf{T}}\mathbf{C}_{jj}\mathbf{w}_j}},
\end{aligned} \tag{3.9}$$

where $\mathbf{C}_{ii}$ and $\mathbf{C}_{jj}$ are within-set covariance matrices of $\mathbf{x}_i$ and $\mathbf{y}$, respectively, whilst $\mathbf{C}_{ij}$ is between-set covariance matrix.

The maximisation of $\boldsymbol{\rho}_{ij}(\tau)$ at each time delay index τ can be solved by setting the derivatives in Equation (3.9) to zero, yielding the following eigenvalue equations:

$$\begin{cases} \mathbf{C}_{ii}^{-1}\mathbf{C}_{ij}\mathbf{C}_{jj}^{-1}\mathbf{C}_{ji}\mathbf{w}_i = \boldsymbol{\rho}_{ij}^2(\tau)\mathbf{w}_i \\ \mathbf{C}_{jj}^{-1}\mathbf{C}_{ji}\mathbf{C}_{ii}^{-1}\mathbf{C}_{ij}\mathbf{w}_j = \boldsymbol{\rho}_{ij}^2(\tau)\mathbf{w}_j \end{cases}, \tag{3.10}$$

where the eigenvalues $\boldsymbol{\rho}_{ij}^2(\tau)$ are the square canonical correlations and the eigen vectors $\mathbf{w}_i$ and

$\mathbf{w}_j$ are the basis vectors. Only one of the eigenvalue equations needs to be solved since the equations are related by:

$$\begin{cases} \mathbf{C}_{ij}\mathbf{w}_j = \boldsymbol{\rho}_{ij}(\tau)\lambda_i\mathbf{C}_{ii}\mathbf{w}_i \\ \mathbf{C}_{ji}\mathbf{w}_i = \boldsymbol{\rho}_{ij}(\tau)\lambda_j\mathbf{C}_{jj}\mathbf{w}_j \end{cases}, \tag{3.11}$$

where

$$\lambda_i = \lambda_j^{-1} = \sqrt{\frac{\mathbf{w}_j^{\mathsf{T}}\mathbf{C}_{jj}\mathbf{w}_j}{\mathbf{w}_i^{\mathsf{T}}\mathbf{C}_{ii}\mathbf{w}_i}}. \tag{3.12}$$

The time delay that maximises the canonical correlation between $\mathbf{x}_i(t)$ and $\mathbf{x}_j(t)$ is computed as:

$$\hat{\tau}_{ij} = \underset{\tau}{\operatorname{argmax}} \frac{\sum^{\Gamma}\boldsymbol{\rho}_{ij}(\tau)}{\Gamma}, \tag{3.13}$$

where $\Gamma = \min(\operatorname{rank}(\mathbf{x}_i), \operatorname{rank}(\mathbf{x}_j))$. Note that the canonical correlation function is averaged with Γ to obtain a single correlation value at each time delay index. The associated maximum canonical correlation is then obtained by locating the peak value in the averaged canonical correlation function as:

$$\hat{\rho}_{ij} = \frac{\sum^{\Gamma}\boldsymbol{\rho}_{ij}(\hat{\tau}_{ij})}{\Gamma}. \tag{3.14}$$

The maximum canonical correlation and the associated time delay for each pair of regional activity patterns are computed to construct a regional activity affinity matrix:

$$\mathbf{P} = [\hat{\rho}_{ij}]_{n\times n}, \tag{3.15}$$

and a time delay matrix

$$\mathbf{D} = [\hat{\tau}_{ij}]_{n\times n}. \tag{3.16}$$

Note that $0 \leq \hat{\rho}_{ij} \leq 1$ with equality to 1 if, and only if the two regional time series are identical. If $\tau = 0$, xCCA is equivalent to performing CCA on $\mathbf{x}_i(t)$ and $\mathbf{x}_j(t)$.

3.1.3 Computational Cost Analysis

The computational cost of each component of the proposed approach is analysed below. The actual run time cost will be given in Section 3.3.4.

- Activity-based scene decomposition: A total of $B(B-1)/2$ computations are required to obtain the pairwise correlation distances among local activity patterns of each block pair. The spectral clustering involves computation of eigenvectors of affinity matrix $\mathbf{A}$

(Equation (3.6)) with computational complexity of $O(B^3)$.

- Cross Canonical Correlation Analysis: To obtain each element in a regional activity affinity matrix (Equation (3.15)) and a time delay matrix (Equation (3.16)), a regional time-series is shifted against another regional time-series and canonical correlation is performed at each shifting step. A total of $2T-1$ shifting steps are required where T is the total of training frames. However, if one bounds the maximum time delay as $\tau_{\max}$, the total number of shifting steps can be reduced to $\tau_{\max}-1$. The computational cost of the CCA is dominated by singular value decomposition (SVD). However, since Γ in Equation (3.13) is small, the complexity of SVD, $O(\Gamma^3)$ is low.

3.2 Applications

3.2.1 Topology Inference

A camera topology can be estimated once the pairwise regional activity correlations of arbitrary time delays are discovered and quantified (Equations (3.15) and (3.16)). It is observed that considerably high correlations may be found between some region pairs even though they are not close to each other both spatially and temporally, owing to noise or constant crowdedness in both regions. As a result, when shifting is performed to compute the correlation function of two regional time-series, the algorithm may locate a 'spurious' peak that does not reflect the true correlation among their activity patterns. Fortunately in most cases, those 'spurious' peaks are found at a point where the time delay has a large value. Therefore, both time delay and correlation strength are exploited for topology inference in the proposed approach. Specifically, two cameras will be connected in the inferred topology if they contain connected regions, which are defined as those with high correlation value (Equation (3.14)) and short time delay (Equation (3.13)).

First, a region connectivity matrix $\mathbf{\Psi} = [\Psi_{ij}]_{n\times n}$ is computed to represent how likely each pair of regions in the camera network are connected. More specifically, each element in the region connectivity matrix, or the strength of connectivity is computed as:

$$\Psi_{ij} = \overline{\hat{\rho}_{ij}}\left(1-\overline{|\hat{\tau}_{ij}|}\right), \tag{3.17}$$

where $\overline{\hat{\rho}_{ij}}$ is obtained from normalised regional activity affinity matrix $\mathbf{P}$, so that it has a value range of $[0,1]$. Whilst $\overline{|\hat{\tau}_{ij}|}$ is obtained by normalising the absolute values of the elements of the

time delay matrix $\mathbf{D}$. These two normalisations ensure one has $0 \leq \Psi_{ij} \leq 1$. The higher the value of Ψ_{ij}, the stronger the connectivity between a region pair .

Once the region connectivity matrix is obtained, the camera topology, represented as a camera connectivity matrix $\mathbf{\Phi} = [\Phi_{ij}]_{m \times m}$, can be inferred. Specifically the strength of the connectivity between the ith and jth camera nodes is obtained by averaging the regional activity connectivity strength (Equation (3.17)) between each pair of regions across the two camera views. In order to reduce the influence of possible noise and redundant connectivities in $\mathbf{\Psi}$, the strongest connectivity between a region in the ith camera view with all regions in the jth camera view is first identified. This searching step is repeated for all regions in the ith camera view. Subsequently, the top n_{e} connectivities are averaged to obtain Φ_{ij}, with n_{e} being set to half of the number of regions in the ith camera view. Finally $\mathbf{\Phi}$ is normalised so that its elements have a value range of [0,1]. Two cameras are then deemed as being connected if the corresponding Φ_{ij} value is greater than the mean value of all the elements of $\mathbf{\Phi}$.

3.2.2 Context-aware Person Re-identification

The goal of person re-identification is to search for a given individual who disappeared in one camera view over other camera views. This section describes a way of using the learned time delayed activity correlations as contextual information for reducing the search space and resolving ambiguities arising from:

1. Similar visual features presented by different people.

2. Feature variations caused by different poses, camera viewing angles and illumination changes.

A simple colour histogram feature is used for discriminating an individual against others. Though more sophisticated features are available [69, 82, 89], the use of simple feature provides a baseline for evaluating to what extent the time delayed correlations could improve the person re-identification accuracy. Specifically, given bounding boxes of two people a and b observed in different camera views, the bounding boxes are first normalised to equal size and subsequently segmented into n_{h} horizontal strips of equal height, from which colour histograms are computed and concatenated for representing the visual appearances of persons a and b.

The similarity between the two concatenated colour histograms H^a and H^b of persons a and b is measured using Bhattacharyya score [19,49] as follows:

$$S_{\text{bha}}^{a,b} = \sum_{i=1}^{n_{\text{bin}}} \sqrt{H_i^a H_i^b}, \tag{3.18}$$

where n_{bin} represents the number of bins in a histogram. Each histogram bin is normalised using the total number of pixels in the normalised image, so that $\sum_{i=1}^{n_{\text{bin}}} H_i^a = 1$ and $\sum_{i=1}^{n_{\text{bin}}} H_i^b = 1$. Note that the Bhattacharyya score is close to zero (minimum value is 0) if H^a and H^b are very different, or have a maximum value of 1 if the two histograms are identical. The Bhattacharyya score is first computed for each colour channel separately. The overall Bhattacharyya score $\overline{S}_{\text{bha}}^{a,b}$ is then obtained by multiplying the scores $S_{\text{bha}}^{a,b}$ computed over all channels.

To incorporate the learned pairwise activity correlations and time delays into the final score for person re-identification, the regions (see Section 3.1.1) occupied by person a and b and the associated inter-region correlation and time delay are first determined. In particular, if a person's bounding box overlaps n_{r} regions in the image space, the occupancy fractions of individual regions within the bounding box are computed and represented as a set of weights:

$$\boldsymbol{\mu} = \{\mu_i | i = 1, \ldots, n_{\text{r}}\}, \tag{3.19}$$

where $\sum_{i=1}^{n_r} \mu_i = 1$. The weights are used to calculate the correlation between regions occupied by persons a and b as follows:

$$\hat{\rho}^{a,b} = \sum_{i=1}^{n_{\text{r}}^a} \mu_i^a \left(\sum_{j=1}^{n_{\text{r}}^b} \mu_j^b \, \hat{\rho}_{g(i)\,g(j)} \right), \tag{3.20}$$

where $\hat{\rho}_{g(i)\,g(j)}$ is the maximum cross canonical correlation computed using Equation (3.14), with $g(\cdot)$ as a mapping function that maps the local regional index i and j to the corresponding global regional index, with $1 \leq g(\cdot) \leq n$. The associated time delay is given as:

$$\hat{\tau}^{a,b} = \sum_{i=1}^{n_{\text{r}}^a} \mu_i^a \left(\sum_{j=1}^{n_{\text{r}}^b} \mu_j^b \, \hat{\tau}_{g(i)\,g(j)} \right), \tag{3.21}$$

where $\hat{\tau}_{g(i)g(j)}$ is computed using Equation (3.13). The overall score is obtained as follows:

$$S^{a,b}_{\text{overall}} = \begin{cases} \overline{S}^{a,b}_{\text{bha}}\,\hat{\rho}^{a,b} & \text{if } 0 < t^{a,b}_{\text{gap}} < \alpha\hat{\tau}^{a,b} \\ 0 & \text{otherwise} \end{cases}, \tag{3.22}$$

where $t^{a,b}_{\text{gap}}$ is the time gap of observing the two people in the two camera views, whilst α is a factor that determines the maximum allowable transition time between cameras during person matching.

3.2.3 Activity-based Temporal Segmentation

The goal of activity-based temporal segmentation is to segment a continuous video according to the activity captured in a video [242], *e.g.* to segment a traffic surveillance video into different phases such as leftward traffic and rightward traffic in accordance to vehicle movements.

For robust and accurate temporal segmentation in a camera network, it is useful to model correlated activities collectively across multiple camera views. This is because by utilising visual evidence collected from different views, video temporal segmentation is more robust to noise and visual ambiguities than modelling activities separately within individual camera views. Note that the regional activity affinity matrix **P** (Equation (3.15)) is only concerned with the pairwise correlations of regional activities. It does not reveal either the contributions of these regional activities to the actual correlated activities or the associated temporal dynamics one wishes to model.

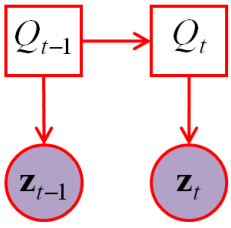


Figure 3.3: Hidden Markov model with two time slices unrolled. Observation nodes are shown as shaded circles and hidden nodes are shown as clear squares.

A complex camera network may capture many activities occurring simultaneously. However, not all the activities are correlated and they should be excluded during multi-view activity modelling. In this study, the underlying correlated activities are discovered and modelled by taking the following steps:

1. The same spectral clustering algorithm used in Section 3.1.1 is employed to group regional activities using the regional activity affinity matrix $\mathbf{P}$ (Equation (3.15)) as an input affinity matrix.

2. The clusters returned by the spectral clustering algorithm are examined for discovering highly-correlated activities. Specifically, those clusters that consist of cross-camera regions with the highest mean cross canonical correlations are selected.

3. Activity patterns in one of the n_{selected} selected regions are set as a reference point to temporally align activity patterns of other regions in accordance to the respective temporal offsets $\hat{\tau}_{ij}$ computed using Equation (3.13).

4. The aligned regional activity patterns, each represented as a two-dimensional time series (*i.e.* $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$), are concatenated together to form a joint correlated activity pattern:

$$\mathbf{z}_t = \hat{\mathbf{u}}_{1,t}||\hat{\mathbf{v}}_{1,t}||\ldots||\hat{\mathbf{u}}'_{n_{\text{selected}},t}||\hat{\mathbf{v}}'_{n_{\text{selected}},t}, \tag{3.23}$$

   with the prime symbol indicating an aligned time-series according to the temporal offset. The time-aligned correlated activity patterns, $\mathbf{z}_t$ are then used as inputs to train a K-hidden states hidden Markov model (HMM) to model the temporal dynamics of the correlated activity patterns.

5. The HMM structure is shown in Figure 3.3. It is an ergodic (fully-connected) model with Q_t being discrete random variable, $Q_t \in \{q^i | i = 1, \ldots, K\}$. The model is assumed to be a first-order Markov model, *i.e.* $p(Q_t|Q_{1:t-1}) = p(Q_t|Q_{t-1})$. The transition between observations are also assumed to be a conditionally first-order Markov process, *i.e.* $p(\mathbf{z}_t|Q_t, \mathbf{z}_{1:t-1}) = p(\mathbf{z}_t|Q_t)$.

6. Automatic model selection is performed based on the Bayesian Information Criterion (BIC) score to find the number of hidden states K in the model. For model parameter estimation, the aligned regional activity patterns at different time instances are first grouped into K groups using a K-means clustering algorithm. The clustering results, *i.e.* the means and covariances of individual groups are used to initialise a K-hidden states HMM. The model parameters are then estimated using the Baum-Welch algorithm [13].

The learned HMM for each set of correlated activities can be used for real-time activity-based

temporal segmentation. The objective is to segment an unseen video stream into activity phases based on 'what is happening' not only in a particular view but also in other views with highly-correlated activities. These activity phases are obtained by inferring the hidden states Q_t at each time instance using an on-line filtering method [160]. In particular, given $\mathbf{z}_t$ observed as a continuous data stream, the probability of a particular hidden state $p(Q_t|\mathbf{z}_{1:t})$ is computed as a function of current input $\mathbf{z}_t$ and prior belief state $p(Q_{t-1}|\mathbf{z}_{1:t-1})$:

$$\begin{aligned} p(Q_t|\mathbf{z}_{1:t}) &\propto p(\mathbf{z}_t|Q_t,\mathbf{z}_{1:t-1})p(Q_t|\mathbf{z}_{1:t-1}) \\ &= p(\mathbf{z}_t|Q_t)[\textstyle\sum_{Q_{t-1}} p(Q_t|Q_{t-1})p(Q_{t-1}|\mathbf{z}_{1:t-1})]. \end{aligned} \tag{3.24}$$

Based on the Markovian assumption, $p(\mathbf{z}_t|Q_t,\mathbf{z}_{1:t-1})$ can be replaced with $p(\mathbf{z}_t|Q_t)$. Similarly, $p(Q_t|\mathbf{z}_{1:t-1})$ can be computed from the prior belief state under the Markovian assumption. To infer the activity phase Q_t^*, the probabilities $p(Q_t = q^i|\mathbf{z}_{1:t})$ are first computed using Equation (3.24). The most likely hidden state is then determined by choosing the hidden state that yields the highest probability:

$$Q_t^* = \underset{q^i}{\operatorname{argmax}}\ p(Q_t = q^i|\mathbf{z}_{1:t}). \tag{3.25}$$

3.3 Experiments

3.3.1 Datasets

The two datasets employed in experiments of this chapter contain synchronised and static views, captured at a frame rate of 0.7 fps from uncalibrated and disjoint cameras installed at two busy underground stations. Each image frame has a size of 320×230 pixels. Detailed descriptions on each dataset are given as follows:

Station A dataset – A snapshot of each of the eight camera views and the camera topology of this station are depicted in Figure 3.4. The two train platforms of this station are covered by three cameras each (Cam 1-6). Another two cameras (Cam 7-8) monitor a connected concourse, which is far away from the two platforms. The video from each camera lasts over 19 hours from 5:28am to 12:38am the next day, giving a total of 153 hours of video footage. Typically, when a train arrives at one of the platforms, passengers on the train get off and leave the platform whilst passengers waiting on the platform get into the train. Nonetheless, it is also common that some passengers remain staying at the platform to wait for a later train to a different destination.

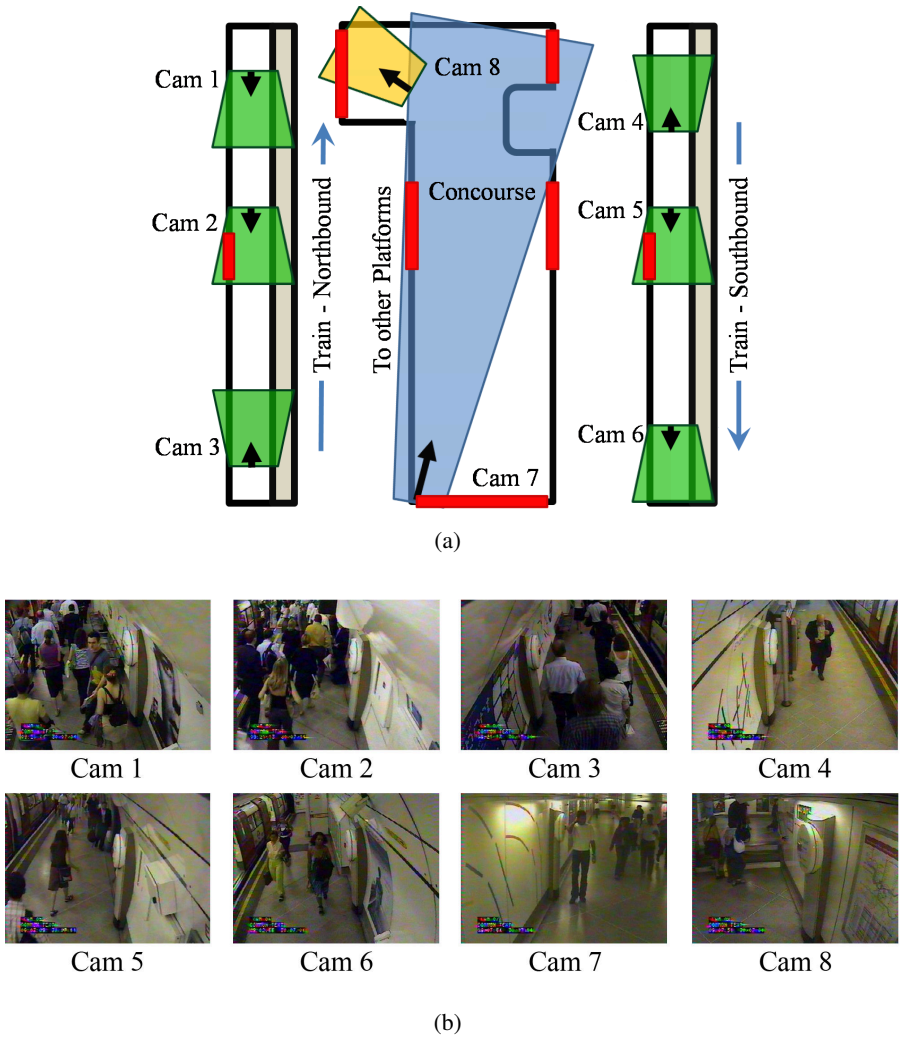


Figure 3.4: (a) The station layout and camera topology of Station A dataset. (b) Example frames of each camera view. Entry and exit points are highlighted in red bars.

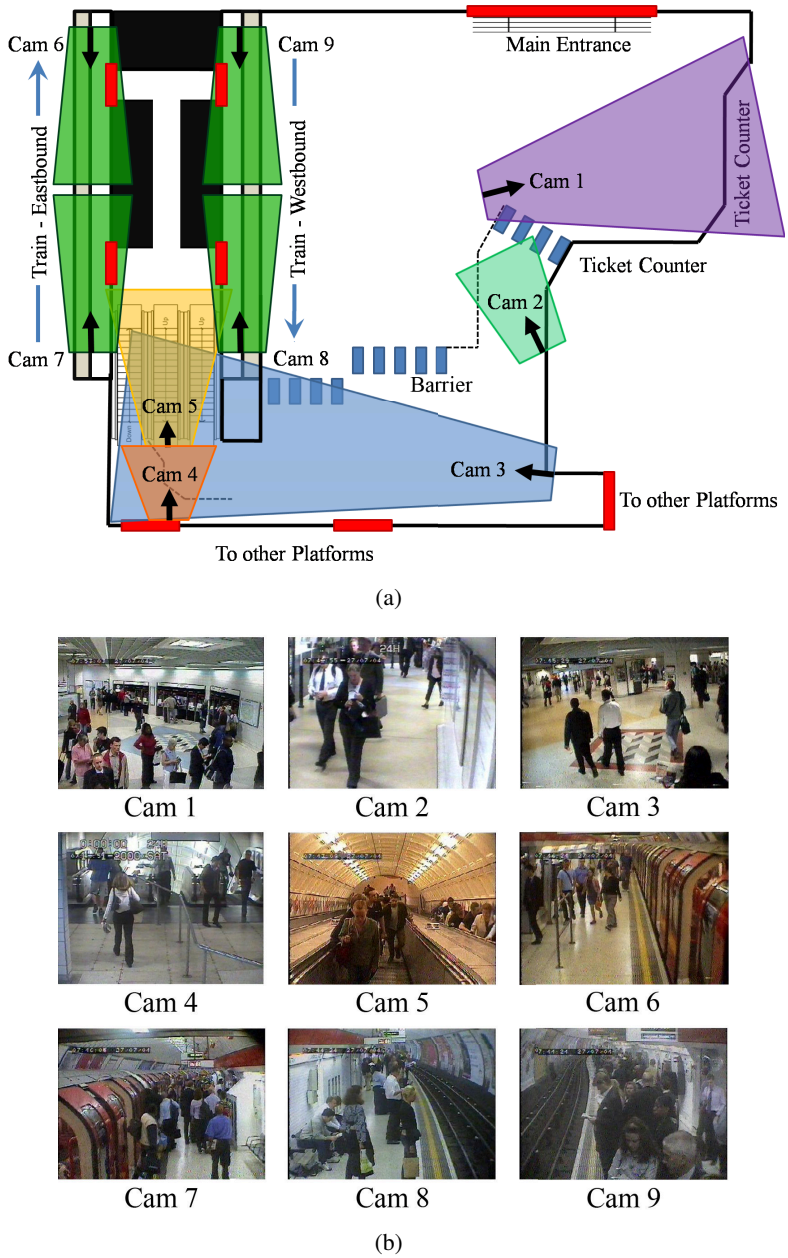


Figure 3.5: (a) The station layout and camera topology of Station B dataset. (b) Example frames of each camera view. Entry and exit points are highlighted in red bars.

Station B dataset – The camera topology of this station is shown in Figure 3.5, alongside sample images of nine camera views. The station has a ticket hall and a concourse leading to two train platforms via escalators. Three cameras are placed in a ticket hall and two cameras are positioned to monitor the escalator areas. Both train platforms are covered by two cameras each. The video from each camera lasts around 20 hours from 5:42am to 01:19am the next day, giving a total of 177 hours of video footage. Typically, passengers enter from the main entrance, walk through the ticket hall or queue up for tickets (Cam 1), enter the concourse through the ticket barriers

(Cam 2, 3), take the escalators (Cam 4, 5), and enter one of the platforms. The opposite route is taken if they are leaving the station. Apart from the two platforms in Cam 6-7 and Cam 8-9, the passengers may also proceed from the concourse to other platforms (not visible in the camera views) without taking the escalators. In addition, after getting off a train they may also go to a different platform without leaving the station.

The two datasets employed are different in that Station A dataset has larger time gaps between cameras, thus it is more challenging for the person re-identification task. Whilst Station B dataset features more diverse scenes and complex activities, hence it is more ideal for experiments in topology inference. In general, both datasets are difficult in several aspects:

1. Complexity and diversity of the scenes. Activities observed in the scenes take place at ticket hall, concourse, train platforms and escalators.

2. Low video temporal and spatial resolutions.

3. The lighting conditions are very different across camera views.

4. Heavy inter-object occlusions due to the enormous number of objects in the scene especially during peak hours.

5. Complex crowd dynamics, *e.g.* passengers may appear in a group or individually, remain stationary at any point in the scene, or do not get on an arrived train.

6. Only limited areas of the two large underground stations are covered by the cameras. In particular, there are multiple entry and exit points that are not visible in the camera views. This increases the uncertainties in the interpretation of the observed activities.

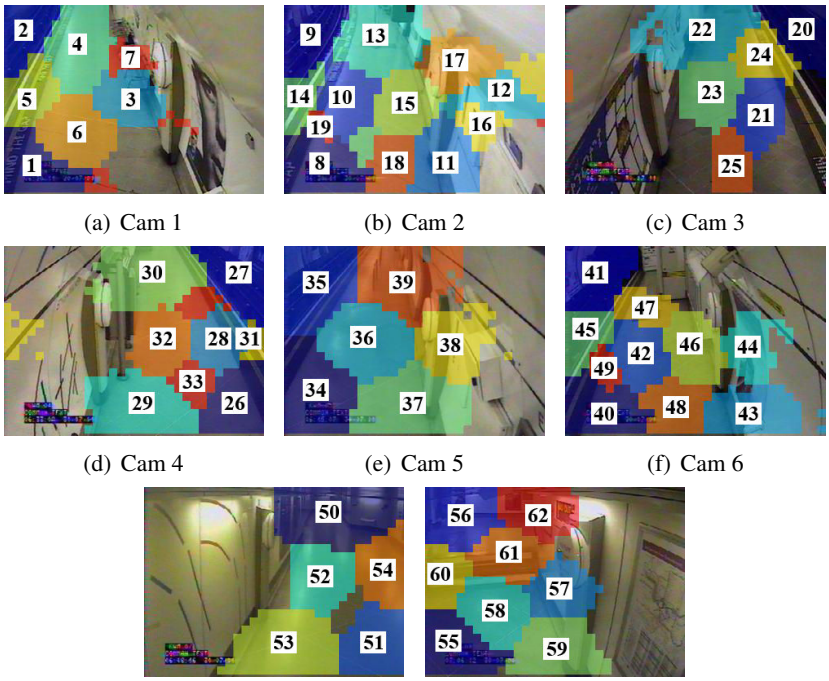
3.3.2 Background Subtraction

A comparison between the proposed mean-shift based background subtraction method and the frame differencing based method (see Section 3.1.1) was carried out. Some qualitative results are shown in Figure 3.6. As can be seen, foreground masks yielded by the proposed method are noticeably better. Apart from qualitative evaluation, quantitative evaluation on both methods was also performed on topology inference task. The results are reported in Section 3.3.5.

Figure 3.6: The figure shows background models (the second column) and foreground masks (the third column) yielded by frame differencing method without background adjustment (the first row) and the proposed approach (the second row) on frames with abrupt global intensity level change (the first column).

3.3.3 Activity-based Scene Decomposition

A total of 5000 frames ($\approx$ 2-hour in length) from each camera view were used for activity-based scene decomposition. In particular, the eight camera views from Station A dataset were automatically decomposed into 62 regions (Figure 3.7). Whilst the nine camera views from Station B dataset were decomposed into 96 regions (Figure 3.8). As can be seen from Figure 3.7 and Figure 3.8, the camera views were decomposed automatically into semantically relevant regions in spite of the heavy inter-object occlusions and low temporal resolution. For instance, the areas corresponding to the train tracks and platforms formed distinctive regions. The sitting areas (*e.g.* regions 3 and 7 of Station A dataset, regions 80 and 86 of Station B dataset) were also segmented from areas where people standing or walking. Another example is the different escalators exits (regions 40, 43, 46) in Station B dataset, which were clearly decomposed into



(a) Cam 1 (b) Cam 2 (c) Cam 3

(d) Cam 4 (e) Cam 5 (f) Cam 6

(g) Cam 7 (h) Cam 8

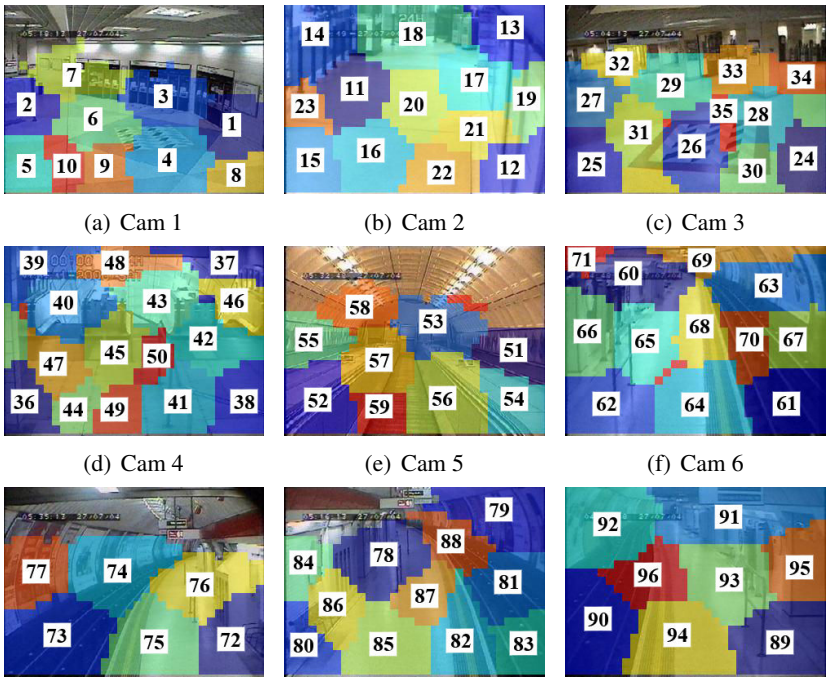
Figure 3.7: Station A dataset: activity-based scene decomposition results.

different regions in accordance to the object dynamics.

Both qualitative and quantitative comparisons between scene decomposition method introduced by Li *et al.* [141] and the proposed method were conducted. The two methods differ mainly in their feature representations, *i.e.* time-series representation in the proposed method and bag of words representation in that of Li *et al.*'s method [141].

1. Qualitative result: The time-series representation was found to yield more semantically relevant region boundaries. As can be seen from most of the camera views depicted in column (a) of Figure 3.9, the train track regions were clearly separated from the platform regions using the time-series representation. In contrast, some train track areas and platforms were segmented as a single region using the bag of words representation (column (b) in Figure 3.9).

2. Quantitative result: It is difficult to provide quantitative result on activity-based scene decomposition as the correct region segmentation is subjective, especially when the segmentation is not based on visual information but activity patterns observed over time. Therefore, quantitative evaluation was performed on a synthetic dataset, in which the segmentation ground truth is known. In the dataset, all blocks (10×10 pixels each) within the same region encompassed similar time-series patterns, whilst blocks located in different



(a) Cam 1 (b) Cam 2 (c) Cam 3

(d) Cam 4 (e) Cam 5 (f) Cam 6

(g) Cam 7 (h) Cam 8 (i) Cam 9

Figure 3.8: Station B dataset: activity-based scene decomposition results.

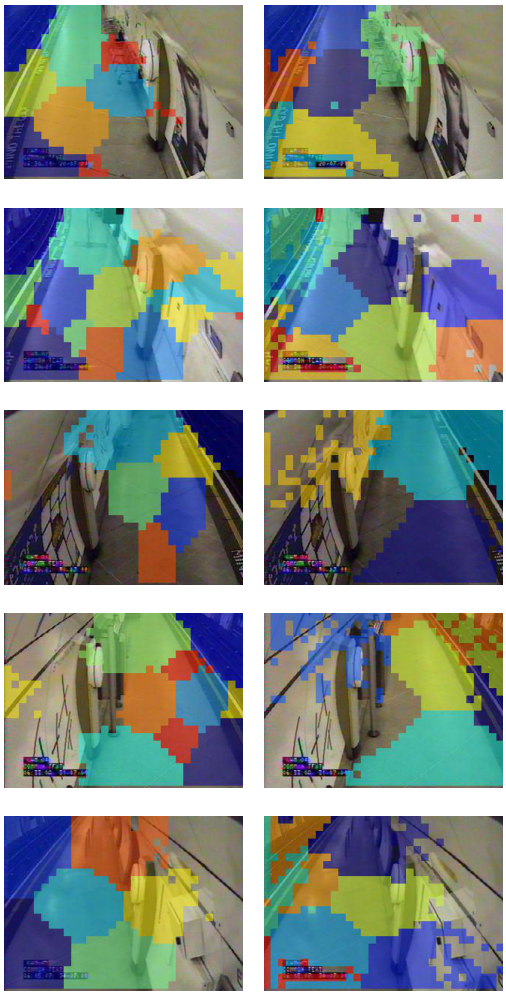
regions contained different time-series patterns. All time series had a length of 5000 and were corrupted by Gaussian noise. The decomposition accuracy was measured based on the agreement between estimated segmentation mask and the segmentation ground truth. As can be seen from Figure 3.10, the proposed time-series representation yielded higher accuracy, 99.73% compared to 83.83% obtained by using the bag of words representation presented by Li *et al*. [141].

As explained in Section 3.1.1, better performance is obtained because time-series activity representation captures the temporal dynamics of activity whilst the bag of words representation utilised by Li *et al*. [141] ignores the temporal order of the activity occurrences.

3.3.4 Pairwise Activity Correlation Analysis

Discovering and quantifying regional activity correlation – The proposed xCCA was compared with xCA for learning regional activity correlations. The regional activity affinity matrices **P** (Equation (3.15)) (normalised to have a value range of [0,1]) and the time delay matrices **D** (Equation (3.16)) yielded by different methods for Station A and Station B datasets are shown in Figure 3.11 and Figure 3.12 respectively.

It can be seen from Figure 3.11 and 3.12 that all methods were able to discover high correlations and short time delays between regions from the same camera views (see the block structure



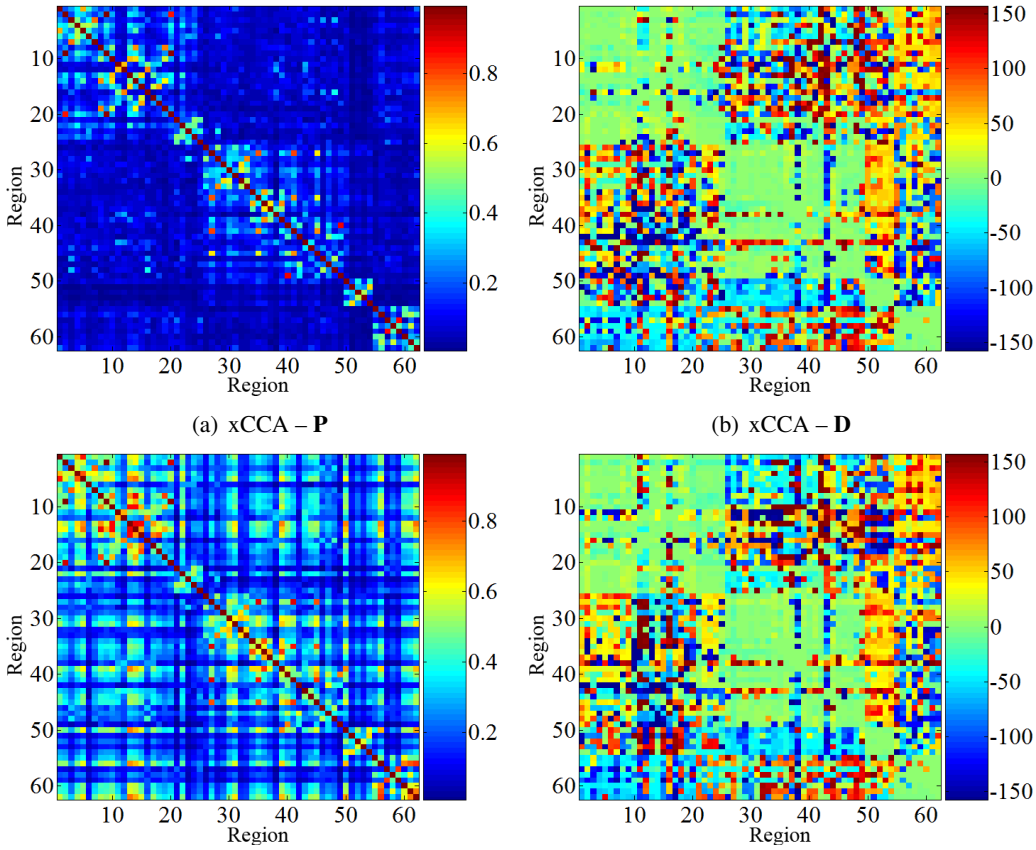
(a) (b)

Figure 3.9: Semantically more relevant scene decomposition (a) was obtained using the proposed time-series activity representation and correlation based distance metric, as compared to the result (b) obtained using the bag of words representation [141].

(a) Ground truth (b) Time-series (c) Bag of words

Figure 3.10: Quantitative comparison between the proposed time-series representation (decomposition accuracy = 99.73%) against bag of words representation [141] (decomposition accuracy = 83.83%) on a synthetic dataset.

along the diagonals of the **P** matrices). Importantly, a number of interesting cross-camera correlations were discovered and quantified accurately by xCCA. For instance, in Station B dataset, high correlation value (see Figure 3.12(a)) with a time delay of 9 frames or 13 seconds (see Figure 3.12(b)) were discovered by xCCA between region 46 (Cam 4) and region 51 (Cam 5). This corresponded to the frequently occurred inter-camera activity of passengers taking the upward



(a) xCCA – **P** (b) xCCA – **D**

(c) xCA – **P** (d) xCA – **D**

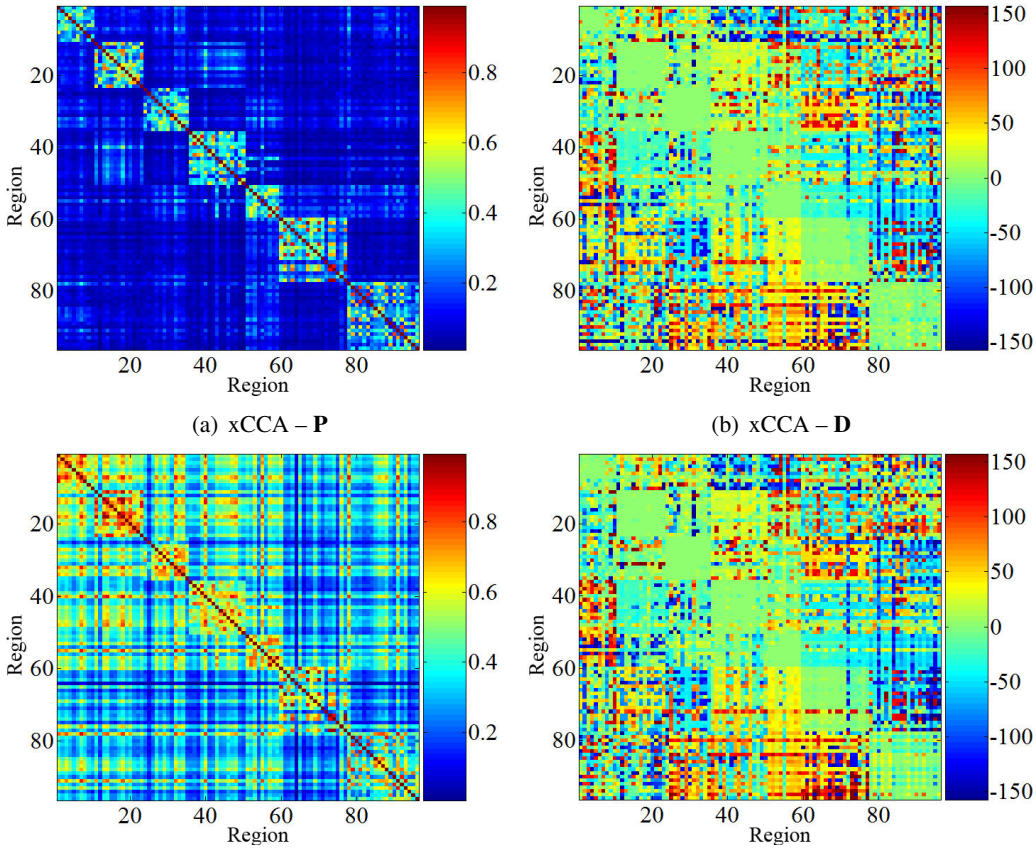
Figure 3.11: Station A dataset: regional activity affinity matrices **P** (normalised to have a value range of [0,1]) and the associated time delay matrices **D** obtained using xCCA and xCA.

escalator (with part of the escalator invisible from the view), and leaving from the escalator exit (see Figure 1.3(a)). In comparison, although xCA was also able to find these correlations, it tended to 'over-correlate' regions, *i.e.* detect pairwise correlations that do not exist (*e.g.* region pairs 3–91 and 18–48).

Computational cost – For activity-based scene decomposition, the computation of correlation distances in Matlab takes approximately six minutes on each camera view, whilst the spectral clustering implemented in C code requires seven seconds on a 2.8 GHz single-core machine. For xCCA, Matlab implementation takes approximately 12 minutes on Station A dataset and 30 minutes on Station B datasets.

3.3.5 Topology Inference

Given the regional activity affinity matrices and time delay matrices yielded by different methods, the camera topologies $\mathbf{\Phi}$ were generated by following the steps described in Section 3.2.1. The camera topologies for both Station A and Station B dataset are shown in Figure 3.13 and



(a) xCCA – **P**

(b) xCCA – **D**

(c) xCA – **P**

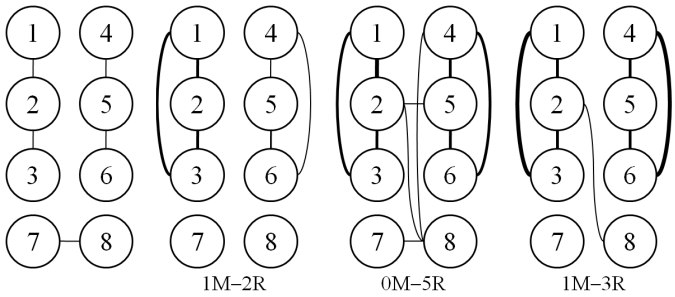
(d) xCA – **D**

Figure 3.12: Station B dataset: regional activity affinity matrices **P** (normalised to have a value range of [0,1]) and the associated time delay matrices **D** obtained using xCCA and xCA. The block structure along the diagonals of the **P** matrices indicate high correlations between regions located in the same camera view. There should be nine blocks in the diagonal for Station B dataset. However, only seven blocks were visible because Cam 6 and 7 formed a single block, whilst Cam 8 and 9 formed another block due to their high regional correlations between camera pair.

Figure 3.14 respectively. The inferred topologies were compared with the actual topology obtained manually and the numbers of missing edges (M) and redundant edges (R) are shown in the two figures.

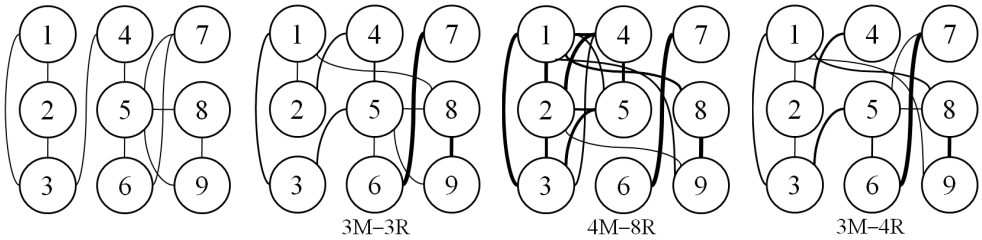
For both datasets, it is observed that the xCCA yielded the closest topology to the actual one based on the M and R metrics. As expected, xCA yielded a number of redundant edges in both datasets. The better performance of xCCA compared to xCA is due to its ability in capturing the underlying mutual patterns of two regional activity time series by projecting them onto an optimal subspace. This is critical for analysing a busy public space such as an underground station where significant variations exist for correlated activities in different views caused by different camera view angles and uncertainties on activity time delays between views.

Let us discuss those missing and redundant edges in the topologies. For Station A dataset,



(a) Actual (b) xCCA (c) xCA (d) Full Frame

Figure 3.13: Station A dataset: xCCA yielded the closest topology to the actual one as compared to other methods. A thicker edge indicates a stronger inter-camera correlation. M = missing edges, R = redundant edges.



(a) Actual (b) xCCA (c) xCA (d) Full Frame

Figure 3.14: Station B dataset: xCCA yielded the closest topology to the actual one as compared to other methods. A thicker edge indicates a stronger inter-camera correlation. M = missing edges, R = redundant edges.

in comparison to the xCA that tended to 'over-correlate', the proposed method failed to infer the connection between Cam 7 and Cam 8 because the area in Cam 8 adjacent to Cam 7 is too far away from the camera (at the end of the concourse). In addition, there are four entry/exit points in the field of view of Cam 7 leading to spaces not covered by Cam 8 (see Figure 3.4). This weakened the correlation between these two camera views and explains why the edge was miss-detected. Similarly, the proposed method failed to infer the connection between Cam 3 and 4 in Station B data as the connection point is too far away from the field of view as well as the existence of multiple entry/exit points. All methods inferred additional edges for camera pairs 1-3 and 4-6 in Station A dataset. Again this is not unexpected. Specifically, although they are not directly adjacent to each other (*e.g.* as shown in Figure 3.4, Cam 1 is adjacent to Cam 2, which is then next to Cam 3), they cover the same platforms therefore sharing a number of common activities, which are highly correlated, *e.g.* the arrival/departure of trains, passenger getting on/off trains.

To demonstrate the importance of activity-based scene decomposition on topology inference, xCCA was also carried out without scene decomposition, *i.e.* the activities within each camera

view as a whole are correlated with those in other camera views to infer the camera topology. The results are shown in Figure 3.13(d) and Figure 3.14(d), which suggest that without scene decomposition, even the proposed xCCA would not be able to learned the correct camera topology.

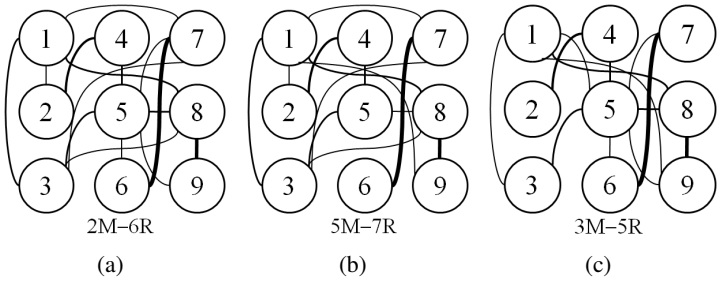


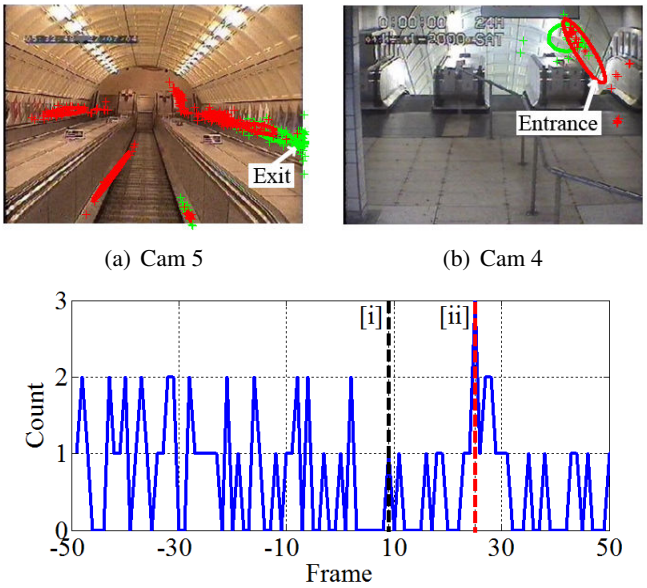
Figure 3.15: Camera topology inferred on station B dataset (a) without robust background subtraction + using correlation alone, (b) without robust background subtraction + using both correlation and time delay and (c) with robust background subtraction + using correlation alone.

Comparison with topology inference method without robust background subtraction and based solely on correlation strength – Experiments were conducted on Station B dataset to compare the topology inferred using the proposed method with those generated using (1) naive background subtraction method based on frame differencing and (2) region connectivity matrix $\mathbf{\Psi}$ characterised by correlation strength alone. The results are shown in Figure 3.15.

As can be seen from Figure 3.15(a), poor topology was inferred with naive background subtraction method and based only on correlation strength. In contrast, as shown in Figure 3.14(b), camera topology inferred using the proposed method produces fewer missing and redundant edges.

Even if both the correlation and time delay were exploited, topology inferred with the naive background subtraction method still consists of a number of missing and redundant edges (Figure 3.15(b)). On the other hand, if one employed the robust background subtraction method, a poor topology was still obtained when the topology was inferred based solely on correlation strength (Figure 3.15(c)). These results demonstrate that robust background modelling as well as the use of both correlation strength and time delay play important roles in making the proposed method scalable to challenging and complicated multi-camera scenes such as the one in the Station B dataset.

Comparison with tracking-based method – To highlight the inadequacy of tracking-based topology inference approach, a comparative experiment was carried out between the proposed approach with a method proposed by Makris *et al.* [149]. In particular, given a two hours video



(a) Cam 5 (b) Cam 4

(c) Exit/entry transition time distribution

Figure 3.16: (a) Passengers leave the field of view of Cam 5 from a zone marked with 'Exit' and (b) enter Cam 4 from a zone marked with 'Entrance'. (c) The exit/entry transition time distribution for selected pairs of zones obtained using tracking-based method proposed by [149]. Dotted lines labelled as [i] at 9 frames and [ii] at 25 frames represent the time delays between the selected pairs of zones estimated using the proposed method and the tracking-based method respectively. The average time delay obtained from manual observations is 9.12 frames.

clip, tracking was performed on Cam 4 and Cam 5 of Station A dataset using a multi-object tracker [41]. The starting and ending points of individual trajectories were clustered using the Gaussian Mixture Model (GMM) to automatically locate the entry and exit zones, as shown in Figure 3.16(a-b). Two entry and exit zones that correspond to the upward escalator and exit were selected and the corresponding exit/entry transition time distribution was plotted in Figure 3.16(c). Due to the low-frame rate problem, the detection of entry and exit points were intrinsically difficult. Therefore, not many correspondences can be established within the time window $-50:50$ as can be observed in Figure 3.16(c). Importantly, no clear peak can be seen from the exit/entry transition time distribution - only a maximal point at 25 frames was found, which may suggests the existence of an inter-zone connection. To verify this result, by using a small portion of the video clip, the amount of time taken by 50 objects passing across Cam 4 and Cam 5 was manually recorded. It is observed that on average, an object took 9.12 frames to pass between the two zones. This demonstrates that the tracking-based method failed to estimate the correct transition time. The failure of the tracking-based method is mainly due to the difficulty in performing object tracking in low-frame rate video featuring heavy occlusion. The resultant fragmentation of object trajectories produced unreliable trajectory starting points

and ending points, leading to inaccurate estimation of the entry/exit zones and transition time distribution. In comparison, using the proposed method, region 46 and 51 were automatically segmented corresponding to the entry and exit zones. The time delay between the two regions was estimated using xCCA as 9 frames, which was very close to the manual observation.

3.3.6 Context-aware Person Re-identification

In this experiment, the performance of matching people across camera views using colour histogram (CH) alone, CH+xCA, and CH+xCCA were compared. The probe set consists of 250 individuals, which were matched against a gallery set of 1800 people extracted from Station A dataset. The image of each person was manually segmented and normalised to 48×128 pixels. Each image was then divided into $N_{\mathrm{h}} = 8$ horizontal strips equally, from which the concatenated colour histograms were extracted. To select the best setting for CH, the number of bins n_{bin} was varied from $\{8, 16, 32, 64, 128, 256\}$ and both RGB and YUV colour spaces were attempted. It turned out that RGB colour space with 256 bins yielded the best result. Score returned by CH was computed as $\overline{S}^{a,b}_{\mathrm{bha}}$ (see Section 3.2.2), whilst score returned by CH + other methods were computed by Equation (3.22). The factor α that defines the size of the search window was set to 10.

Given a probe image, the matching scores over all 1800 people were computed and ranked from the most likely match to least likely one. To examine the recognition rate at different ranks, a Cumulative Matching Characteristic (CMC) curve [89] with a cut-off rank of 30 was plotted (Figure 3.18). Example matches are given in Figure 3.17. It can be seen that despite the poor image quality and drastic feature variations across camera views, results better than CH were obtained using both CH+xCCA and CH+xCA, with CH+xCCA yielding better result. In comparison, the result of using CH, *i.e.* visual appearance alone, was significantly worse. In particular, CH+xCCA yielded the best performance with approximately **94.00**% of the queries generated a true match in the top 20 rank, compared to 88.40% and 41.60% using CH+xCA and CH alone.

Without considering the activity correlation and time delay factor (CH alone), each person has to be compared against all possible candidates. However, as shown in Figure 3.19, passengers in the underground stations tend to wear clothes with similar colours (*e.g.* white shirt with black trousers). It is thus difficult to match the same person over a large camera network by considering the colour information or any visual appearance information alone. On the contrary, with

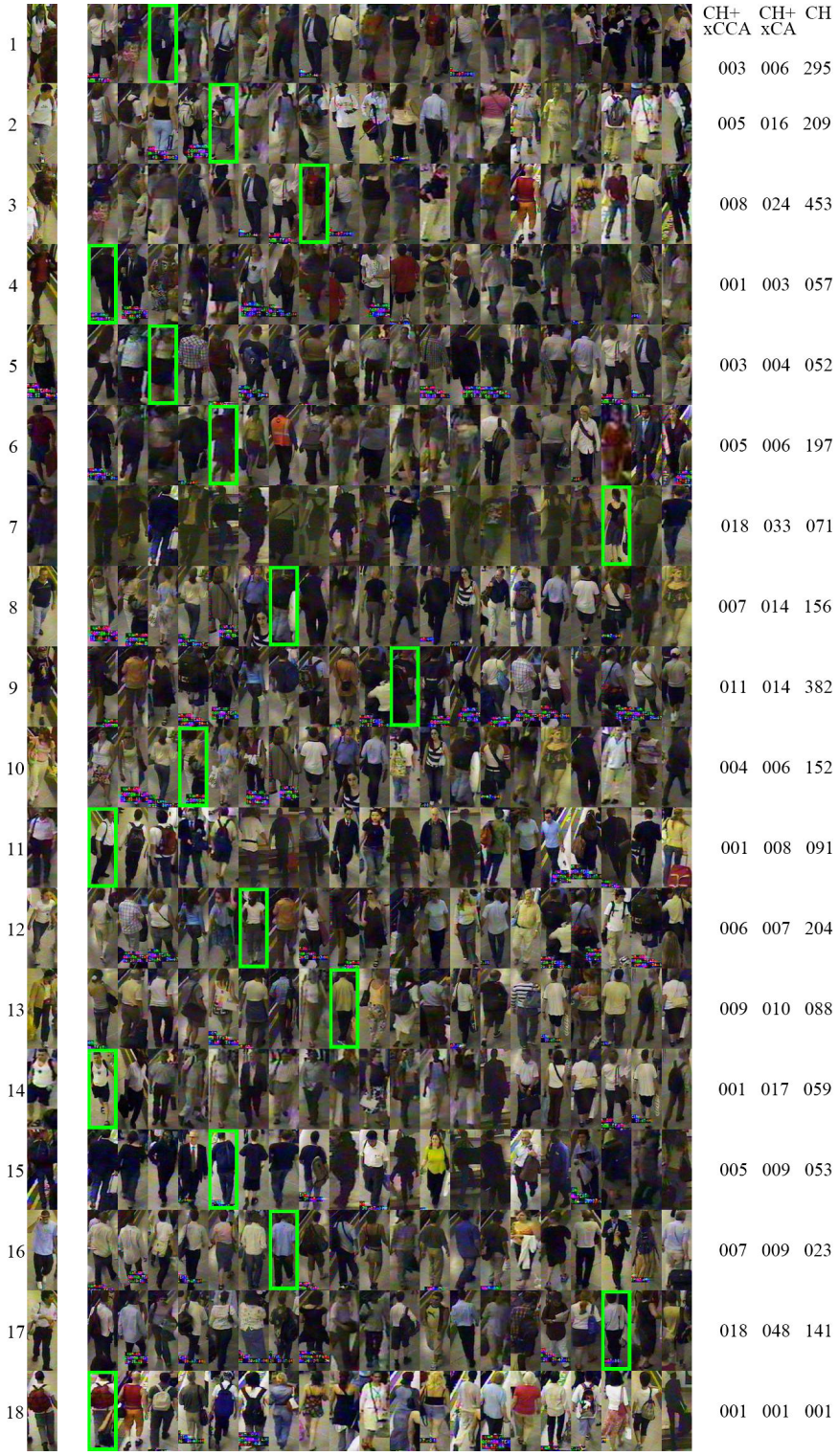


Figure 3.17: Example queries selected from the person re-identification experiment. The first image in each row is a probe image. It is followed by top 20 results, sorted from left to right according to the ranking obtained using CH+xCCA, with the correct match highlighted using a green bounding box. The ranks returned by the evaluated methods are included at the rightmost columns for comparison. Note the visual ambiguity in the search space due to variations of pose, colours, lighting changes; as well as poor image quality caused by low spatial resolution.

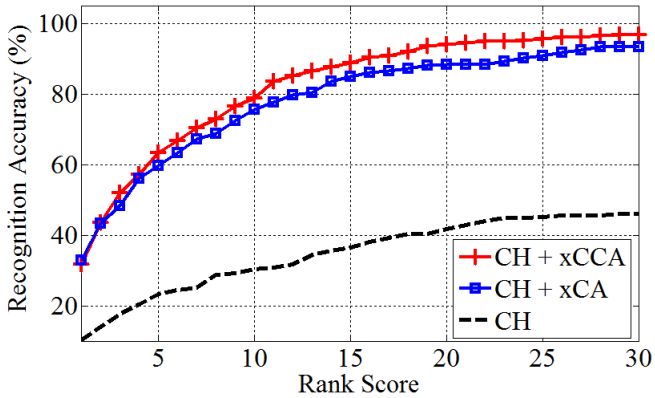


Figure 3.18: Cumulative Matching Characteristic (CMC) curve for CH+xCCA, CH+xCA and CH.

Figure 3.19: Comparing person re-identification result obtained using CH+xCCA and CH alone. Given the probe image at the leftmost column, CH+xCCA found the same person in another camera view at rank 1, whilst CH can only find the true match at rank 59. Ambiguities due to similar visual features presented by multiple objects are greatly reduced by introducing time delayed activity correlation as contextual information.

the inferred time delayed activity correlations employed as contextual information (CH+xCCA or CH+xCA), the search space and ambiguities were greatly reduced, which has resulted in significantly better recognition rate. Note that one can also employ the time delays estimated using tracking-based methods [149, 217] as contextual information to reduce the search space. However, given low-frame rate videos and crowded scenes, the estimation becomes inaccurate due to unreliable tracking (see Figure 3.16(c)). Incorporating the time delays estimated thus will harm instead of improving the person re-identification performance.

3.3.7 Activity-based Temporal Segmentation

Highly-correlated activities across camera views were discovered by performing spectral clustering on the regional activity affinity matrix **P**. For each set of highly-correlated activity patterns, 5000 frames were employed to train an HMM following the steps described in Section 3.2.3. The test set that consists of the rest of the videos was used to evaluate the performance of a model in temporal segmentation. The segmentation result obtained using the proposed multi-view activity analysis was compared with those from (1) individual single camera view without activity-based

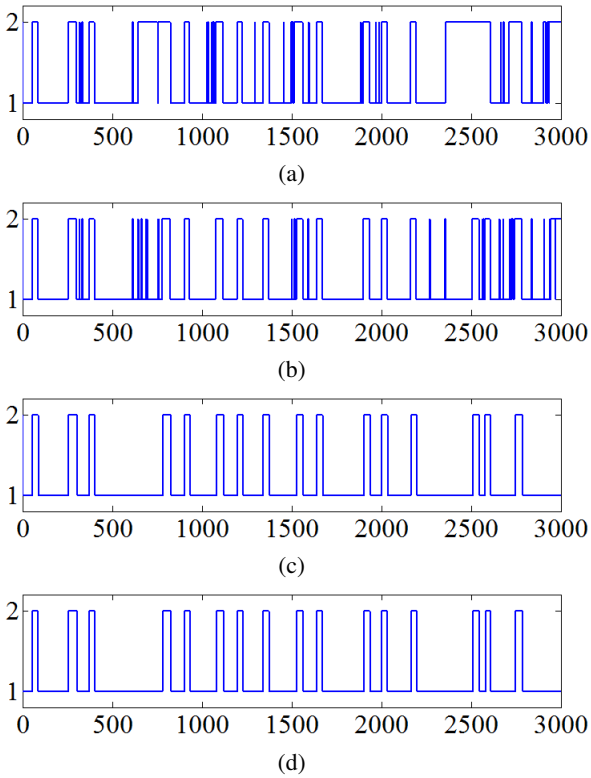


Figure 3.20: Station A dataset: example of phases inferred using (a) single view activity analysis without activity-based scene decomposition, (b) single view activity analysis with activity-based scene decomposition, and (c) multi-view activity analysis. The ground truth is shown in (d). Y-axis represents the inferred phases and X-axis represents the frame index. Only 3000 frames from the test set are shown.

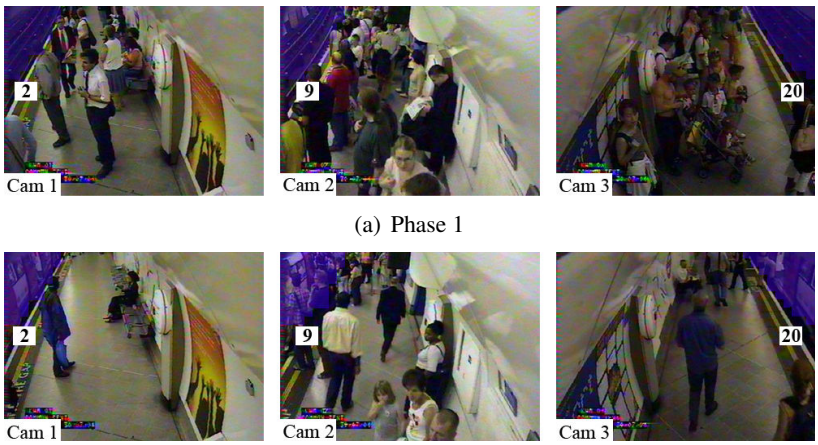


Figure 3.21: Station A dataset: example frames from the phases inferred using the proposed multi-view activity analysis. Phase 1: train is absent and passengers are waiting for train on the platform. Phase 2: train arrives and passengers get on/off the train.

scene decomposition and (2) single camera view with activity-based scene decomposition.

For Station A dataset, two sets of highly-correlated activity patterns were learned by clustering **P** (see Figure 3.11(a)), corresponding to the platform activities observed by Cam 1, 2, 3 and

Cam 4, 5, 6 respectively. For Cam 1, 2, 3, it turned out that an HMM with two hidden states gave the best BIC score in the model selection process. The two phases have clear semantic meaning: phase one corresponds to the period when a train is absent, whilst phase two is the period when a train is present. The phases inferred using the three methods were compared against the ground truth. The accuracy yielded by single view analysis (Cam 3) without scene decomposition was 73.40%. The accuracy increased to 83.78% after scene decomposition was employed on the single view analysis, whilst the proposed method based on multi-view activity analysis gave **97.90**%. Examples of the inferred phases by different methods and some example frames from the segmented phases are shown in Figure 3.20 and Figure 3.21 respectively. Similar results were obtained on Cam 4, 5, 6 – the accuracies were 86.59%, 91.70% and **94.09**% for methods without scene decomposition, with scene decomposition, and scene decomposition + single view analysis.

The same procedures were repeated on Station B dataset. Several sets of highly-correlated activity patterns were discovered, which include the platform activities monitored by Cam 6, 7 and Cam 8, 9, as well as escalator activities captured by Cam 4, 5. The activity set reported here was that which occurred at the escalator area, from which regions 46 and 51 were automatically detected as highly-correlated regions. A two-state HMM was selected in the automatic model selection process. The two phases contain clear semantic meaning: phase one occurs when passengers on the escalator track approach the escalator exit, whilst phase two takes place when passengers move clear of the escalator exit area. In general, the results achieved on Station B dataset were relatively poorer compared to those obtained on Station A dataset as the occlusion problem was more severe. In particular, single view analysis (Cam 4) without and with scene decomposition yielded similar results on this dataset, giving accuracies of 67.32% and 67.10% respectively. Scene decomposition failed to improve the result on single view analysis because region 46 in Cam 4 (see Figure 3.8(d)) only occupies a small portion of many regions in the whole view. The activity model was thus dominated by activity patterns learned from other regions. As opposed to the platform scene in Station A dataset, there were multiple co-existing activities in the scene captured by Cam 4 and 5, namely passengers travelling upwards and downwards on different escalator tracks. Using all regions blindly thus would not help in improving the video segmentation accuracy. Overall, the proposed method based on multi-view activity analysis gave the best accuracy of **79.08**%. Examples of the inferred phases by different methods and

some example frames from the segmented phases are shown in Figure 3.22 and Figure 3.23 respectively.

The results on both datasets demonstrate the effectiveness of the proposed multi-view activity modelling method, which is based on the learning of regional activity correlations. In particular, single view activity analysis was susceptible to noise and visual ambiguities due to heavy occlusions and low frame rate. As compared to single view activity analysis, the proposed method utilises evidence collected from multiple correlated regions across different camera views. It therefore reduced visual ambiguities, resulting in a more accurate segmentation result.

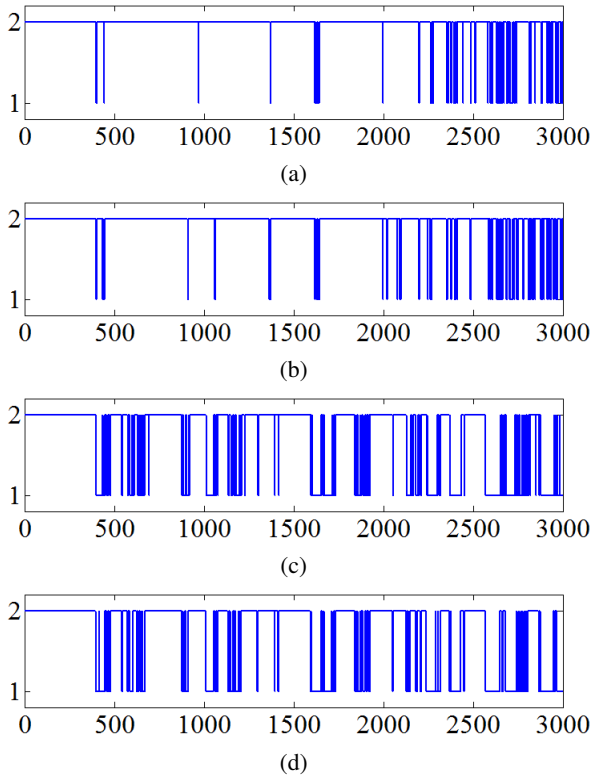


Figure 3.22: Station B dataset: example of phases inferred using (a) single view activity analysis without activity-based scene decomposition, (b) single view activity analysis with activity-based scene decomposition, and (c) multi-view activity analysis. The ground truth is shown in (d). Y-axis represents the inferred phases and X-axis represents the frame index. Only 3000 frames from the test set are shown.

3.4 Summary

This chapter has presented a new approach to multi-camera activity understanding by discovering and modelling the pairwise correlations with unknown time delays between activities observed within and across non-overlapping camera views. In particular, a novel Cross Canonical Correlation Analysis (xCCA) is introduced to detect and quantify correlation and temporal relationships

(a) Phase 1 (b) Phase 2

Figure 3.23: Station B dataset: example frames from the phases inferred using multi-view activity analysis. Phase 1: passengers on the escalator track are approaching the escalator exit; Phase 2: passengers move clear of the escalator exit area.

between partial observations across local regions. Experimental results have shown that the time delayed activity correlations are not only useful for inferring the spatial and temporal topology of a camera network, but also important as contextual information to facilitate more robust and accurate person re-identification, and activity-based temporal segmentation. The proposed approach does not rely on either inter-camera or intra-camera tracking. Consequently, as demonstrated through experiments in this chapter, it can be applied to challenging surveillance videos, which featured heavy occlusions due to enormous number of objects in the scenes, as well as poor image quality caused by low video frame rate and image resolution.

However, the discovered activity correlations are limited to pairwise correlations and multiple dependencies in a global context are not considered. Hence, the current framework is not ready for global unusual event detection in a camera network, where multiple complex dependencies among activity patterns are observed from different camera views. In addition, the HMM proposed in Section 3.2.3 is only tractable to handle limited number of camera views. Generalising the model for unusual event detection task in a large camera network would inevitably entails large numbers of parameters. The next chapter addresses these problems by formulating a new probabilistic graphical model whose time delayed activity dependencies are modelled and optimised globally with a two-stage structure learning method. The new method is computationally tractable therefore can be applied to large numbers of camera views.

Chapter 4

Discovering Global Activity Dependency

As discussed in Section 1.2.1, detecting global unusual events in multiple disjoint cameras is nontrivial due to visual discontinuity owing to the arbitrary and unknown time gaps between cameras. In particular, a global unusual event may look perfectly normal when it is examined in isolation from different camera views. Reasoning based solely on visual evidence collected within an isolated FOV is likely to miss such an unusual event.

The preceding chapter has described a new approach based on Cross Canonical Correlation Analysis (xCCA) for global activity understanding in multiple non-overlapping camera views. In particular, it has demonstrated that time delayed correlations between regional activity patterns can serve as a useful contextual cue to facilitate camera topology inference, more robust person re-identification and more accurate video temporal segmentation.

Whilst the xCCA-based method has shown superior performance in the aforementioned tasks, it is not suitable for global unusual event detection since it can only discover and quantify pairwise correlation among regional activity patterns without considering multiple dependencies in a global context. In order to build a graphical model for global reasoning, a different formulation is required to discover key dependencies among different regions across multiple views.

In this chapter, a new approach is proposed for global unusual event detection in multiple disjoint cameras. In particular, multi-camera activities are modelled using a Time Delayed Probabilistic Graphical Model (TD-PGM) with different nodes representing activities observed in different decomposed regions from different views, and the directed links between nodes encoding the time delayed dependencies between the activities. The time delayed dependencies modelling

problem is formulated as a graph structure learning (*i.e.* to discover conditional dependency links between a set of nodes) and parameter learning task (*i.e.* to learn the parameter associated with the links). In particular, a new two-stage structure learning is formulated to discover and quantify the globally optimised time delayed activity dependencies among regional activity patterns observed across views. This differs from the approach proposed in the preceding chapter, which only performs pairwise correlation learning.

Once learned, the TD-PGM can be used for real-time unusual event detection and localisation of activity regions that jointly contribute a global unusual event. Specifically, a global-level unusual event can be detected when the learned normal time delayed dependencies are not supported by visual evidence collected across camera views on the fly. However, since this work is interested in analysing busy public scenes featured with severe occlusions and low image resolution both spatially and temporally, the detection could potentially be sensitive to noise resulting in a large number of false alarms. To overcome this problem, a new Cumulative Abnormality Score (CAS) is introduced to replace the conventional log-likelihood (LL) score for more robust and reliable real-time unusual event detection.

The remainder of this chapter is organised as follows: Section 4.1 presents the proposed TD-PGM and two-stage structure learning. This is followed by explanation of the proposed global unusual event detection approach in Section 4.2. Experimental results are reported in Section 4.3. Finally, a summary is given in Section 4.4.

4.1 Global Activity Dependency Modelling

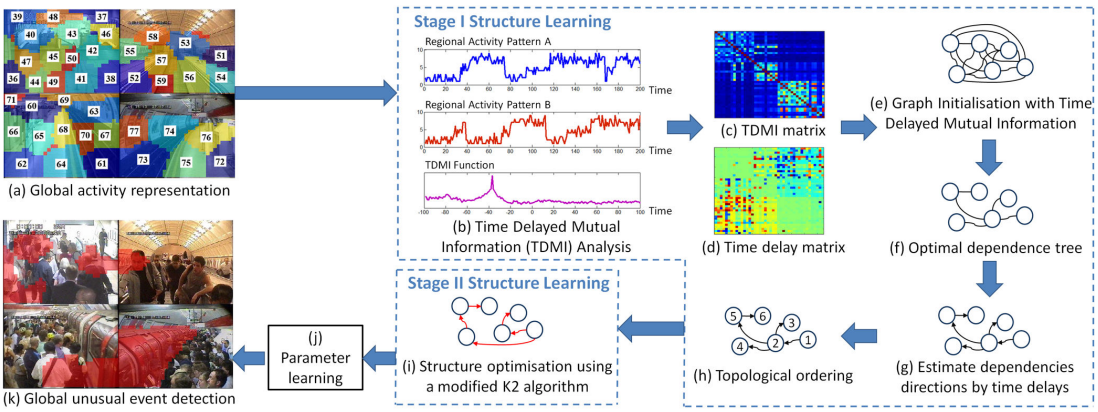


Figure 4.1: A diagram illustrating the proposed approach for modelling global time delayed dependencies between activities observed in multiple disjoint cameras.

An overview of the key steps of the proposed approach is depicted in Figure 4.1. Specifically,

given disjoint cameras with non-overlapping views, the proposed approach first decomposes the scene into regions (Figure 4.1(a)). Local activities in each region are then represented as time-series and used as input to the proposed two-stage structure learning method for learning time delayed dependencies. In the first stage, time delays between activities of each region are discovered using Time Delayed Mutual Information (TDMI) analysis (see Section 4.1.3 for detailed description). A prior structure of the graphical model is then learned using a constraint-based method based on the time delay constraint (Figure 4.1(b-h)). This is followed in the second stage (Figure 4.1(i)) by a scored-searching based structure learning method derived by re-formulating the K2 algorithm [51]. With both parameter and structure updated, the model can then be used for real-time unusual event detection (Figure 4.1(j-k)).

4.1.1 Global Activity Representation

To facilitate global activity understanding across disjoint cameras, it is useful to decompose each camera view into regions (Figure 4.1(a)) where different activity patterns are observed (*e.g.* decompose a traffic junction into different lanes and waiting zones). To this end, the scene decomposition approach introduced in Chapter 3 is adopted to segment a scene using spectral clustering [250] based on correlation distances of local block spatiotemporal activity patterns. This results in n regions across all views, which are indexed in a common reference space.

Given the decomposed scene, activity patterns observed over time in the ith region is represented as a bivariate time series $(\hat{\mathbf{u}}_i, \hat{\mathbf{v}}_i)$ similar to Equation (3.8). Specifically, $\hat{\mathbf{u}}_i$ represents the percentage of static foreground pixels within the ith region, whilst $\hat{\mathbf{v}}_i$ is the percentage of pixels within the region that are classified as moving foreground.

To obtain a more compact representation for learning a TD-PGM, the original time-series in two-dimensional feature space $(\hat{\mathbf{u}}_i, \hat{\mathbf{v}}_i)$ for each region are clustered using a Gaussian Mixture Model (GMM). The GMM is learned using Expectation-Maximisation (EM) with the number of components K_i determined by automatic model order selection using the Bayesian Information Criterion (BIC). The learned GMM is then used to classify activity patterns detected in each region at each frame into one of the K_i components. Activity patterns in the ith region over time are thus represented using the class labels and denoted as a one-dimensional time-series:

$$\mathrm{x}_i(t) = (x_{i,1}, \ldots, x_{i,t}, \ldots), \tag{4.1}$$

where $x_{i,t} \in \{1,2,\ldots,K_i\}$ and $i=1,\ldots,n$.

4.1.2 Time Delayed Probabilistic Graphical Model

Time delayed dependencies among regional activity patterns are modelled using a TD-PGM. This is achieved by taking two steps: (1) two-stage structure learning (Figure 4.1(b-i)) and (2) parameter learning (Figure 4.1(j)). Let us first formally define the model. A TD-PGM is defined as $B = \langle G, \Theta \rangle$, which consists of a directed acyclic graph (DAG)[1], G whose nodes represent a set of discrete random variables $\mathbf{X} = \{X_i | i = 1,2,\ldots,n\}$, where X_i is the ith variable representing activity patterns observed in the ith region. Each specific value taken by a variable X_i is denoted as x_i. A stream of values x_i of variable X_i is denoted as $\mathrm{x}_i(t) = (x_{i,1},\ldots,x_{i,t},\ldots)$ (see Equation (4.1)).

The model is quantified by a set of parameters denoted by Θ specifying the conditional probability distribution (CPD), $p(X_i|\mathbf{Pa}(X_i))$. Since all the observations in the model are discrete variables due to the GMM clustering, the CPD between a child node X_i and its parents $\mathbf{Pa}(X_i)$ in G is represented using a multinomial probability distribution. Consequently, Θ contains a set of parameters $\boldsymbol{\theta}_{x_i|\mathbf{pa}(X_i)} = p(x_i|\mathbf{pa}(X_i))$ for each possible discrete value x_i of X_i and $\mathbf{pa}(X_i)$ of $\mathbf{Pa}(X_i)$. Here $\mathbf{Pa}(X_i)$ represents the set of parents of X_i, and $\mathbf{pa}(X_i)$ is an instantiation of $\mathbf{Pa}(X_i)$.

Conditional independence is assumed which implies that X_i is independent from its non-descendants given its parents. These relationships are represented through a set of directed edges $\mathbf{E}$, each of which points to a node from its parents on which the distribution is conditioned. Given any two variables X_i and X_j, a directed edge from X_i to X_j is represented by denoting $X_i \rightarrow X_j$, where $(X_i, X_j) \in \mathbf{E}$ and $(X_j, X_i) \notin \mathbf{E}$. Note that the $p(X_i|\mathbf{Pa}(X_i))$ are not quantified using a common time index but with relative time delays that are discovered using Time Delayed Mutual Information (TDMI) discussed in the next section.

Other notations used in this Chapter are given as follows: the number of states of X_i is r_i, and the number of possible configurations of $\mathbf{Pa}(X_i)$ is q_i. A set of discrete value x_i across all variables is given as $\mathbf{x} = \{x_i | i = 1,2,\ldots,n\}$. Consequently, a collection of m cases of $\mathbf{x}$ is denoted as $\mathcal{X} = \{\mathbf{x}_1,\ldots,\mathbf{x}_m\}$. The number of cases of $(x_i, \mathbf{pa}(X_i))$ in $\mathcal{X}$ is represented as $N_{x_i|\mathbf{pa}(X_i)}$, specifically $N_{ijk} = N_{x_i=k|\mathbf{pa}(X_i)=j}$.

[1]The TD-PGM assumes acyclicity and is limited to model unidirectional dependency. Cyclic dependency modelling is desirable but it requires more elaborated model, which can be intractable given a large camera network. More discussion can be found in Section 7.1.

4.1.3 Two-Stage Structure Learning

By representing the regional activities as variables in the TD-PGM, the proposed approach wishes to discover and quantify their optimal time delayed dependencies through learning the structure of TD-PGM. In particular, the first stage of the proposed two-stage structure learning approach aims to obtain a prior graph structure, which can be further used to derive an ordering constraint. The constraint is propagated to the second-stage learning to *reduce and constrain the structure search space*, by eliminating any candidate structure inconsistent with the constraint. This consequently leads to a significant computational speed up in the searching process and accuracy in second-stage structure learning. Let us now discuss the steps involved in the first-stage learning:

Constraint-based learning with Time Delayed Mutual Information Analysis

The first stage of the proposed structure learning approach consists of three key steps: *Step 1* - Time Delayed Mutual Information (TDMI) analysis for learning time delays and establishing initial associations between nodes (Figure 4.1(b-e)); *Step 2* - Generation of an optimal dependence tree using Chow-Liu algorithm [47] (Figure 4.1(f)); *Step 3* - Orientation of edges in the optimal dependence tree (Figure 4.1(g)).

1. *Step 1 - TDMI analysis* - Time Delayed Mutual Information [70] analysis is explored here to learn the initial time delayed association between each pair of regional activity patterns. The TDMI was first introduced by Fraser and Swinney [70] for determining the delay parameter in chaotic dynamical system, through measuring the Mutual Information (MI) between a time series $\mathrm{x}(t)$ and a time shifted copy of itself $\mathrm{x}(t+\tau)$ as a function of time delay τ. The main rationale behind the use of TDMI is that if two regional activity patterns are dependent, information conveyed by a region provides a large amount of information on another region. Note that both the xCCA presented in Chapter 3 and the TDMI are capable of measuring the pairwise time delayed relationships between regional activity patterns. Nevertheless, the TDMI appears to be a more appropriate option in this framework for discovering optimised activity dependencies globally. The reason is that the TDMI is theoretically motivated for finding an optimal dependence tree in Step-2 of the two-stage structure learning. In particular, as proven by Chow and Liu [47], maximising the total MI of a tree structure produces the optimum first-order dependence tree (see Appendix A for details). In contrast, using the xCCA does not necessarily return a tree structure that best approximates a target distribution. The advantage of using the TDMI

can be observed from its better performance compared to that of the xCCA in an unusual event detection experiment (see Section 4.3.3).

In TDMI analysis, if one treats two arbitrary regional activity patterns as time series data and denotes them as $\mathrm{x}(t)$ and $\mathrm{y}(t)$ respectively, the TDMI of $\mathrm{x}(t)$ and time shifted $\mathrm{y}(t+\tau)$ can be written as follows:

$$\mathrm{I}\left(\mathrm{x}(t);\ \mathrm{y}(t+\tau)\right) = \sum_{j=1}^{K_{\mathrm{x}}} \sum_{k=1}^{K_{\mathrm{y}}} p_{\mathrm{xy}}\left(j,k\right) \log_2 \frac{p_{\mathrm{xy}}\left(j,k\right)}{p_{\mathrm{x}}\left(j\right) p_{\mathrm{y}}\left(k\right)}, \tag{4.2}$$

where $p_{\mathrm{x}}\left(\cdot\right)$ and $p_{\mathrm{y}}\left(\cdot\right)$ denote the marginal probability distribution functions of $\mathrm{x}(t)$ and $\mathrm{y}(t+\tau)$ respectively, whilst $p_{\mathrm{xy}}\left(\cdot\right)$ is the joint probability distribution function of $\mathrm{x}(t)$ and $\mathrm{y}(t+\tau)$. The probability distribution functions are approximated by constructing histograms with K_i equal-width bins, each of which corresponds to one GMM class discovered using approach described in Section 4.1.1. Consequently, $p_{\mathrm{x}}\left(j\right)$ represents the probability that the time series $\mathrm{x}(t)$ takes a value inside bin j of the histogram, whilst $p_{\mathrm{xy}}\left(j,k\right)$ is the probability that $\mathrm{x}(t)$ is in bin j and $\mathrm{y}(t+\tau)$ is in bin k. Note that $\mathrm{I}\left(\mathrm{x}(t);\ \mathrm{y}(t+\tau)\right) \geq 0$ with the equality if, and only if $\mathrm{x}(t)$ and $\mathrm{y}(t+\tau)$ are independent. If $\tau = 0$, TDMI is equivalent to MI of $\mathrm{x}(t)$ and $\mathrm{y}(t)$.

Subsequently, a TDMI function $\mathcal{I}_{\mathrm{xy}}\left(\tau\right)$ is obtained as a sequence of TDMI values $\mathrm{I}\left(\mathrm{x}(t);\ \mathrm{y}(t+\tau)\right)$ at different time delay τ:

$$\mathcal{I}_{\mathrm{xy}}\left(\tau\right) = \left(\mathrm{I}\left(\mathrm{x}(t);\ \mathrm{y}(t-T)\right), \ldots, \mathrm{I}\left(\mathrm{x}(t);\ \mathrm{y}(t+T)\right)\right), \tag{4.3}$$

where $-T \leq \tau \leq T$.

In general, given a TDMI function $\mathcal{I}_{ij}\left(\tau\right)$, one can estimate the time delay $\hat{\tau}_{ij}$ between ith and jth regions as:

$$\hat{\tau}_{ij} = \underset{\tau}{\operatorname{argmax}}\ \mathcal{I}_{ij}\left(\tau\right). \tag{4.4}$$

By repeating the same process for local activities observed in each pair of decomposed regions, one can construct a time delay matrix $\mathbf{D}$ as follows:

$$\mathbf{D} = \left[\hat{\tau}_{ij}\right]_{n \times n}. \tag{4.5}$$

The corresponding TDMI matrix is obtained as:

$$\hat{\mathrm{I}}_{ij} = \mathcal{I}_{ij}\left(\hat{\tau}_{ij}\right) \tag{4.6}$$

$$\mathbf{I} = [\hat{\mathrm{I}}_{ij}]_{n\times n}. \tag{4.7}$$

2. *Step 2 - Generating an optimal dependence tree* - In this step, the proposed approach finds an optimal dependence tree (Chow-Liu tree [47]) $\mathcal{T}$ that best approximates the graph joint probability $p(\mathbf{X})$ by a product of second-order conditional and marginal distributions (see Appendix A for details), given as

$$p(\mathbf{X}) = p(X_1,\ldots,X_n) = \prod_{i=1}^{n} p\left(X_i|X_{\psi(i)}\right), \tag{4.8}$$

where mapping function $\psi(i)$ with $0 \leq \psi(i) < n$ defines $\mathcal{T}$ so that $X_{\psi(i)} = \mathbf{Pa}(X_i)$ if $\psi(i) > 0$ and $X_{\psi(i)} = \mathbf{Pa}(X_i) = \emptyset$ if $\psi(i) = 0$.

The optimal dependence tree $\mathcal{T}$ can be obtained based on the TDMI matrix $\mathbf{I}$ found in the TDMI analysis. In particular, weights are assigned following $\mathbf{I}$ to each possible edges of a weighted graph with node set $\mathbf{X}$ that encodes no assertion of conditional independence. Prim's algorithm [183] is then applied to find a subset of the edges that forms a tree structure including every node, in which the total weight is maximised, *i.e.* to find a maximum-weight dependence tree $\mathcal{T}$ such that

$$\sum_{i=1}^{n} \hat{\mathrm{I}}_{i,\psi(i)} \geq \sum_{i=1}^{n} \hat{\mathrm{I}}_{i,\psi'(i)}. \tag{4.9}$$

for all $\mathcal{T}' \in \boldsymbol{\mathcal{T}}$, where $\boldsymbol{\mathcal{T}}$ represents the set of all possible first-order dependence trees.

3. *Step 3 - Edge orientation* - In step 3, the undirected tree $\mathcal{T}$ is transformed to a directed prior graph structure G^{p} by assigning orientations to the edges. Typically, one can assign edge orientations by either selecting a random node as a root node, or by performing a conditional independence test [42] and a scoring function optimisation over the graph [43]. These methods are either inaccurate or require exhaustive search on all possible edge orientations and therefore computationally costly.

To overcome these problems, this chapter proposes to orient the edges by tracing the time delays for a pair of nodes in the tree structure using $\mathbf{D}$ learned by the TDMI analysis (Equation (4.5)). In particular, if the activity patterns observed in X_i are lagging the patterns

observed in X_j with a time delay τ, it is reasonable to assume that the distribution of X_i is conditionally dependent on X_j. The edge is therefore directed from X_j to X_i. With G^{p} defined by the edges, one can derive the ordering of variables $\prec$ by performing topological sorting [52]. In particular, the ordering $\prec$ specifies that a variable X_j can only be the parent of X_i if, and only if, X_j precedes X_i in $\prec$, *i.e.* $X_j \in \mathbf{Pa}(X_i)$ iff $X_j \prec X_i$. The whole procedure of obtaining G^{p} and $\prec$ is summarised in Algorithm 1.

Algorithm 1: Finding a prior graph structure and a topological ordering.

Input: An undirected weighted graph with a node set $\mathbf{X} = \{X_i | i = 1, 2, \ldots, n\}$, and edge set $\mathbf{E}$. TDMI matrix $\mathbf{I}$ and time delay matrix $\mathbf{D}$.

Output: Prior graph structure G^{p} defined by $\mathbf{X}^{\mathrm{p}}$ and $\mathbf{E}^{\mathrm{p}}$. Topological ordering $\prec$.

```
X^p = X, where X is an arbitrary node from X;
E^p = ∅;
while X^p ≠ X do
    Choose an edge (X_i, X_j) ∈ E with maximum weight Î_ij, where X_i ∈ X^p and X_j ∉ X^p;
    X^p = X^p ∪ {X_j};
    E^p = E^p ∪ {(X_i, X_j)};
    if τ̂_ij > 0 then
        X_i → X_j;
    else
        X_i ← X_j;
    end
end
≺ = topological_sort (E^p);
```

Time Delayed Scored-Searching based Learning

In the second-stage of the proposed structure learning approach, a popular heuristic search method known as the K2 algorithm [51] is re-formulated to generate an optimised time delayed dependency structure based on $\prec$ derived from the first-stage learning. Note that without the first-stage learning, one may set $\prec$ randomly. However, a randomly set $\prec$ does not guarantee to give the most probable model structure. Alternatively, one can apply the K2 algorithm exhaustively on all possible orderings to find a structure that maximises the score. However, this solution is clearly

infeasible even for a moderate number of nodes, since the space of ordering is $n!$ for a n-node graph.

Let us now describe the details of the second-stage learning (Algorithm 2). The K2 algorithm iterates over each node X_i that has an empty parent set, $\mathbf{Pa}(X_i)$ initially. Candidate parents are then selected in accordance with the node sequence specified by $\prec$ and they are added incrementally to $\mathbf{Pa}(X_n)$ whose addition increases the score $S(G|\mathcal{X})$ of the structure G given dataset $\mathcal{X}$.

In this study, two widely used scoring functions are considered, which are both score equivalent and decomposable [97], namely the Bayesian Dirichlet equivalent uniform (BDeu) score [34] and the Bayesian Information Criterion (BIC) score [196]. Specifically, the BDeu score is defined as:

$$\begin{aligned} S_{\text{BDeu}}(G|\mathcal{X}) &= \sum_{i=1}^{n} S_{\text{BDeu}}\left(X_i|\mathbf{Pa}(X_i)\right) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{q_i}\left[\log\frac{\Gamma\left(\frac{\eta}{q_i}\right)}{\Gamma\left(N_{ij}+\frac{\eta}{q_i}\right)} + \sum_{k=1}^{r_i}\log\frac{\Gamma\left(N_{ijk}+\frac{\eta}{r_i q_i}\right)}{\Gamma\left(\frac{\eta}{r_i q_i}\right)}\right], \end{aligned} \tag{4.10}$$

where η is a parameter known as equivalent sample size [97], $\Gamma(\cdot)$ denotes the Gamma function, and $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$. The number of states of X_i is r_i and the number of possible configurations of $\mathbf{Pa}(X_i)$ is q_i. The BIC score is given as:

$$\begin{aligned} S_{\text{BIC}}(G|\mathcal{X}) &= \sum_{i=1}^{n} S_{\text{BIC}}\left(X_i|\mathbf{Pa}(X_i)\right) \\ &= \sum_{i=1}^{n}\sum_{t=1}^{m}\log p(x_{i,t}|\mathbf{pa}(X_i), \boldsymbol{\theta}_{x_i|\mathbf{pa}(X_i)}) - \log m \sum_{i=1}^{n}\frac{b_i}{2}, \end{aligned} \tag{4.11}$$

where $b_i = q_i(r_i - 1)$ is the number of parameters needed to describe $p(X_i|\mathbf{Pa}(X_i))$. Unlike the BDeu score, an explicit penalty term $\log m \sum_{i=1}^{n}\frac{b_i}{2}$ is included in the BIC score to penalise complexity of a model. The proposed formulation differs from the original K2 algorithm in that any addition of candidate parent is required not only to increase the graph structure score, but it must also satisfy the constraint imposed by the time delays discovered in the first-stage learning. In addition, the score computations (Equations (4.10),(4.11)) are carried out by shifting a parent's activity patterns with a relative delay to a child node's activity patterns based on $\mathbf{D}$.

Algorithm 2: The re-formulated K2 algorithm with a time delay factor being introduced.

Input: A graph with a node set $\mathbf{X} = \{X_i | i = 1, 2, \ldots, n\}$. An ordering of nodes $\prec$. An upper bound φ on the number the parents a node may have. Time delay matrix $\mathbf{D}$.

Output: Final structure G defined by $\{(X_i, \mathbf{Pa}(X_i)) \mid i = 1, 2, \ldots, n\}$.

```
for i = 1 to n do
    Pa(X_i) = ∅;
    score_old = S(X_i|Pa(X_i));
    OKToProceed = true;
    while OKToProceed and |Pa(X_i)| < φ do
        Let X_j ≺ X_i, X_j ∉ Pa(X_i), with activity patterns x_j(t+τ), τ = D(X_i,X_j) ≤ 0,
        which maximises S(X_i|Pa(X_i) ∪ {X_j});
        score_new = S(X_i|Pa(X_i) ∪ {X_j});
        if score_new > score_old then
            score_old = score_new;
            Pa(X_i) = Pa(X_i) ∪ {X_j};
        else
            OKToProceed = false;
        end
    end
end
```

Note that both the two stages of structure learning are important to discover and learn the time delayed dependencies among regional activities. Specifically, without the first-stage structure learning, vital time delay information would not be available for constraining the search space. On the other hand, as one shall see later in the experimental section (Section 4.3), poorer results may be obtained if one uses the tree structure alone without the second stage learning. This is because the tree structure can only approximate an optimum set of $n-1$ first-order dependence relationship among the n variables but not the target distribution, which may include more complex dependencies. Furthermore, studies have shown that the constraint-based learning can be sensitive to failures in independence tests [75]. Therefore, a second-stage scored-searching based learning is needed to discover additional dependencies and correct potential error in the first-stage learning.

A heuristic search algorithm is chosen for the proposed second-stage structure learning instead of an exact learning algorithm. In general, exact structure learning is intractable for large graphs, since there are $2^{O(n^2 \log n)}$ DAGs for a n-node graph [73]. A search using a typical exact algorithm would take exponential time on the number of variables n, *e.g.* $O(n2^n)$ for a dynamic programming-based technique [57]. Such a high complexity prohibits its use from learning any typical camera network, which may consist of hundreds of local activity regions depending on scene complexity.

Among various heuristic search algorithms, the K2 algorithm [51] is found to be well suited for learning the dependency structure of a large camera network due to its superior computational efficiency given the ordering constraint. Specifically, thanks to the ordering constraint, the search space of the K2 algorithm is much smaller than that of a conventional greedy hill-climbing (GHC) search [10]. In addition, the constraint also helps in avoiding the costly acyclicity checks since the topological order already ensures acyclicity of structure. Besides, the K2 algorithm is also more efficient than alternative methods such as Markov Chain Monte Carlo (MCMC) based structure learning [159], which requires a sufficiently long burn-in time to obtain a converged approximation for a large graph [84].

4.1.4 Parameter Learning

There are two typical methods for estimating the parameters of a probabilistic graphical model given fully observed data, namely Maximum Likelihood Estimation (MLE) and Bayesian learning. In the MLE methodology, one aims to find a set of parameters that maximise the likelihood function. In the Bayesian learning methodology, one begins with a prior probability over the model parameters and computes a posterior when new instances are observed. Then, this posterior can be treated as one's current belief and use it as a prior for the next learning iteration. This method has an attractive property that the belief state at time t is the same as the posterior after observing $\mathbf{x}_1, \ldots, \mathbf{x}_t$ from an initial prior belief state. In other words, one can update the posterior distribution sequentially and efficiently using a closed-form formula with a conjugate prior[2] is used. In this study, the latter approach is adopted since it offers an tractable solution for finding the optimal parameters during learning. Specifically, the BDeu prior (likelihood equivalent uniform Bayesian Dirichlet) [34] - a conjugate prior of multinomial distribution is applied over the model parameters. To account for a cross-region time delay factor, regional activity patterns are

[2]Conjugate prior is a prior that has the same functional form as the posterior distribution [21].

temporally shifted according to the time delay matrix **D** during the parameter learning stage.

4.1.5 Computational Cost Analysis

For the first-stage learning (see Algorithm 1), the total possible region pairs to be considered for obtaining pairwise TDMI function (Equation (4.3)) is in the order of $\mathrm{O}(n^2)$, where n is the number of nodes/regions. In each TDMI function computation, if one bound the maximum time delay to be $\tau_{\max}$, the total number of TDMI calculations (Equation (4.2)) is $\tau_{\max} - 1$. Hence, the overall complexity of TDMI analysis (Step-1) is $\mathrm{O}(n^2\tau_{\max})$. The run time complexity of the optimal dependence tree approximation (Step-2) (Section 4.1.3) is $\mathrm{O}(e \log n)$, and the topological sorting (Step-3) takes $\mathrm{O}(n+e)$ time [52], where e is the number of edges.

For the second-stage structure learning (see Algorithm 2), the **for** statement loops $\mathrm{O}(n)$ times. The **while** statement loops at most $\mathrm{O}(\varphi)$ times once it is entered, where φ denotes the maximum number of parents a node may have. Inside the **while** loop, line 6 in Algorithm 2 is executed for at most $n-1$ times since there are at most $n-1$ candidate parents consistent with $\prec$ for X_i. Hence, line 6 in Algorithm 2 takes $\mathrm{O}(sn)$ time if one assumes each score evaluation takes $\mathrm{O}(s)$ time. Other statements in the **while** loop takes $\mathrm{O}(1)$ time. Therefore, the overall complexity of the second-stage structure learning is $\mathrm{O}(sn)\,\mathrm{O}(\varphi)\,\mathrm{O}(n) = \mathrm{O}(sn^2\varphi)$. In the worst case scenario where one do not apply an upper bound to the number of parents a node may have, the time complexity becomes $\mathrm{O}(sn^3)$ since $\varphi = n$.

4.2 Global Unusual Event Detection

A conventional way for detecting unusual events is to examine the log-likelihood (LL), $\log p(\mathbf{x}_t|\Theta)$ of the observations given a model, *e.g.* [259]. Specifically, an unseen global activity pattern is detected as being unusual if

$$\log p(\mathbf{x}_t|\Theta) = \sum_{i=1}^{n} \log p(x_{i,t}|\mathbf{pa}(X_i), \boldsymbol{\theta}_{x_i|\mathbf{pa}(X_i)}) < \mathrm{Th}, \tag{4.12}$$

where Th is a pre-defined threshold, and $\mathbf{x}_t = \{x_{i,t}|i = 1,2,\ldots,n\}$ are observations at time slice t for all n regions. However, given a crowded public scene captured using videos with low image resolution both spatially and temporally, observations $\mathbf{x}_t$ inevitably contain noise and the LL-based method is likely to fail in discriminating the 'true' unusual events from noisy observations because both can contribute to a low value in $\log p(\mathbf{x}_t|\Theta)$, and thus cannot be distinguished by examining $\log p(\mathbf{x}_t|\Theta)$ alone.

In this study, this problem is addressed by introducing a Cumulative Abnormality Score (CAS) that alleviates the effect of noise by accumulating the temporal history of the likelihood of unusual event occurrences in each region over time. This is based on the assumption that noise would not persist over a sustained period of time and thus can be filtered out when visual evidence is accumulated over time. Specifically, an abnormality score (set to zero at $t = 0$) is computed for each node in the TD-PGM on-the-fly to monitor the likelihood of abnormality for each region. The log-likelihood of a given observation $x_{i,t}$ for the ith region at time t is computed as:

$$\log p\left(x_{i,t}|\mathbf{pa}(X_i), \boldsymbol{\theta}_{x_i|\mathbf{pa}(X_i)}\right) = \log \frac{N_{x_{i,t}|\mathbf{pa}(X_i)} + \frac{\eta}{r_i q_i}}{\sum_{k=1}^{r_i}\left(N_{x_{i,t}=k|\mathbf{pa}(X_i)} + \frac{\eta}{r_i q_i}\right)}. \tag{4.13}$$

If the log-likelihood is lower than a threshold Th_i, the abnormality score for $x_{i,t}$, denoted as $c_{i,t}$, is increased as: $c_{i,t} = c_{i,t-1} + |\log p\left(x_{i,t}|\mathbf{pa}(X_i), \boldsymbol{\theta}_{x_i|\mathbf{pa}(X_i)}\right) - \mathrm{Th}_i|$. Otherwise it is decreased from the previous abnormality score: $c_{i,t} = c_{i,t-1} - \delta\left(|\log p\left(x_{i,t}|\mathbf{pa}(X_i), \boldsymbol{\theta}_{x_i|\mathbf{pa}(X_i)}\right) - \mathrm{Th}_i|\right)$ where δ is a decay factor controlling the rate of the decrease. $c_{i,t}$ is set to 0 whenever it becomes a negative number after a decrease. Therefore $c_{i,t} \geq 0, \forall\{i,t\}$, with a larger value indicating a higher likelihood of being unusual. Note that during the computation of log-likelihood (Equation (4.13)), the activity patterns of a parent node are aligned based on the relative delay between the parent node and the child node.

A global unusual event is detected at each time frame when the total of CAS across all the regions is larger than a threshold Th, that is

$$C_t = \sum_{i=1}^{n} c_{i,t} > \mathrm{Th}. \tag{4.14}$$

Overall, there are two thresholds to be set for global unusual event detection. Threshold Th_i is set automatically to the same value for all the nodes as:

$$\mathrm{Th}_i = \overline{LL} - \sigma_{LL}^2, \tag{4.15}$$

where the $\overline{LL}$ and σ_{LL}^2 are the mean and variance of the log-likelihoods computed over all the nodes for every frame, which are obtained from a validation dataset. The other threshold Th is set according to the detection rate/false alarm rate requirement for specific application scenarios.

Once a global unusual event is detected, the contributing local activities of individual regions can be localised by examining $c_{i,t}$. Particularly, $c_{i,t}$ for all regions are ranked in a descending order. Local activities that contribute to the unusual event are then identified as those observed from the first few regions in the rank that are accounted for a given fraction $P = [0,1]$ of C_t.

4.3 Experiments

4.3.1 Datasets and Settings

Dataset – The same Station B dataset[3] used in Chapter 3 is employed here to evaluate the proposed method. The dataset was divided into ten subsets, each of which contains 5000 frames per camera ($\approx$ 2-hour in length, the last subset contains 1500 frames) . Two subsets were used as validation data. For the remaining eight subsets, 500 frames per camera from each subset were used for training and the rest for testing, *i.e.* 10% for training.

4.3.2 Global Activity Dependency Modelling

Using the training data, the nine camera views were automatically decomposed into 96 semantically meaningful regions (see Figure 3.8). Given the scene decomposition, the global activities, composed of 96 regional activities, were modelled using a TD-PGM. The model structure, which encodes the time delayed dependencies among regional activities, was initialised using pairwise TDMI and then optimised globally using the proposed two-stage structure learning method. This study employed two scoring functions, namely the BDeu score (Equation (4.10)) and the BIC score (Equation (4.11)) for the second-stage learning. As can be seen, the structure yielded using the BDeu (see Figure 4.2) score was more complex than that obtained using the BIC score (see Figure 4.3) in terms of the number of parent-child dependencies. This may not be desirable since more complex structure demands more training data to better train a model. Furthermore, additional dependencies may not necessarily better describe the inter-region relationships since some of them may be redundant and thus harmful to unusual event detection. To determine which structure to be adopted, experiments were performed to evaluate the performance of both structures in unusual event detection. Experimental results (see Section 4.3.3) shows that the structure yielded using the BIC score gave better unusual event detection performance compared to that using the BDeu score. Consequently, in the remainder of this section, only results obtained using the BIC score are reported, unless mentioned otherwise.

[3]Processed data is available at `http://www.eecs.qmul.ac.uk/~ccloy/files/datasets/liv_processed.zip`

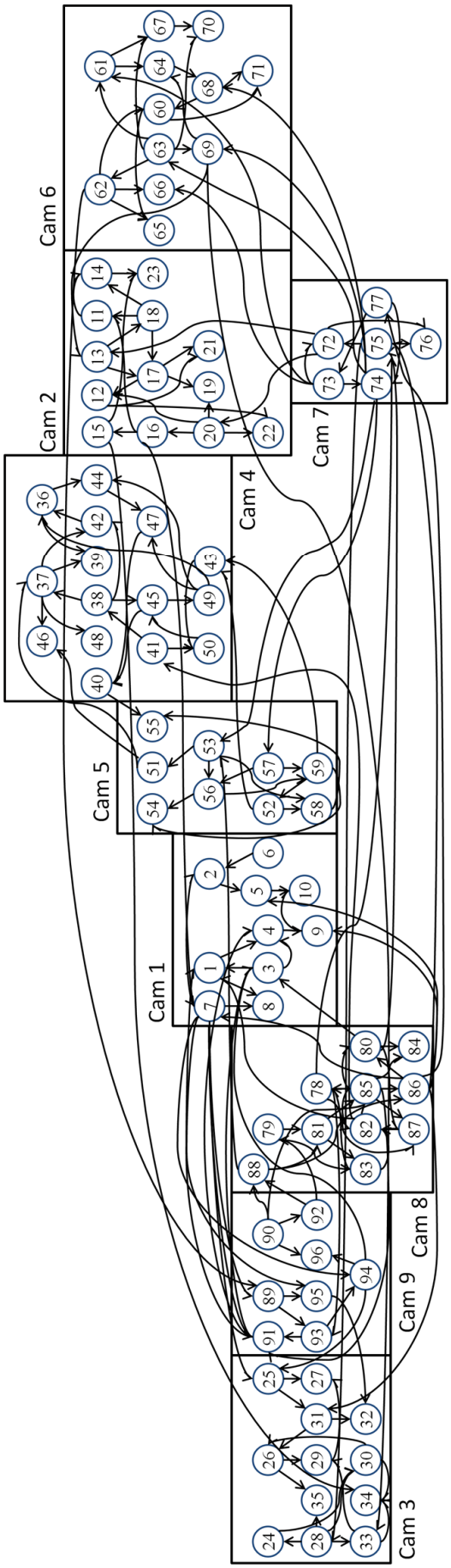


Figure 4.2: Activity global dependency graphs learned using the proposed two-stage structure learning method with BDeu scoring function. Time delays associated to graph edges are omitted in this graph.

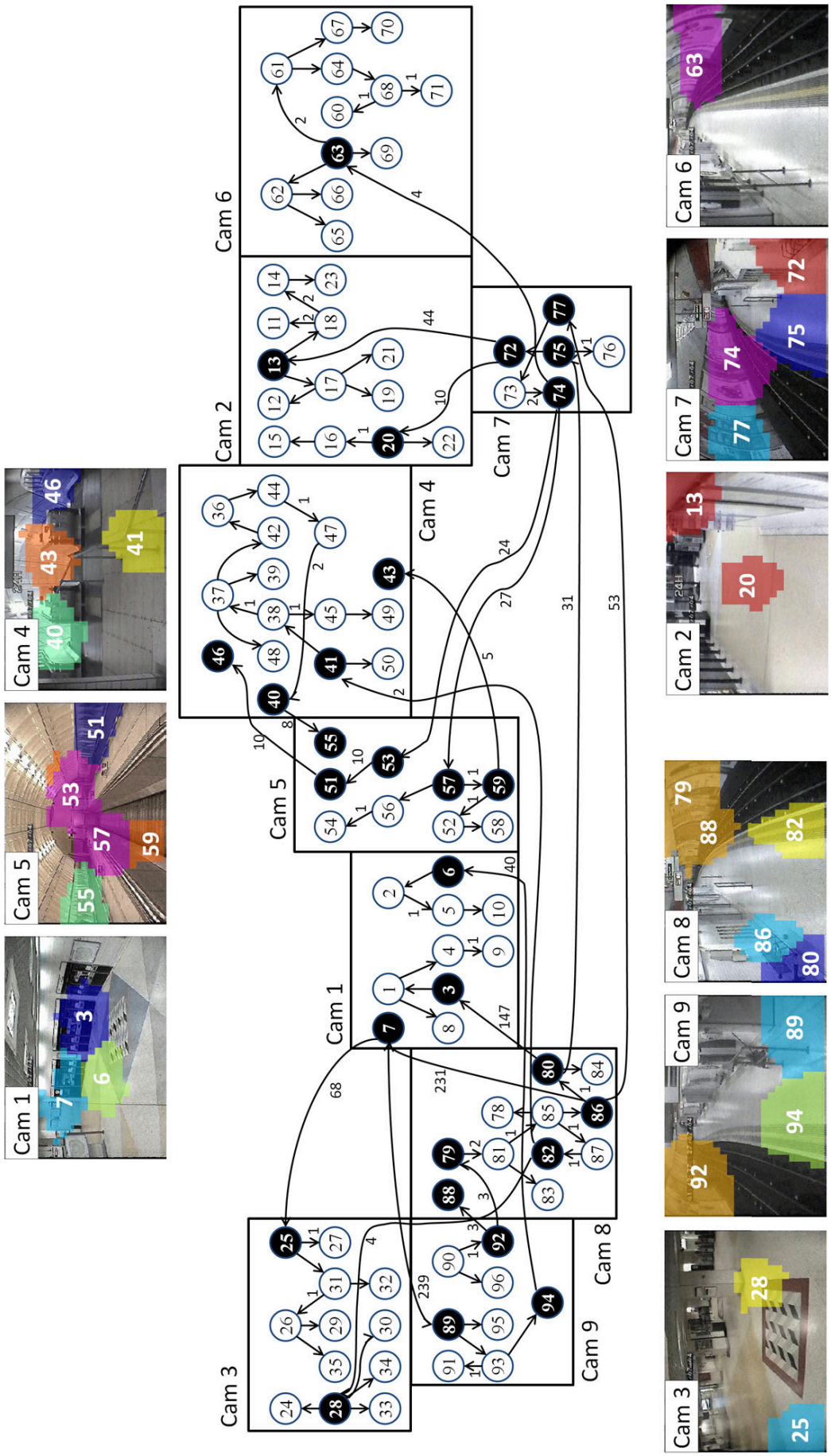


Figure 4.3: An activity global dependency graph learned using the two-stage structure learning method with BIC scoring function. Edges are labelled with the associated time delays discovered using the Time Delayed Mutual Information analysis. Unmarked edge represents dependency link with zero time delay. Regions and nodes with discovered inter-camera dependencies are highlighted.

As can be seen from Figure 4.3 (a tabular summary is given in Figure 4.4), most of the discovered dependencies were between regions from the same camera views that have short time delays. However, a number of interesting dependencies between inter-camera regions were also discovered accurately. For instance, three escalator entry and exit zones in Cam 4 (Regions 40, 43, and 46) were found to be connected with individual escalator tracks in Cam 5 (Regions 55, 59, and 51) despite some of the connecting regions not being visible in the camera views. Importantly, correct directions of edge dependency were also discovered, *e.g.* an edge pointing from upward escalator track (Region 51) towards the corresponding exit zone (Region 46). The inter-regional time delays estimated were also very close to the time gap manually observed, *e.g.* 8, 5, and 10 frames for region pairs 40-55, 43-59, 46-51 respectively. Other examples on meaningful dependencies that were discovered include $(92 \rightarrow \{88, 79\})$ with $\tau = 3$ and $(74 \rightarrow 63)$ with $\tau = 4$, which corresponded to the directions and time gaps of train movement between cameras.

The proposed method (*TDMI+K2*) was compared with four alternative dependency learning methods:

1. *MI+K2* - The proposed two-stage structure learning method but initialised using MI rather than TDMI, to demonstrate the importance of encoding time-delay.

2. *TDMI* - First-stage structure learning only, to highlight the importance of having two stages in structure learning.

3. *xCCA without K2* - Pairwise Cross Canonical Correlation Analysis (xCCA) proposed in Chapter 3 without global structure learning, to highlight the effectiveness of global dependency optimisation. Specifically, activity dependencies were obtained by considering Ψ_{ij} (see Equation (3.17)) greater than the mean Ψ_{ij} between regions of two camera views.

4. *xCCA+K2* - The proposed structure learning method but initialised using xCCA rather than TDMI, to show the advantage of modelling non-linearity among activity dependencies using TDMI for global unusual event detection.

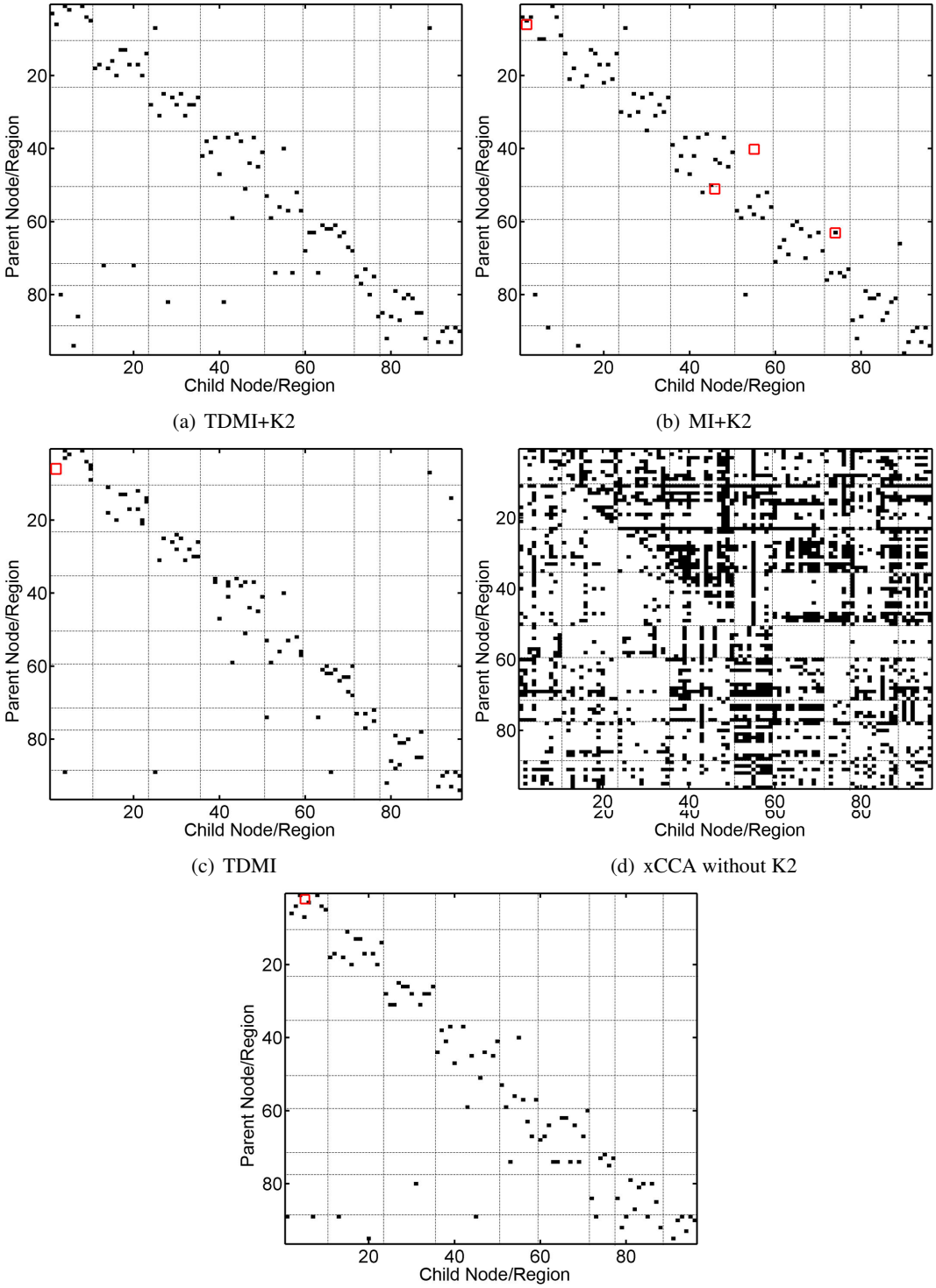
Note that the same global activity representation described in Section 4.1.1 was applied on both TDMI and xCCA based approaches.

The dependency structures discovered by the proposed method and the four alternative approaches are depicted in Figure 4.5, with some critical missed/incorrect dependency links high-

Parent → Child	Cam	Delay
1 → 4	1	0
1 → 8	1	0
2 → 5	1	1
3 → 1	1	0
4 → 9	1	1
5 → 10	1	0
6 → 2	1	0
13 → 17	2	0
13 → 18	2	0
14 → 23	2	0
16 → 15	2	0
17 → 12	2	0
17 → 19	2	0
17 → 21	2	0
18 → 11	2	2
18 → 14	2	2
20 → 16	2	1
20 → 22	2	0
25 → 27	3	1
25 → 31	3	0
26 → 29	3	0
26 → 35	3	0
28 → 24	3	0
28 → 30	3	0
28 → 33	3	0
28 → 34	3	0
31 → 26	3	1
31 → 32	3	0
36 → 44	4	0
37 → 39	4	0
37 → 42	4	0
37 → 48	4	0
38 → 37	4	1
38 → 45	4	1
41 → 38	4	0
41 → 50	4	0
42 → 36	4	0
44 → 47	4	1
45 → 49	4	0
47 → 40	4	2
52 → 58	5	0
53 → 51	5	10
56 → 54	5	1
57 → 56	5	0
57 → 59	5	1
59 → 52	5	1
61 → 64	6	0

Parent → Child	Cam	Delay
61 → 67	6	0
62 → 65	6	0
62 → 66	6	0
63 → 61	6	2
63 → 62	6	0
63 → 69	6	0
64 → 68	6	0
67 → 70	6	0
68 → 60	6	1
68 → 71	6	1
73 → 74	7	2
75 → 72	7	0
75 → 76	7	1
77 → 73	7	0
79 → 81	8	2
80 → 84	8	0
81 → 83	8	0
81 → 85	8	1
85 → 78	8	0
85 → 86	8	0
85 → 87	8	1
86 → 80	8	1
87 → 82	8	1
89 → 93	9	0
89 → 95	9	0
90 → 92	9	1
90 → 96	9	0
93 → 91	9	1
93 → 94	9	0
7 → 25	1 and 3	68
7 → 89	1 and 9	239
40 → 55	4 and 5	8
51 → 46	5 and 4	10
59 → 43	5 and 4	5
72 → 13	7 and 2	44
72 → 20	7 and 2	10
74 → 53	7 and 5	24
74 → 57	7 and 5	27
74 → 63	7 and 6	4
80 → 3	8 and 1	147
80 → 75	8 and 7	31
82 → 28	8 and 3	4
82 → 41	8 and 4	2
86 → 7	8 and 1	231
86 → 77	8 and 7	53
92 → 79	9 and 8	3
92 → 88	9 and 8	3
94 → 6	9 and 1	40

Figure 4.4: This figure summarises the graph depicted in Figure 4.3, with columns showing the parent and child nodes, the cameras where the nodes are located, and the associated time delays discovered using the Time Delayed Mutual Information analysis.



(a) TDMI+K2

(b) MI+K2

(c) TDMI

(d) xCCA without K2

(e) xCCA+K2

Figure 4.5: Activity global dependency structures learned using different methods. The y-axis represents the parent nodes, whilst the x-axis represents the child nodes. A black mark at (y,x) means $y \rightarrow x$. Some missed or false edges are highlighted using squares in red, except in (d) where there are too many.

lighted with red squares. As one shall see in the unusual event detection experiment (see Section 4.3.3), these links play an important role in unusual event detection. From Figure 4.5(b), it is observed that without taking time delay into account, MI+K2 yielded a number of missed dependency links such as $40 \rightarrow 55$ and $51 \rightarrow 46$; as well as incorrect one such as $63 \rightarrow 74$, which were against the causal flow of activity patterns. Figure 4.5(c) shows that without global dependency optimisation, structure yielded by TDMI alone was inferior to that obtained using TDMI+K2. In particular, some important dependency links such as $6 \rightarrow 2$ were still not discovered. Figure 4.5(d) clearly shows that without global dependency optimisation, most of the dependencies discovered by the method proposed in Chapter 3 were redundant. The result was greatly improved after applying the proposed structure learning method (see Figure 4.5(e)). However, some links such as $2 \rightarrow 5$ were still missing. This is due to the use of pairwise linear correlations without taking into account non-linearity among activity dependencies across regions.

4.3.3 Global Unusual Event Detection

Table 4.1: Ground truth of unusual events in Station B dataset.

Cases	Unusual Event Description	Cam	Total frames (% from total)
1-6	The queue in front of the ticket counters was built to a sufficient depth in regions 2 and 6 that it blocked the normal route from Region 2 to 5 taken by passenger who did not have to buy ticket (Figure 4.8)	1	1021 (3.14)
7-8	Faulty train observed in Cam 6 and 7 led to overcrowding on the platform. To prevent further congestion on the platform, passengers were disallowed to enter the platform via the escalator (Region 55 in Cam 5). This in turn caused congestion in front of the escalator entry zone in Cam 4 (Figure 4.9)	3, 4, 5, 6, 7	446 (1.37)
9	Train moved in reversed direction	6,7	118 (0.36)

For quantitative evaluation of the proposed unusual event detection method, ground truth was obtained by exhaustive frame-wise examination on the entire test set. Consequently, nine unusual cases were found, each of them lasting between 34 to 462 frames with an average of 176 frames (254 secs). In total, there were 1585 atypical frames accounting for 4.88% of the total frames in the test set. As shown in Table 4.1, these unusual cases fall into three categories, all of which involve multiple regional activities.

A TD-PGM learned using the proposed TDMI+K2 method was employed for unusual event

detection using the proposed CAS. One of the two free parameters, the decay factor δ of CAS was found to produce consistent results when it is set above five. Consequently, it was set to 10 throughout the experiments. The performance of the proposed approach (*TDMI+K2+CAS*) was assessed using a receiver operating characteristic (ROC) curve[4] by varying the other free parameter threshold Th. This was compared with four alternative approaches as shown in Figures 4.6 and 4.7. In addition, different scoring functions in the two-stage structure learning, namely BDeu and BIC scores were also evaluated (see Table 4.2). In this study, an unusual event is considered detected if and only if its CAS $>$ Th and at least half of the detected regions are consistent with the manually labelled regions in the ground truth.

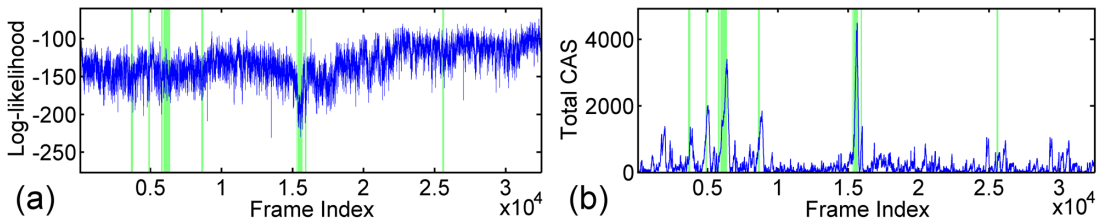


Figure 4.6: Unusual event scores computed using (a) log-likelihood (LL), and (b) cumulative abnormality score (CAS). Ground truths of unusual events are represented as bars in green colour.

CAS vs. LL - The effectiveness of the proposed CAS for unusual event detection was first examined. Specifically the proposed approach was compared with a method that uses the same TD-PGM but with the conventional LL score, denoted as TDMI+K2+LL. As can be seen from Figure 4.6, using the LL-based abnormality score, the true unusual events were overwhelmed by the noise collected from the large number of regions and thus difficult to detect. Since there were excessive number of regions falsely identified as atypical, TDMI+K2+LL essentially gave zero true positive rate across all Th, its performance is thus not available to be shown in Figure 4.7. In contrast, the proposed CAS effectively alleviated the effect of noise, thus offering superior unusual event detection performance.

Table 4.2: Comparing BDeu and BIC scoring functions for unusual event detection task.

	Area under ROC	
Method	BDeu	BIC
TDMI+K2+CAS	0.5531	0.8458
MI+K2+CAS	0.3652	0.8020
xCCA+K2+CAS	0.3674	0.7180

[4]An alternative performance metric is precision and recall curve.

TDMI+K2+CAS with different scoring functions - An experiment was conducted to examine the unusual event detection performance when different scoring functions, namely BDeu and BIC were used in the two-stage structure learning. Specifically, TD-PGMs were learned using TDMI+K2, MI+K2, and xCCA+K2 with different scoring functions used in the K2 structure learning algorithm. The area under ROC (AUROC) yielded by different methods were summarised in Table 4.2. As can be seen, BIC was superior to BDeu in all cases. This suggests that a more compact structure yielded by BIC scoring function is more suitable for unusual event detection task compared to BDeu scoring function.

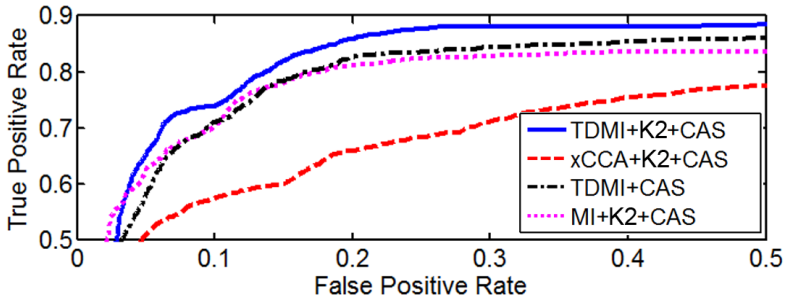


Figure 4.7: Receiving operating characteristic (ROC) curves obtained using time delayed probabilistic graphical model with different learning methods.

TDMI+K2+CAS vs. other learning methods - Experiments were carried out to further investigate how the unusual event detection performance can be affected when the global time delayed dependencies among regional activities are not learned accurately. More specifically, TD-PGMs were learned using MI+K2, TDMI alone without second-stage learning, and xCCA+K2 respectively as described above. Cumulative abnormality score was then used for unusual event detection. These three methods are denoted as MI+K2+CAS, TDMI+CAS, and xCCA+K2+CAS respectively. Note that comparison was not conducted with xCCA without structure learning (xCCA without K2) since the structure discovered was not acyclic (the proposed TD-PGM requires a structure to be acyclic in order to model activity dependencies). It is observed from Figure 4.7 that without accurate dependencies learned using the proposed TDMI+K2, all three methods yielded poorer performance. In particular, the missing time delayed dependencies shown in Figure 4.5 caused missed detection or weak response to unusual events.

Besides the K2 algorithm, another popular scored-searching method known as greedy hill-climbing (GHC) search [46] was also re-formulated for learning time delayed dependencies on the same dataset. Slightly poorer performance was obtained, with AUROC of 0.7558 and 0.3170 with BIC and BDeu scores, respectively; compared to 0.8458 and 0.5531 obtained using

TDMI+K2. The poorer detection performance of GHC method was mainly due to its weaker responses on atypical long queue events. In addition, the method proposed in [259] was also attempted by constructing a CHMM with each chain corresponded to a region. However, the model is computationally intractable on the machine employed in this study (single-core 2.8GHz with 2G RAM) due to the high space complexity during the inference stage.

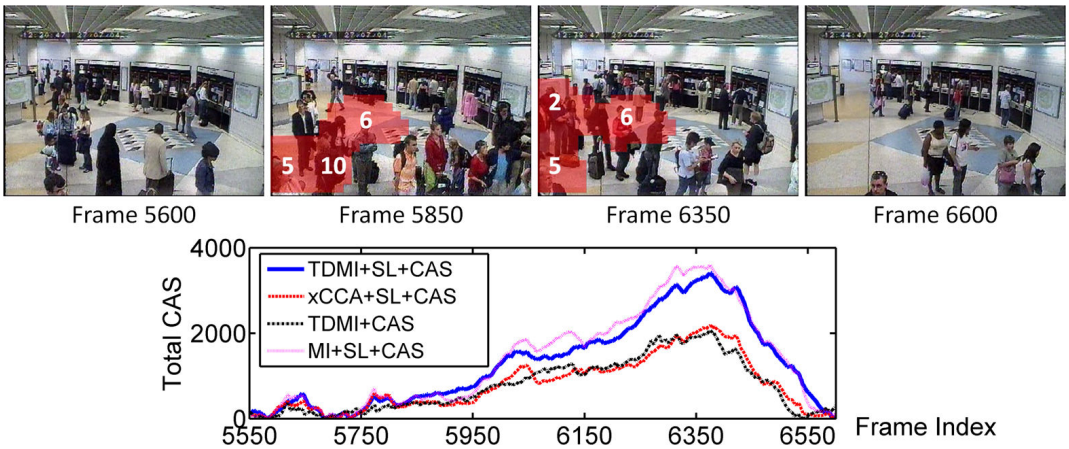


Figure 4.8: Example frames from detection output using the proposed approach on analysing unusual events caused by atypical long queues. The plot depicts the associated cumulative abnormality scores produced by different methods over the period. In ground truth, unusual events occurred at frames (5741-5853) and (5915-6376).

An example of detected unusual event using the proposed TDMI+K2+CAS approach is given in Figure 4.8. The contributing atypical regions are highlighted in red following the method described in Section 4.2 with $P = 0.8$. The atypical long queue was robustly detected using the proposed solution. In comparison, other methods such as TDMI+CAS and xCCA+K2+CAS yielded a weaker response. Note that MI+K2+CAS was able to detect this unusual event since it occurred within a single view, of which the time delays between regional activities can be ignored. One shall see in the next example, in which MI+K2+CAS failed in detecting a global unusual event that took place across multiple camera views.

Another example of unusual event detection using the proposed approach is shown in Figure 4.9. This event was Case 7 listed in Table 4.1. As can be seen, TDMI+K2+CAS detected the unusual event across Cam 3, 4, 5, 6 and 7 successfully. Specifically, TD-PGM first detected atypical crowd dynamics in Cam 6 and Cam 7, *i.e.* all train passengers were asked to get off the train. From frame 15340 to frame 15680, passengers were disallowed to use the downward escalator and therefore started to accumulate at the escalator entry zone in Cam 4. The congestion led to high CAS in several regions in Cam 4 and Cam 3 (Region 32). A large volume

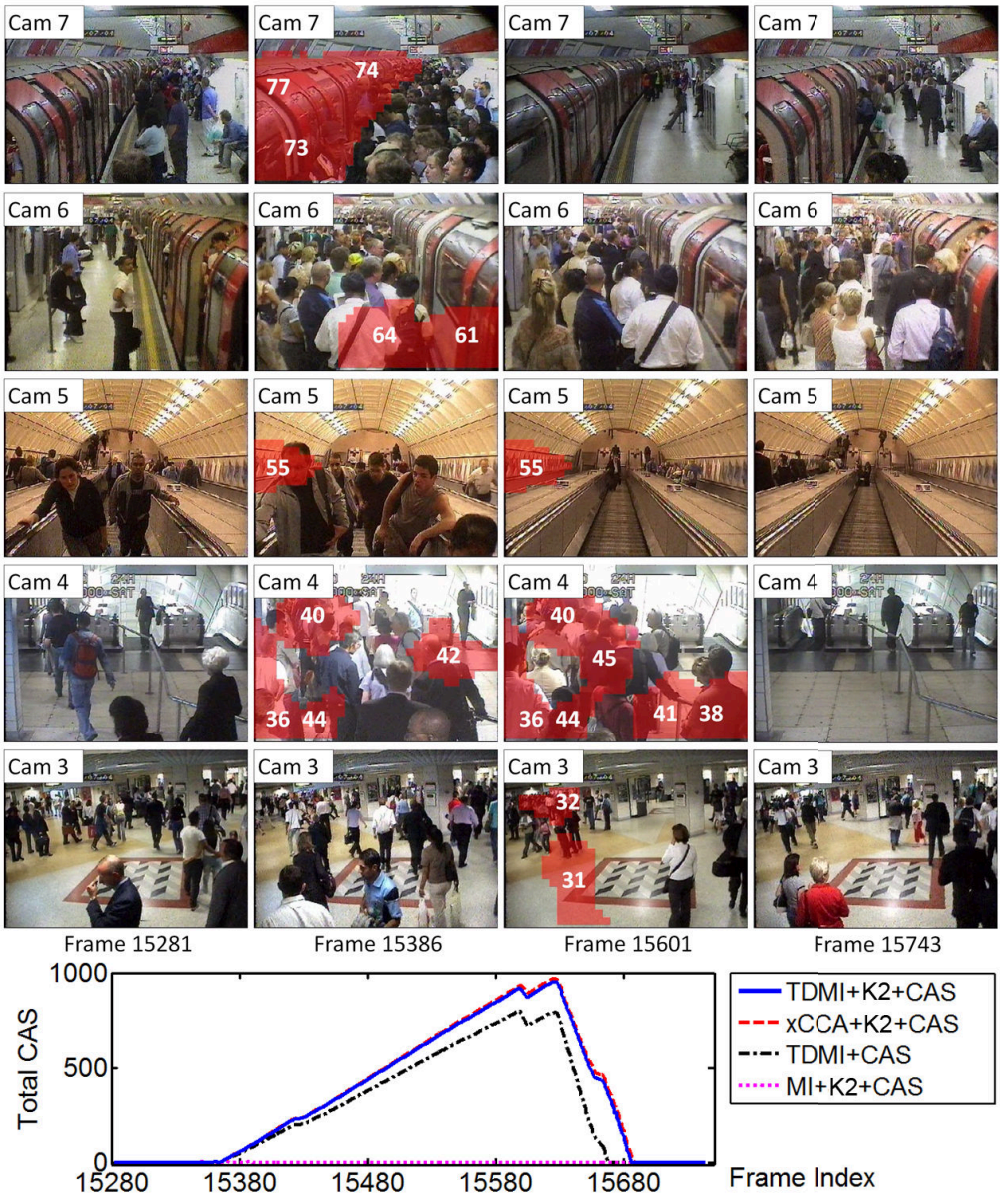


Figure 4.9: Global unusual event due to faulty train, which first occurred in Cam 6 and 7, and later propagated to Cam 5, 4, and 3. The plot depicts the cumulative abnormality scores in Region 55 produced by different methods over the period. In ground truth, this unusual event occurred at frames (15340-15680).

of crowd in Region 55 of Cam 5 was expected due to the high crowd density in Region 40 of Cam 4. However, the fact that Region 55 was empty violated the model's expected time delayed dependency, therefore causing a high CAS in Region 55 (see Figure 4.9). Despite the event in Region 55 appeared perfectly normal when examined in isolation, it was successfully detected as being unusual since the proposed approach associated Region 55 with Region 40, which has an immediate and direct causal effect to it (see Figure 4.3). In contrast, MI+K2+CAS failed to discover the time delayed dependencies between Region 40 and Region 55; it therefore missed the unusual event in Region 55. In this example, xCCA+K2+CAS yielded comparable result as the proposed approach. However, it is observed from the ROC curve (see Figure 4.7) that the overall performance of xCCA+K2+CAS was still inferior to that of TDMI+K2+CAS.

4.3.4 Computational Cost

Both first-stage and second-stage structure learning were performed off-line and took approximately 20 minutes and 2 minutes, respectively. Parameter learning required around 30 seconds. On-line detection achieved a real-time performance of 65 fps. All algorithms were implemented in Matlab on a single-Core 2.8GHz machine.

4.3.5 Discussion

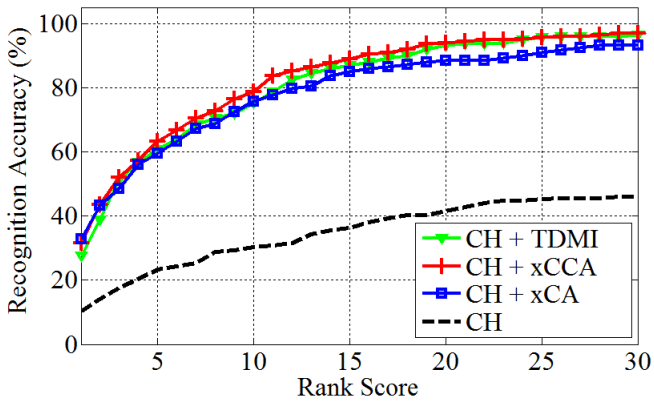


Figure 4.10: Cumulative Matching Characteristic (CMC) curve for CH+TDMI, CH+xCCA, CH+xCA and CH.

From the experimental results presented in Section 4.3.3, TDMI has shown better performance compared to the xCCA. This is not surprising since the TDMI is theoretically motivated for finding the optimal dependence tree for the proposed two-stage structure learning. In contrary, using the xCCA does not necessarily produce a tree structure that best approximates a target

distribution. In addition, TDMI also captures non-linear dependencies that are critical for global unusual event detection, whilst the xCCA can only discover linear correlations.

Nevertheless, when it comes to other activity understanding tasks such as person re-identification, pairwise regional correlations are still required. In essence, it is empirically found that the xCCA outperformed the TDMI in person re-identification experiment carried out using the same dataset employed in Chapter 3. The better performance of xCCA over TDMI in person re-identification can be seen from the CMC curve depicted in Figure 4.10. In particular, despite CH+TDMI performed better than CH+xCA and CH alone (*i.e.* true matches in the top 20 rank are 93.20% versus 88.40% and 41.60%, respectively), the overall winner was still CH+xCCA, which achieved 94.00%. In practise, the TDMI may be used to further boost the performance yielded by the xCCA. Specifically, a global activity model is first constructed to infer optimal inter-regional dependencies, which can be exploited to filter out less likely regions while performing cross camera person matching. Given the reduced searching space, the xCCA is expected to give higher matching rate.

4.4 Summary

This chapter has presented a novel approach for global unusual event detection in multiple disjoint cameras by discovering and optimising time delayed activity dependencies globally using a new two-stage structure learning method. Extensive experiments have demonstrated that the new approach outperforms methods that disregard the time delay factor or without learning dependency structure globally. In addition, the proposed cumulative abnormality score has yielded superior results in achieving robust and reliable unusual event detection compared to conventional log-likelihood score.

Nevertheless, activity dependencies are assumed to be static once learned in the current approach. The proposed method therefore cannot cope with visual context changes in dynamic environment, including gradual behaviour drift or sudden context changes involving physical changes to camera network (*e.g.* addition or removal of camera). To overcome this limitation, the following chapter presents a time and space-efficient incremental learning technique that flexibly react to visual context changes.

Chapter 5

Incremental Learning on Activity Dependency

The preceding chapter has demonstrated the importance of learning visual context in achieving reliable detection of global unusual events, of which partial observations can span across multiple camera views. Nevertheless, the proposed method assumes fixed time delayed dependency model once learned. This may be unrealistic given an unconstrained environment, in which either gradual or abrupt visual context changes are expected. In particular, gradual context change may involve gradual behaviour drift over time, *e.g*. different volumes of crowd flow at different time periods. Consequently, parameters learned previously may no longer be valid given the new interpretation. On the other hand, abrupt context change implicates more drastic changes such as camera angle adjustment, or removal/addition of cameras from/to the camera network. These changes essentially render both the current model structure and parameters obsolete.

It is thus necessary to incrementally update a model based on new observations to cope with the inevitable visual context changes. In contrast to a batch-mode learning method that performs single-round learning using a full training set, an incremental learning method outputs a model at each time point based on a stream of observations. Formally, given a new observation $\mathbf{x}_t$ at each time step t, an incremental graphical model learning method produces a model B_t with a refined structure G_t and the associated parameters Θ_t. In practice, the incremental learning process may only be invoked after collecting some number of h instances.

5.1 Naïve and MAP Approaches

Typically, one can perform a **Naïve** incremental structure learning, in which all the observations seen so far $\mathbf{x}_1, \ldots, \mathbf{x}_t$ are used to estimate B_t at every time step t. Obviously, the method should yield the optimal results since all the observed information is used for estimation. The method, however, needs to either store all the previously seen instances or keep a count of the number of times each distinct instantiation to all the variables in $\mathbf{X}$ are observed. Memory required by the former way grows linearly with the number instances collected, it is thus infeasible if the model is expected to operate for a long period of time. The latter requires only a constant amount of memory to store the count of distinct instantiations, but the memory needed can be enormous given a large number of variables. For instance, given a 50-variable graph with binary states in each variable, the number of distinct instantiations is 2^{50}. It is clear that one cannot store the counts for all possible instantiations during incremental learning.

Alternatively, one can approximate a maximum *a-posteriori* probability (MAP) model, *i.e.* a model that is considered most probable given the data seen so far. All the past observations can be summarised using the model, which is then exploited as a prior in the next learning iteration for posterior approximation. The **MAP** approach is memory efficient because it only needs to store new instances that one has observed since the last MAP update. This method, however, may lead to poor incremental learning since subsequent structures can be easily biased to the initial model [72].

5.2 Incremental Two-Stage Structure Learning

This chapter describes an extension of the two-stage structure learning described in Chapter 4 for incremental structure learning. Unlike **Naïve** and **MAP**, the proposed incremental structure learning method takes constant time regardless of the number of instances observed so far, and it is memory efficient without sacrificing the accuracy of the structure learned. In particular, this work follows a general principle introduced by Friedman and Goldszmidt [72], whereby an obsolete structure is replaced by searching from a set of most probable candidate structures at the current time, which are stored in a frontier $\mathcal{F}$. The associated sufficient statistics $\boldsymbol{\xi}$ of $\mathcal{F}$ are kept to allow tractable update of model parameters via the Bayesian learning. Although following the same principle, the proposed method differs from Friedman's method in several aspects:

1. *The proposed method allows more tractable structure update for a large camera network -*

Specifically, the prior work [72] employs a single-stage greedy hill-climbing (GHC) structure learning [46] without any constraint on structure search space. The method is thus intractable given a large graph with hundred of nodes [75]. In contrast, the proposed two-stage structure learning approach achieves a more tractable learning by exploiting the time delay information to derive an ordering constraint for reducing the search space.

2. *The proposed method requires less memory* - Previous approach [72] stores a large amount of sufficient statistics to update the dependency links given a large graph structure. In the proposed approach, inter-regional time delayed information is utilised as a constraint to prune less probable candidate structures and the associated sufficient statistics during the searching process, therefore resulting in lower memory consumption.

The detailed steps of the incremental two-stage structure learning are summarised in Algorithm 3. In the proposed approach, the structure learning process is invoked after receiving h instances, $\mathbf{x}_{t-h+1:t}$, to ensure sufficient information for learning the TDMI functions. In addition, there must be at least half of the h instances scoring below a predefined filtering threshold $\mathrm{Th}_{\mathrm{CAS}}$ during unusual event detection (Algorithm 3(L3-4)). The filtering step is introduced to prevent excessive number of outliers from being inadvertently incorporated into the model updating process. Similar to the computation of Th_i (Equation (4.15)), the threshold $\mathrm{Th}_{\mathrm{CAS}}$ is obtained from a validation set. Specifically, after one obtains Th_i as described in Section 4.2, C_t is computed for every frame and $\mathrm{Th}_{\mathrm{CAS}}$ is set to $\frac{\sum_{t=1}^{l} C_t}{2l}$, where l is the total frames of the validation dataset.

Algorithm 3: Incremental two-stage structure learning.

Input: Data stream $(\mathbf{x}_1, \ldots, \mathbf{x}_t, \ldots)$. An upper bound, φ, on the number the parents a node may have. Number of past instances to keep, h. An initial structure, G_0. A set of sufficient statistics, $\boldsymbol{\xi}_0 = \xi(G_0)$. Update coefficient β.

Output: G_t and $\boldsymbol{\xi}_t$.

1 **for** t *from* $1, 2, \ldots$ **do**

2 $\quad G_t = G_{t-1}$, $\boldsymbol{\xi}_t = \boldsymbol{\xi}_{t-1}$;

3 $\quad$ Receive $\mathbf{x}_t$. Compute C_t [Equation (4.14)] ;

4 $\quad$ **if** $t \bmod h = 0$ *and* $|\{C_i | t-h+1 \leq i \leq t, C_i < \mathrm{Th}_{\mathrm{CAS}}\}| \geq \frac{h}{2}$ **then**

$\quad\quad$ `// Stage One`

5 $\quad\quad$ Compute $\mathcal{I}(\tau)$ using $\mathbf{x}_{t-h+1:t}$ [Equations (4.2) and (4.3)] ;

6 $\quad\quad$ **if** $t = 1$ **then**

7 $\quad\quad\quad$ Set $\mathcal{I}^{\mathrm{acc}}(\tau) = \mathcal{I}(\tau)$;

8 $\quad\quad$ **else**

9 $\quad\quad\quad$ Update $\mathcal{I}^{\mathrm{acc}}(\tau)$ using $\mathcal{I}(\tau)$ with updating rate β [Equation (5.1)] ;

10 $\quad\quad$ **end**

11 $\quad\quad$ $\overline{\mathcal{I}}^{\mathrm{acc}}(\tau) = \mathcal{I}^{\mathrm{acc}}(\tau)$;

12 $\quad\quad$ Compute $\mathbf{D}$ and $\mathbf{I}$ using $\mathcal{I}^{\mathrm{acc}}(\tau)$ [Equations (4.4) to (4.7)] ;

13 $\quad\quad$ Find the ordering of variables $\prec$ [Algorithm 1] ;

$\quad\quad$ `// Stage Two`

14 $\quad\quad$ Create $\mathcal{F}$ based on $\prec$ and G_{t-1} ;

15 $\quad\quad$ Obtain $\boldsymbol{\xi}_t$ by updating each record in $\boldsymbol{\xi}_{t-1}$ using $\mathbf{x}_{t-h+1:t}$;

16 $\quad\quad$ Search for the highest scored G_t from $\mathcal{F}$ [Algorithm 2];

17 $\quad$ **end**

18 **end**

5.2.1 Finding a Topological Order $\prec$

Similar to the batch-mode learning described in Chapter 4, there are two stages in the proposed incremental structure learning procedure. The learning process commences with the estimation of ordering of variables $\prec$ in the first-stage learning (Algorithm 3(L5-13)).

In particular, up-to-date cumulative TDMI functions $\mathcal{I}^{\mathrm{acc}}(\tau)$ for each pair of regional activity

patterns are first estimated by accumulating past TDMI functions. Specifically, $\mathcal{I}_{ij}^{\text{acc}}(\tau)$ between the ith region and the jth region is estimated as follows:

$$\mathcal{I}_{ij}^{\text{acc}}(\tau) = \beta\overline{\mathcal{I}}_{ij}^{\text{acc}}(\tau) + (1-\beta)\mathcal{I}_{ij}(\tau), \tag{5.1}$$

where β denotes an update coefficient that controls the updating rate of the function, $\overline{\mathcal{I}}_{ij}^{\text{acc}}(\tau)$ represents the cumulative TDMI function found in previous learning iteration, and $\mathcal{I}_{ij}(\tau)$ denotes a TDMI function computed using $\mathbf{x}_{t-h+1:t}$. Given $\mathcal{I}_{ij}^{\text{acc}}(\tau)$, one can obtain the updated $\mathbf{I}$, $\mathbf{D}$, and $\prec$ using the procedures described in Section 4.1.3.

5.2.2 Building a Frontier $\mathcal{F}$

After obtaining $\prec$, the method proceeds to the second-stage learning. A frontier $\mathcal{F}$ is first constructed based on $\prec$ and a structure estimated in previous iteration G_{t-1} (Algorithm 3(L14)). Formally, $\mathcal{F}$ is defined by a set of *families* composed of X_i and its parent set $\mathbf{Pa}(X_i)$:

$$\mathcal{F} = \{(X_i, \mathbf{Pa}_j(X_i)) \mid 1 \leq i \leq n,\ 1 \leq j \leq \Omega\}, \tag{5.2}$$

where Ω denotes the total number of different parent sets $\mathbf{Pa}_j(X_i)$ associated with X_i.

Note that a node may be associated with multiple different parent sets. The reason is that we construct $\mathcal{F}$ by including existing families in G_{t-1} as well as using different combinations of candidate parents of X_i consistent with $\prec$. With this strategy, $\mathcal{F}$ is enriched by a diverse set of candidate structures that could be simpler or more complex than G_{t-1} through combining different families in $\mathcal{F}$.

To prevent proliferation of parent set combinations and to constrain the search space to a set of most promising structures for incremental learning, one needs to *prune less probable structure candidates* from joining the final scoring process. In particular, different combinations of parent sets for X_i is formed by selecting only a set of most probable parents, $\mathbf{mpp}_i$ consistent with $\prec$, with $|\mathbf{mpp}_i| \leq \varphi < n$. Here, φ denotes the maximum number of parents a node may have and $\mathbf{mpp}_i$ contains parents that return the highest TDMI among other candidate parents. For instance, given $\varphi = 2$ and the $\mathbf{mpp}$ of X_1 are X_2 and X_3, the possible parents sets of X_1 would be $\{(X_2), (X_3), (X_2, X_3), (\emptyset)\}$. In general, the maximum number of parent combinations a node may

have is given as:

$$\Omega = 1 + \sum_{k=1}^{\varphi} \binom{\varphi}{k}. \tag{5.3}$$

There are different ways of constructing the frontier $\mathcal{F}$, depending on different scored-searching based structure learning methods. For example, if one chooses to employ a greedy hill-climbing (GHC) algorithm [46], $\mathcal{F}$ would consist of families corresponding to all neighbours that are one change away (with a single edge deletion, insertion and reversal) from the current structure G. Alternatively, it is also possible to use a beam searching method [192], which only maintains a predetermined number of candidate graph structures and all the associated neighbours.

5.2.3 Updating Sufficient Statistics $\boldsymbol{\xi}$

Since $\mathcal{F}$ at time t may be different from that at $t-1$, one needs to update the associated sufficient statistics of each family in $\mathcal{F}$ (Algorithm 3(L15)). In a model characterised by multinomial distributions, the sufficient statistics that quantify its CPDs are defined as:

$$\xi(G) = \left\{ \mathbf{N}_{X_i|\mathbf{Pa}(X_i)} \mid 1 \leq i \leq n \right\}, \tag{5.4}$$

where $\mathbf{N}_{X_i|\mathbf{Pa}(X_i)} = \left\{ N_{x_i|\mathbf{pa}(X_i)} \right\}$ are extracted from $\mathbf{x}_{t-h+1:t}$ and a set of such sufficient statistics at time t is denoted as $\boldsymbol{\xi}_t$. Given $\mathcal{F}$, $\boldsymbol{\xi}_t$ is updated based on $\mathbf{x}_{t-h+1:t}$ as follows:

$$\boldsymbol{\xi}_t = \boldsymbol{\xi}_{t-1} \bigcup \left\{ \mathbf{N}_{X_i|\mathbf{Pa}(X_i)} \mid (X_i, \mathbf{Pa}(X_i)) \in \mathcal{F} \right\}. \tag{5.5}$$

The updated sufficient statistics $\boldsymbol{\xi}_t$ will then be used in the next step for structure scoring. Note that after the incremental two-stage structure learning, $\boldsymbol{\xi}_t$ will also be used for parameter update via the Bayesian learning as described in Section 4.1.4.

5.2.4 Scoring a Structure

The goal of this step is to search for an optimal structure G_t within $\mathcal{F}$ to replace G_{t-1} (Algorithm 3(L16)). This is achieved by comparing the scores returned from a set of candidate structures that can be evaluated using the records in $\boldsymbol{\xi}_t$, that is:

$$G_t = \underset{\{G' \mid \xi(G') \subset \boldsymbol{\xi}_t\}}{\operatorname{argmax}} \; S^*(G' | \boldsymbol{\xi}_t), \tag{5.6}$$

where $S^*(\cdot)$ denotes a modified version of the original score $S(\cdot)$ defined in Equations (4.10) and (4.11).

As pointed out by Friedman and Goldszmidt [72], the scores need to be modified because one may start collecting new sufficient statistics or may remove redundant ones at different times, due to addition/removal of families from $\mathcal{F}$ during the incremental structure learning. The numbers of instances $\mathbf{N}_{X_i|\mathbf{Pa}(X_i)}$ recorded in a family's sufficient statistics would affect the final score value, *e.g.* a lower score may be assigned to a family that observes more instances. To avoid unfair comparison of different candidate structures, it is thus necessary to average the score yielded by each family with the total instances recorded in its sufficient statistics. In particular, this work follows the method proposed by Friedman and Goldszmidt [72] to modify S_{BDeu} and S_{BIC}:

$$S^*_{\text{BDeu}}(X_i|\mathbf{Pa}(X_i)) = \frac{S_{\text{BDeu}}(X_i|\mathbf{Pa}(X_i))}{\sum\limits_{(x_i|\mathbf{pa}(X_i))} N_{x_i|\mathbf{pa}(X_i)}}. \tag{5.7}$$

$$S^*_{\text{BIC}}(X_i|\mathbf{Pa}(X_i)) = \frac{S_{\text{BIC}}(X_i|\mathbf{Pa}(X_i))}{\sum\limits_{(x_i|\mathbf{pa}(X_i))} N_{x_i|\mathbf{pa}(X_i)}}. \tag{5.8}$$

Since the proposed method includes the previous graph structure G_{t-1} in $\mathcal{F}$ and its sufficient statistics in every learning iteration, the incremental learning procedure shall improve monotonically as it must return a structure G_t that scores at least as well as G_{t-1}, *i.e.* $S^*(G_t|\boldsymbol{\xi}) \geq S^*(G_{t-1}|\boldsymbol{\xi})$.

5.3 Experiments

5.3.1 Gradual Context Change

This experiment was similar to the global unusual event detection experiment reported in Section 4.3.3. In this experiment, however, a model was no longer trained using data subsets obtained from different time periods, but initialised using only training data extracted in the morning (5:42am - 9:42am) and updated using subsequent observations using an incremental structure learning method. The goal of this experiment is to compare the proposed incremental structure learning approach (**Incremental**, described in Algorithm 3) with three alternative strategies in dealing with gradual context changes, *e.g.* crowd flow transitions at different time periods. The three methods were:

1. **ParamAlone** - this method only update individual parameters without adapting the struc-

ture of a model.

2. **Naïve** - this method stores all past observations for incremental structure learning (described in Section 5.1).

3. **MAP** - this method uses the best model so far as a prior for subsequent structure learning (described in Section 5.1).

The aforementioned methods were evaluated on the same Station B underground dataset employed in Chapters 3 and 4 (see Section 3.3.1 for a detailed description on the dataset). Specifically, two subsets in the morning period were used to initialise a model. A subsequent subset was reserved as validation data to compute the thresholds Th_i and $\mathrm{Th}_{\mathrm{CAS}}$. Other subsets were employed for testing and incremental learning. For all methods, the following settings were used: a weak uniform prior with equivalent sample size $\eta = 10$ (see Equation (4.10)), a slow updating rate with update coefficient $\beta = 0.9$, and an upper limit of the number of parents a node may have $\varphi = 3$. All incremental structure learning approaches generated an updated model by invoking the TDMI+K2 structure learning along with individual incremental learning scheme, together with the parameter learning every time $h = 500$ instances were observed. After each learning iteration, the updated model was employed for unusual event detection on subsequent observations.

Since better results were obtained using the BIC score in batch-mode learning (see Table 4.2 in Chapter 4), **Naïve** was carried out with the BIC score and **Incremental** with the modified BIC score (Equation (5.8)). The BIC score, however, is not suitable for the **MAP** method if one wishes to take into account the prior information represented in a MAP model. Therefore, a modified BDeu score (Equation (5.7)) [72] was employed for **MAP**.

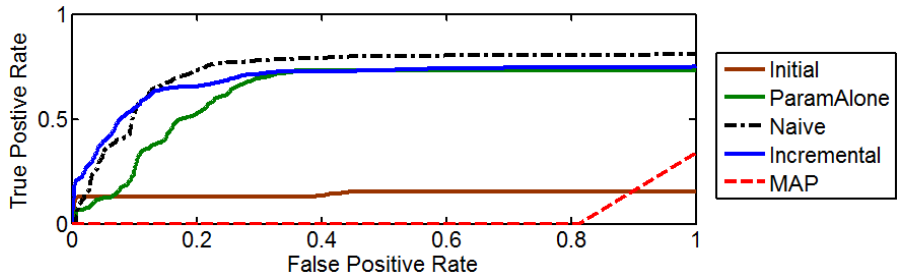


Figure 5.1: Receiver operating characteristic (ROC) curves obtained using different incremental structure learning methods.

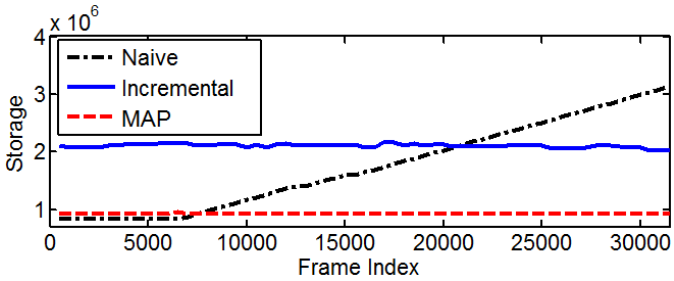


Figure 5.2: Memory requirement of different incremental structure learning methods.

Similar to the experiment reported in Section 4.3.3, the performance of an approach was assessed using a ROC curve, which was generated by varying the threshold Th. The ROC curves yielded by a baseline method **Initial** (*i.e.* an initial model was used without any structure/parameter update), **ParamAlone**, **Naïve**, **Incremental**, and **MAP** are shown in Figure 5.1. The memory requirement[1] associated with different incremental structure learning methods is also given in Figure 5.2. Poor detection performance (AUROC = 0.1413) of **Initial** is expected since the initial model only accessed observations in the morning period, which was quiet most of the time. It therefore failed to cope with busier context in the subsequent subsets. Among three incremental structure learning approaches, it was found that **Naïve** yielded the best unusual event detection performance, with AUROC of **0.7303**. However, its memory requirement increased linearly along with the number of observations seen, as depicted in Figure 5.2. Despite **MAP** needed the least memory, it was trapped in a wrong structure and subsequently locked to it, yielding the poorest result (AUROC = **0.0323**) among all methods. Overall, **Incremental** gave comparable detection performance compared to **Naïve**, with an AUROC of **0.6851**. Importantly, memory required by **Incremental** remained constant throughout the test by keeping a handful of sufficient statistics (see Figure 5.2). In comparison to **Naïve** and **Incremental**, inferior performance was observed on **ParamAlone**, with an AUROC of **0.6278**. This suggests that updating parameters alone may still be inadequate when dealing with gradual visual context changes. Nonetheless, it was still better than maintaining a fixed model's parameters without incremental update.

It is observed that the AUROC of **Incremental** was slightly lower than that obtained using batch-mode learning (see Sec. 4.3.3). This is due to incorrect detection of the long queue event in Cam 1 (see Fig. 4.8) by **Incremental**. Such a long queue event was rare over the whole train-

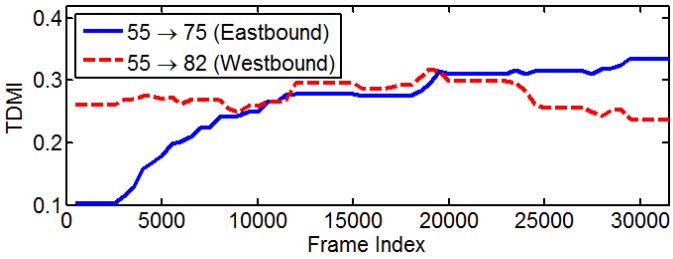
[1]This study estimates of memory usage of **Naïve** and **MAP** based on the number of instances. For **Incremental**, the memory was measured based on the space needed to store the sufficient statistics.

ing set, it was thus assigned as unusual event in the ground truth. However, the event was more frequent at certain period of the time. Consequently, it was determined by the unsupervised **Incremental** method as being usual event in those periods. This caused miss-detections following the same ground truth employed in the batch-mode experiments (Sec. 4.3.3).

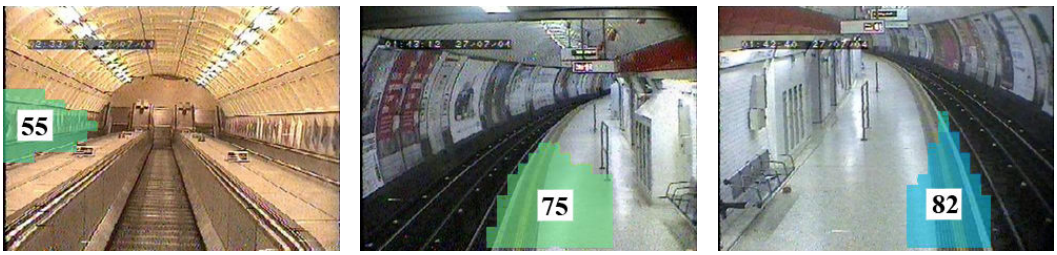
(a) Cam 4 (b) Cam 5

Figure 5.3: The figure shows the decomposed regions of Cam 4 and Cam 5 in Station B dataset. Note that Region 40 is closer to Region 55 spatially, it therefore has a more direct causal impact to Region 55 compared to Region 45.

To give further insights, the differences between the initial model and the up-to-date model induced using **Incremental** were investigated. With the proposed incremental structure learning, it was observed that some errors in the initial model were corrected, *e.g.* initial dependency link 45 → 55 was corrected to 40 → 55, in which Region 40 has a more direct causal impact to Region 55 (see Figure 5.3). In another example, an incorrect time delay estimated at the beginning between Regions 6 and 7 was also corrected from 34 to 2 frames.



(a) Inter-regional dependency changes



(b) Region 55 in Cam 5 (c) Region 75 in Cam 7 (d) Region 82 in Cam 8

Figure 5.4: Inter-regional dependency changes captured using the proposed incremental two-stage learning.

The proposed incremental learning approach also learned meaningful changes of inter-regional dependency strength over time. In an example shown in Figure 5.4, since passengers mostly commuted to the city centre in the morning/afternoon periods, Region 82 (westbound platform toward city centre) thus exhibited a stronger dependency with Region 55 (downward escalator to platforms), as compared to Region 75 (eastbound platform toward residential areas). However, this scenario changed in the late afternoon/evening when people began to travel back home. Hence, eastbound platform became busier than the westbound platform, starting around frame 20000, *i.e.* 6-7pm. Whilst the westbound platform remained busy as many commuters took a train from this platform to transit to other stations, thus it still maintained a strong connection with Region 55. It is evident from Figure 5.4 that the proposed incremental learning method was able to capture this dependency transition.

5.3.2 Abrupt Context Change

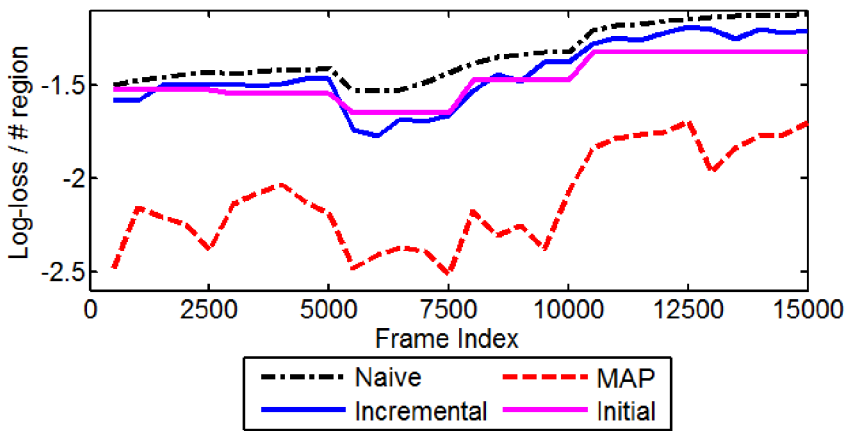


Figure 5.5: This figure shows the log-loss performance yielded by different incremental structure learning methods and a model without incremental learning, in a scenario where Cam 5 was removed at frame 7500. Note that the log-loss values were normalised by the number of decomposed regions.

The goal of this experiment is to evaluate how well an incrementally trained model can adapt to visual context undergoing abrupt changes. Two scenarios were tested: removal and addition of cameras in a camera network. In the first scenario, all observations from Cam 5 were discarded starting from frame 7500 to simulate a faulty camera or removal of camera. The second scenario began with eight cameras, and Cam 5 was attached to the network starting from frame 7500. Note that the incremental structure learning process may branch into slightly different routines depending on the nature of the abrupt change. For instance, if an existing camera is removed from

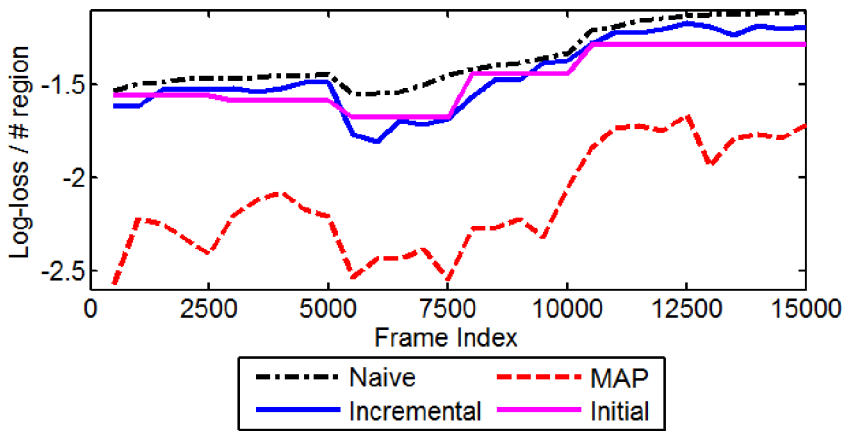


Figure 5.6: This figure shows the log-loss performance yielded by different incremental structure learning methods and a model without incremental learning, in a scenario where Cam 5 was added at frame 7500. Note that the log-loss values were normalised by the number of decomposed regions.

a network, candidate structures associated with the camera would be discarded automatically from the frontier $\mathcal{F}$. New dependencies between remaining regions are then quickly learned through structure learning process based on sufficient statistics accumulated so far. Conversely, if a new camera is added to the network, new TDMI functions and candidates would be created automatically to accumulate sufficient statistics for learning an updated structure.

The goodness of adaptation was evaluated using a common measure of density estimation performance, known as *log-loss* [65], which is defined as:

$$l_{\text{loss}} = \frac{1}{m}\sum_{t=1}^{m} \log p(\mathbf{x}_t|\Theta), \tag{5.9}$$

where m is the total number of test cases. In this experiment, l_{loss} was further divided using the total number of decomposed regions in a camera network, so that a fair comparison can be performed between two networks with different numbers of cameras and decomposed regions. Recall that the underground dataset was divided into ten subsets; 2500 frames from each subset (except for the three subsets for initialisation and validation) were used for incremental structure learning and the rest in a subset were reserved as test samples for log-loss computation. Similar to experiments conducted in Section 5.3.1, all structure learning approaches invoked structure learning and parameter learning every $h = 500$ instances.

The results on both scenarios are depicted in Figures 5.5 and 5.6. In both scenarios, the performance of **Naïve** represented the optimal adaptation to the current visual context since it

learned a new structure in every iteration using all the past observations. As one can observe from Figures 5.5 and 5.6, **MAP** showed a much lower log-loss compared to **Naïve**. This is because it was locked to a poor structure initially and failed to infer a proper structure to adapt to the current visual context based on limited information obtained from the prior model. In contrast, **Incremental** exhibited closer performance to **Naïve** by just maintaining a small amount of sufficient statistics. Note that there was a drop of log-loss performance over all methods from frame 5000 to 7500 owing to a global unusual event due to a faulty train (see Figure 4.9 for example frames of this unusual event). Without support from all previously seen observations, **Incremental** exhibited a larger drop as compared to **Naïve** during the occurrence of the unusual incident, causing a log-loss gap between **Incremental** and **Naïve** methods. Cam 5 was added/removed right after the end of unusual incident at frame 7500. The log-loss gap remained between **Incremental** and **Naïve** after frame 7500 since **Incremental** needed to accumulate new sufficient statistics for the learning of new dependency links when Cam 5 was added/removed from the network. Nevertheless, it quickly approached the distribution modelled by **Naïve** thereafter. It is observed from Figures 5.5 and 5.6 that without incremental structure learning (**Initial**), a model was not able to adapt to the current visual context, resulting in relatively lower log-loss performances (also further away from the optimal performance yielded by **Naïve**) as compared to **Incremental** after a camera was added/removed from the camera network.

It is observed that when Cam 5 (with its FOV "connecting" escalator entry/exit zones in Cam 4 and the platforms) was removed, **Incremental** was able to infer new dependency link between Cam 4 and the platforms (*e.g.* $74 \rightarrow 43$ with $\tau = 57$). When Cam 5 was added, it was also capable of adapting to the rapid context change and infer new links between Cam 4 and Cam 5, *e.g.* $51 \rightarrow 46$ in fewer than seven learning iterations (please refer Figure 3.8 for the decomposed regions).

Let us now discuss the run time cost of the proposed approach. The approach was implemented in Matlab on a single-core 2.8 GHz machine. In every incremental learning iteration, the first stage of the proposed method took an average of 1.61 minutes, mostly spent on the estimation of the pairwise TDMI function. The second stage of the method took approximately 0.04 minutes. Given a learning iteration of $h = 500$, which is around 12 minutes for video with 0.7 fps, there is sufficient time to perform the two-stage structure learning in every iteration. In particular, one may halt the detection routine temporarily at the beginning of each learning iteration to give

way to the incremental structure learning process. The updated model produced by the structure learning algorithm can then be employed for detection on previously skipped data followed by observations thereafter, thus achieving a near real-time performance.

5.3.3 Comparison with Incremental Greedy Hill-Climbing Structure Learning

It is found that the original incremental structure learning method proposed by Friedman and Goldszmidt [72] was intractable given a large graph. Given the flexibility of our incremental two-stage structure learning method, one can replace the K2 algorithm with the GHC algorithm used in [72], so that the latter can take advantage of the ordering constraint for reducing the search space. Similar experimental procedures described in Sec. 5.3.1 were carried out on the GHC-based two-stage structure learning method. It is found that the second-stage GHC search required an average of 22 minutes to infer a new structure in every round of incremental learning, as compared to 0.04 minutes needed by the proposed K2-based approach. In addition, it yielded a poorer result, *i.e*. AUROC of **0.6369**, than the proposed K2-based approach, which achieved an AUROC of **0.6851**.

5.4 Summary

This chapter has extended the two-stage structure learning approach proposed in Chapter 4 to allow incremental structure adaptation on the TD-PGM. Contrary to most existing methods that assume static model once trained, the proposed approach update the activity model's parameters and structure incrementally and adaptively to adapt to the current visual context. Importantly, the proposed solution is computational and memory efficient, it can thus be applied to surveillance applications that demand real-time performance. Experimental results on the public scene data have demonstrated the effectiveness of the proposed incremental approach in dealing with both gradual and abrupt context changes.

The unusual detection approach presented in both Chapters 4 and 5 is essentially based on one-class unsupervised learning approach, in which a model is trained based solely on normal events, and outliers are identified as those that differ from the distribution of ordinary data. This learning scheme may not be able to detect subtle unusual events that are visually similar to a large number of normally behaving objects co-existing in a scene. In addition, it may have difficulty in distinguishing a genuine unusual event from noise since there is no mechanism to exploit

flagged unusual events to refine subsequent detections. The next chapter addresses this problem by learning from human feedback using an active learning approach.

Chapter 6

Stream-based Active Unusual Event Detection

Chapters 4 and 5 follow an outlier detection strategy for unusual event detection, which is adopted by most existing methods [104, 131, 155, 231]. In this strategy, a definition of normal behaviours is constructed using a large amount of samples from ordinary events, assuming that the modelled distribution will yield a low probability on unusual events that one wishes to detect. This strategy essentially relies on passive mining on events, which would have difficulty in distinguishing true unusual events from noise and uninteresting outliers (discussed in Section 1.2.2).

To overcome these intrinsic limitations of unsupervised learning, other source of information needs to be exploited. Human feedback is readily available in most surveillance scenarios to remedy the aforementioned issues. In particular, active learning strategy [197] emerges as a compelling alternative to conventional random sampling strategy and unsupervised unusual event detection methods, since it can automatically seeks human feedback on critical instances according to predefined query criteria. The feedback can then be exploited for resolving ambiguities, so as to achieve a more accurate and relevant detection of subtle unusual events.

However, most conventional active learning approaches assume a pool-based setting, whereby queries are selectively drawn from a pool of unlabelled data in a greedy fashion, *i.e.* the entire collection must be ranked to select the best query. These methods are thus often used in batch mode due to potentially large consumption of memory and processing power. In addition, most existing active learning methods depend only on a single criterion, such as likelihood or uncertainty criterion, to select a query from a pool of data, with an assumption that all data classes are balanced in number (see Section 2.4.4 for a review). Applying these approaches in surveillance

tasks is nontrivial due to several factors:

1. *Dynamic and stream-based observations* - Activity patterns in public scenes are dynamic. In addition, unusual events are unpredictable and not always readily available *a priori*. Moreover, a surveillance system receives a stream of unlabelled video data continuously that demands real-time processing. Hence, an adaptive stream-based active learning method is preferred over a pool-based active learning. A stream-based setting is intrinsically more difficult than a pool-based setting because the active learner has to make decision on-the-fly without complete knowledge on the underlying data distribution [100, 126].

2. *Joint discovery of unknown events and classifier learning* - Some classes, especially unusual event classes have to be discovered since they are not available in the early stage of training. At the same time, a classifier needs to learn the correct decision boundary. Hence, single criterion alone, either likelihood criterion [177, 209] for exploration (finding unknown events) or uncertainty criterion for exploitation [121, 125] (refining the classification boundary) is insufficient.

3. *Imbalanced dataset* - The proportion of data in different classes is highly skewed, *i.e.* most of the samples correspond to normal event classes whilst the remaining interesting classes only constituent a small percentage of the entire dataset. Given a dataset with imbalanced class distribution, classification boundary can be severely skewed toward normal classes [67], leading to inaccurate unusual event detection.

In this chapter, a novel active learning approach is formulated for unusual event detection. Unlike existing active learning approaches, the method does not assume a fixed pool of data, and it is capable of making an immediate decision whether to request for labels when new video data are observed in sequence. Formally, this work considers active learning in a stream-based setting in which an unlabelled sample $\mathbf{x}_t$ is observed at each time step t from an input data stream $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_t, \ldots)$. Consequently, a classifier $\mathcal{C}_t$ is required to determine on-the-fly whether or not to request for label y_t or discard $\mathbf{x}_t$. The goal of this work is to select critical samples from $\mathcal{X}$ for annotation to achieve two tasks simultaneously: (1) to discover unusual event classes or unknown region of existing classes in the input feature space and (2) to refine the classification boundary with higher priority being given to regions surrounding the unusual classes so as to improve the detection accuracy of unusual events.

Consequently, the data presented to human annotator is not selected based on a single criterion, but via an adaptive selection process involving two criteria. The first query criterion is a *likelihood criterion*, which favours samples that have a low likelihood w.r.t. the current model. As a result, unknown classes or unexplored regions of existing classes can be discovered. Note that the proposed method does not assume the availability of predefined classes, *i.e*. once a new class is discovered, the model will expand itself automatically. The second criterion is an *uncertainty criterion* based on a modified Query-by-Committee (QBC) algorithm [9, 152, 199]. It is used to refine the decision boundary by selecting controversial samples in uncertain regions that give rise to the most disagreement among classes, with more emphasis given to the regions surrounding unusual event classes to address the problem of imbalance class distribution.

The two query criteria are dynamically re-weighted based on relative entropy or Kullback-Leibler (KL) divergence [53] measured on the model before and after it is trained using a queried sample. The premise behind this adaptive weighting scheme is to favour the criterion that is more likely to return an informative queried sample that brings most influence to the current model. Intuitively, the likelihood criterion will be preferred at the beginning of the learning process since unknown events will cause a greater impact to the parameter change. When sufficient samples from different classes are discovered, uncertainty criterion will dominate because regions in the input space have been well discovered, so further exploration using the likelihood criterion is less likely to impart a large parameter change to the model.

Note that there is a number of measures that can be used to measure the difference between two distributions, such as Hellinger distance [181] and Wasserstein distance [61]. The KL divergence is chosen in this study due to a variety of properties that make it particularly suitable as a measure of difference between distributions [184]. One of such properties is the intuitive information theoretic interpretation of KL divergence, which closely relates it to the gain of information or decrease in uncertainty. This characteristic matches the goal of the proposed adaptive selection scheme, which aims to select an informative point to improve the current model.

Comparative experiments are carried out on busy public space surveillance videos. This work shows that by exploiting a small cost of human supervision through active learning, more robust and accurate detection of subtle unusual events is achieved compared to conventional random sampling strategy and unsupervised learning methods. In addition, the results also suggest that the proposed adaptive multi-criteria approach outperforms single criterion and multi-criteria

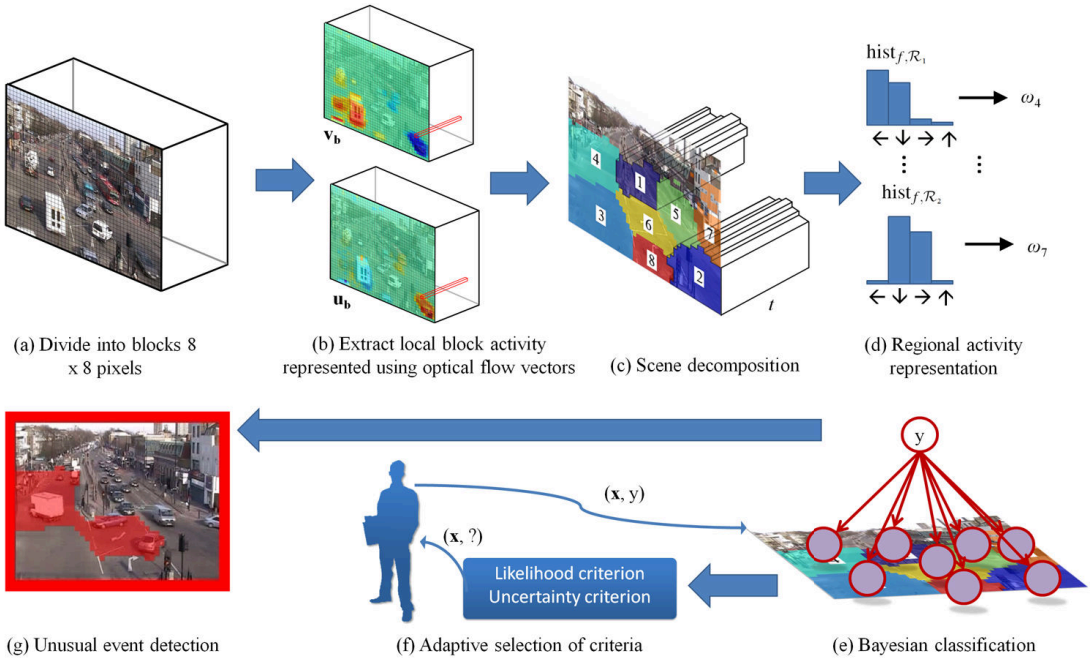


Figure 6.1: A diagram illustrating the proposed stream-based active unusual event detection method.

methods evaluated in this study.

The remainder of this chapter is organised as follows: Section 6.1 describes the activity representation employed in this study, followed by the explanation of the classification strategy in Section 6.2. The formulation of likelihood and uncertainty criteria are given in Section 6.3, whilst the adaptive selection strategy is explained in Section 6.4. Experiments are reported in Section 6.5 and summary is given in Section 6.6.

6.1 Activity Representation

The key component of the proposed approach is illustrated in Figure 6.1. The approach represents activity patterns as location-specific motion word (Figure 6.1(a-d)). A Bayesian classifier is employed to model the activity patterns (Figure 6.1(e)) and it is subsequently used for unusual event detection (Figure 6.1(g)). The activity model would query for human verification on any activity pattern that fulfil the query constraint defined by the likelihood and uncertainty criteria (Figure 6.1(f)).

In this work, activity patterns are represented using location-specific motion information over a temporal window without relying on object segmentation and tracking. This is achieved through the following steps:

1. Given an input video, a method similar to that of Chapter 3 (see Section 3.1.1) is employed to decompose a complex scene automatically into n regions, $\boldsymbol{\mathcal{R}} = \{\mathcal{R}_i | i = 1, \ldots, n\}$ according to the spatial-temporal distribution of motion patterns observed. In particular, the image space is first divided into equal-sized blocks with 8×8 pixels each (Figure 6.1(a)). Then, optical flow in each pair of consecutive frames is extracted using the Lucas-Kanade method [146]. Flow vectors are averaged within each block to obtain local block activity patterns, which are represented using the horizontal and vertical flow components, $\mathbf{u_b}$ and $\mathbf{v_b}$, where $\mathbf{b}$ denotes the two-dimensional coordinates of a block in the image space (Figure 6.1(b)). Both $\mathbf{u_b}$ and $\mathbf{v_b}$ are one-dimensional vectors computed over time (*i.e.* time series). Correlation distances are computed among local block activity patterns to construct an affinity matrix, which is then used as an input to a spectral clustering algorithm [250] for scene decomposition (Figure 6.1(c)). Note that despite using a similar scene decomposition technique, the activity representation proposed in this chapter is slightly different to that presented in previous chapters (see Sections 3.1.1 and 4.1.1). In particular, videos with higher frame rate are assumed. Consequently, instead of using foreground pixel-based representation, optical flow representation that offers richer information is utilised.

2. Motion direction of each moving pixel in each region is quantised into four directions and put into bins.

3. A histogram $\text{hist}_{f,\mathcal{R}_i}$ with a size of four bins is constructed for each region $\mathcal{R}_i$ in each frame f. The whole video sequence is uniformly divided into non-overlapping clips, each having 50 frames in length. Individual bins of a regional histogram within each clip t are then summed up as $\text{hist}_{t,\mathcal{R}_i} = \sum_{f \in \text{clip } t} \text{hist}_{f,\mathcal{R}_i}$, as illustrated in Figure 6.1(d).

4. The four direction bins are ranked in a descending order based on their values. The dominant motion direction is identified from the first few bins in the rank that account for a given fraction $P \in [0,1]$ of total bin values ($P = 0.8$ in this study). Non-dominant motion directions are then removed as they are likely to be caused by error in optical flow computation.

5. Finally, the histogram of region $\mathcal{R}_i$ is discretised to construct a codebook with $r = 16$ words, $\{\omega_j | j = 1, \ldots, 16\}$, representing the dominant motion directions of each region. For example, word ω_1 represents motionless region, word ω_2 means only direction bin 1 is

observed, and word ω_4 indicates both occurrence of direction bins 1 and 2, etc. as depicted in Figure 6.1(d).

Consequently, the ith region of the t-th clip is represented as a variable $x_{i,t}$ of 16 possible discrete values according to its word label, and the clip is denoted as $\mathbf{x}_t = (x_{1,t}, \ldots, x_{n,t})$.

6.2 Bayesian Classification

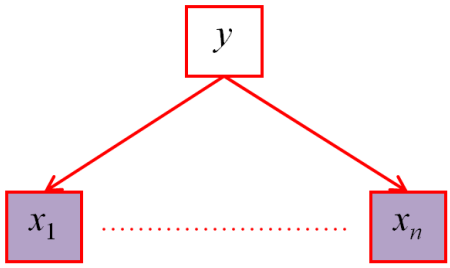


Figure 6.2: A graphical representation of the naïve Bayesian classifier. Each variable in the observed vector $\mathbf{x} = (x_1, \ldots, x_n)$ are conditioned on the class variable y.

Given a n-dimensional observed vector $\mathbf{x} = (x_1, \ldots, x_n)$, this study wishes to assign the vector into one of the K classes, where a class variable is represented by $y = k \in \{1, \ldots, K\}$. This classification task is approached using the Bayesian classification. To facilitate efficient incremental learning, a naïve Bayesian classifier (Figure 6.2) is employed with Bayesian learning as the learning scheme. Conditional independence is assumed among the distributions of input attributes $x_1, \ldots, x_n$ given the class label. It is noted that the assumption of conditional independence is strong and may not be accurate in many cases since variables/features are usually dependent. Nevertheless, even if the assumption is not precisely satisfied, the model may still return satisfactory performance as shown by a number of studies [95, 189]. Discussion on how to relax this assumption for application in multi-camera context is provided in Section 7.2.

The naïve Bayesian classifier is quantified by a parameter set specifying the conditional probability distributions (CPDs). Separate multinomial distribution $p(x_i|y)$ on each x_i for each class label is assumed. Consequently, $\boldsymbol{\theta}_{x_i|y} = \left\{\theta_{x_i=j|y}\right\}$ is used to represent parameters for the multinomial distribution $p(x_i|y)$, where $0 \leq \theta_{x_i=j|y} \leq 1$ and $\sum_{j=1}^{r} \theta_{x_i=j|y} = 1$. Recall that $r = 16$ since each variable x_i can take 16 states, each of which represents the dominant motion directions of each region.

Given the multinomial CPDs, the conditional probability $p(\mathbf{x}|y = k)$ for an observed vector given class $y = k$ is given as $p(\mathbf{x}|y = k) = \prod_{i=1}^{n} p(x_i|y = k)$. Given $p(\mathbf{x}|y)$ and $p(y)$, posterior

conditional distribution $p(y|\mathbf{x})$ can be computed via Bayes rule. A class y^* that best explains $\mathbf{x}$ can then be obtained as follows:

$$y^* = \underset{k\in\{1,\ldots,K\}}{\operatorname{argmax}}\ p(y=k|\mathbf{x}) = \underset{k\in\{1,\ldots,K\}}{\operatorname{argmax}}\ p(y=k)p(\mathbf{x}|y=k). \tag{6.1}$$

Tractable incremental learning is required for stream-based active learning. Since observation at each time step is complete, that is, each $(\mathbf{x}_t, y_t)$ assigns values to all the variables of interest, conjugate prior can be employed to facilitate efficient Bayesian learning (see Section 4.1.4). The conjugate prior of a multinomial distribution with parameters $\boldsymbol{\theta}_{x_i|y}$ is the Dirichlet distribution [21], which is given as:

$$\mathrm{Dir}(\boldsymbol{\theta}_{x_i|y} \mid \boldsymbol{\alpha}_{x_i|y}) \propto \prod_{j=1}^{r} [\theta_{x_i=j|y}]^{\alpha_{x_i=j|y}-1}, \tag{6.2}$$

where $\boldsymbol{\alpha}_{x_i|y}$ denotes $\left(\alpha_{x_i=1|y}, \ldots, \alpha_{x_i=r|y}\right)$ and $\alpha_{x_i=j|y} \in \mathbb{R}^+$ are parameters of the Dirichlet distribution. The classifier is formulated so that it can expand itself automatically whenever a new class is discovered during the active learning process. This is achieved by expanding the class variable y to $K+1$ states and augmenting the CPD of each x_i with $\boldsymbol{\theta}_{x_i|y=K+1}$.

6.3 Query Criteria

In a stream-based setting, the query decision is typically determined by a query score p^{query} derived from a query criterion $\mathcal{Q}$. The query score will be compared against a threshold Th. Specifically, if $p^{\text{query}} \geq \text{Th}$, query is made; otherwise $\mathbf{x}_t$ is discarded. This study proposes to employ two widely used criteria with clear complementary nature, namely likelihood criterion and uncertainty criterion for joint unknown event discovery and classification boundary refinement. Different methods are formulated to compute the respective query scores based on these criteria.

6.3.1 Likelihood Criterion

To use this criterion, a point is selected by comparing its likelihood against current distribution modelled by the classifier. In particular, given a sample $\mathbf{x}$, a class y^* that best explains the sample according to Equation (6.1) is first identified. Secondly, for each feature node, the normalised

probability score of x_i given y^* is computed:

$$\hat{p}(x_i|y^*) = \frac{p(x_i|y^*) - \mathrm{E}\left[p(x_i = j|y^*)\right]}{\sqrt{\mathrm{E}\left[p(x_i = j|y^*) - \mathrm{E}\left[p(x_i = j|y^*)\right]\right]}}.$$

The normalised probability score $\hat{p}(x_i|y^*)$ is bounded to ensure $-0.5 \leq \hat{p}(x_i|y^*) \leq 0.5$. Finally, the likelihood score at time step t is calculated as:

$$p_t^l = 1 - \left(\frac{1}{2} + \frac{1}{n}\sum_{i=1}^{n} \hat{p}(x_i|y^*)\right). \tag{6.3}$$

The likelihood score lies within [0,1]. If p_t^l of a sample is closer to 1, it is more likely to be queried.

6.3.2 Uncertainty Criterion

The proposed uncertainty criterion is re-formulated from the existing QBC algorithm [9, 152], with additional consideration on conflicting classes for yielding a balanced sample selection. There are two main steps that have to be taken for computing an uncertainty score using the QBC.

(i) *Generating a committee* - Given a classifier $\mathcal{C}_t$ and training data $\mathcal{S}_t$, M committee members corresponding to hypotheses $\mathbf{h} = \{h_i\}$ of the hypotheses space $\mathcal{H}_t$ are generated, where each hypothesis is consistent with the training data seen so far [199], *i.e.*

$$h_i \in \mathcal{H}_t | \forall(\mathbf{x}, y) \in \mathcal{S}_t, h_i(\mathbf{x}) = y. \tag{6.4}$$

In a naïve Bayes classifier with multinomial CPDs, this can be done by sampling new parameters from the posterior Dirichlet distribution of classifier [9, 152]. It has been proven (Chapter XI, Theorem 4.1 in [60], see Appendix B for details) that a random vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_r)$ from the Dirichlet distribution with parameter $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_r)$ (see Equation (6.2)) can be generated from the Gamma distributions. Consequently, in this work r random samples $\varepsilon_{x_i=1|y}, \ldots, \varepsilon_{x_i=r|y}$ are first generated from the Gamma distributions each with density

$$\mathrm{Gam}\left(\varepsilon_{x_i=j|y} \mid \alpha_{x_i=j|y}\right) = \frac{1}{\Gamma\left(\alpha_{x_i=j|y}\right)} \mathrm{e}^{-\varepsilon_{x_i=j|y}} \left(\varepsilon_{x_i=j|y}\right)^{\alpha_{x_i=j|y}-1}, \tag{6.5}$$

where $\Gamma(\cdot)$ denotes the Gamma function and $\alpha_{x_i=j|y}$ plays a role as a shape parameter in the

Gamma distribution. The parameter of a committee member is then estimated as:

$$\hat{\theta}_{x_i=j|y} = \frac{\varepsilon_{x_i=j|y} + \lambda}{\sum_j \left(\varepsilon_{x_i=j|y} + \lambda\right)}. \tag{6.6}$$

where λ is a weight added to compensate data sparseness, *i.e.* to prevent zero probabilities for infrequently occurring values $x_i = j$.

(ii) *Measure of disagreement* - For measuring the level of disagreement among committee members, two main approaches have been proposed. The first measure is *vote entropy* [9]:

$$p_t^{\mathrm{VE}} = -\sum_{k=1}^{K} \frac{V(y=k)}{M} \log \frac{V(y=k)}{M}, \tag{6.7}$$

where $V(y=k)$ represents the number of votes that a label receives from the committee members' predictions. The higher the entropy p_t^{VE}, the more uncertain a point is. Another disagreement measure is *average KL divergence* or *KL divergence to the mean* [152]:

$$p_t^{\mathrm{KL}} = \frac{1}{M} \sum_{i=1}^{M} \sum_{k=1}^{K} p_{\Theta_i}(y=k|\mathbf{x}_t) \log \frac{p_{\Theta_i}(y=k|\mathbf{x}_t)}{\overline{p}(y=k|\mathbf{x}_t)}, \tag{6.8}$$

where Θ_i represents a set of parameters of a particular model in the committee, and $\overline{p}(y=k|\mathbf{x}_t)$ denotes the mean of all distributions, *i.e.* $\overline{p}(y=k|\mathbf{x}_t) = \frac{1}{M}\sum_{i=1}^{M} p_{\Theta_i}(y=k|\mathbf{x}_t)$, as the "consensus" probability that k is the correct label [197]. As can be seen from Equations (6.7) and (6.8), it is clear that the two aforementioned methods are not formulated to return the corresponding classes that cause the most disagreement.

To overcome this limitation, a new uncertainty score is formulated as follows: first, a class disagreement score is computed over all possible class labels:

$$s_{y=k,t} = \left\{ \max_{h_i \in \mathcal{H}_t, h_j \in \mathcal{H}_t} \left[p_i(y=k|\mathbf{x}_t) - p_j(y=k|\mathbf{x}_t) \right] \right\}, \tag{6.9}$$

where $i \neq j$. Consequently, the top two classes that return the highest $s_{y=k,t}$ are identified as c_1 and c_2. The final uncertainty score is computed as:

$$p_t^u = \frac{1}{2} \cdot \gamma_u \cdot \left[s_{y=c_1,t} + s_{y=c_2,t} \right], \tag{6.10}$$

where γ_u is the prior introduced to favour the learning of classification boundary for unusual

classes. Specifically, γ_u is set to a low value if c_1 and c_2 are both normal event class, and a high value if any one of c_1 and c_2 is unusual event class. If p_t^u of a sample is closer to 1, it is more likely to be queried.

6.4 Adaptive Selection of Multiple Query Criteria

As explained in Section 1.3.2, adaptive selection of multiple criteria is useful for joint unknown event discovery and classification boundary refinement. In particular, different criteria can be more suitable for different datasets as well as different learning stages. Since one usually does not know the right choice *a priori*, selecting different criteria adaptively has the potential to provide a more reliable and even more optimal solution than using any single criterion alone.

To this end, an adaptive selection approach is formulated to select different query criteria for stream-based active learning. The adaptive selection scheme is based on the assumption that an informative sample would bring more influence to a model in terms of parameter change. If a criterion keeps selecting informative sample, its tendency to be selected will also be increased. Specifically, given multiple query criteria $\boldsymbol{\mathcal{Q}} \in \{\mathcal{Q}_1, \ldots, \mathcal{Q}_a, \ldots, \mathcal{Q}_{\mathcal{A}}\}$, a weight $w_{a,t}$ is assigned to each query criterion $\mathcal{Q}_a$ at time step t. A criterion is chosen by sampling from a multinomial distribution given as:

$$a \sim \text{Mult}(\mathbf{w}_t), \tag{6.11}$$

where $\mathbf{w}_t \in \{w_{1,t}, \ldots, w_{a,t}, \ldots, w_{\mathcal{A},t}\}$. Intuitively, a criterion with higher weight is more likely to be chosen.

The weight $w_{a,t}$ is updated in every time step t as follows:

$$w_{a,t} = \beta w_{a,t-1} + (1-\beta)\frac{\overline{\mathcal{KL}}_a(\Theta \parallel \tilde{\Theta})}{\sum_{a=1}^{\mathcal{A}} \overline{\mathcal{KL}}_a(\Theta \parallel \tilde{\Theta})}, \tag{6.12}$$

where β denotes an update coefficient that controls the updating rate of weights, whilst Θ and $\tilde{\Theta}$ represent sets of parameters of a naïve Bayes classifier $\mathcal{C}_t$ and an updated classifier $\mathcal{C}_{t+1}$ trained using $\mathcal{S}_t \bigcup \{(\mathbf{x}_t, y_t)\}$.

From Equation (6.12), one can observe that the change of weight is guided by the symmetric KL divergence $\overline{\mathcal{KL}}_a(\Theta \parallel \tilde{\Theta})$, which is yielded by a query criterion $\mathcal{Q}_a$ when it last triggered a query. In particular, $\overline{\mathcal{KL}}_a(\Theta \parallel \tilde{\Theta})$ measures the difference between two parameter distributions $p_\Theta(\mathbf{x})$ and $p_{\tilde{\Theta}}(\mathbf{x})$ modelled by the naïve Bayes classifier *before and after* it is updated using

a newly queried sample. Intuitively, a criterion is preferred, therefore being assigned higher weight if it asks for samples that give greater impact to the existing distribution modelled by the classifier.

A symmetric KL divergence $\overline{\mathcal{KL}}(\Theta \parallel \tilde{\Theta})$ is computed as follows:

$$\overline{\mathcal{KL}}(\Theta \parallel \tilde{\Theta}) = \frac{1}{2}.\left[\mathcal{KL}(\Theta \parallel \tilde{\Theta}) + \mathcal{KL}(\tilde{\Theta} \parallel \Theta)\right]. \tag{6.13}$$

with $\mathcal{KL}(\Theta \parallel \tilde{\Theta})$ being defined as:

$$\mathcal{KL}(\Theta \parallel \tilde{\Theta}) = \sum_{\mathbf{x}} p_{\Theta}(\mathbf{x}) \ln \frac{p_{\Theta}(\mathbf{x})}{p_{\tilde{\Theta}}(\mathbf{x})}, \tag{6.14}$$

which can be decomposed as follows [219]:

$$\begin{aligned} \mathcal{KL}(\Theta \parallel \tilde{\Theta}) &= \sum_{i=1}^{n} \mathcal{KL}\left(p_{\Theta}(x_i|y) \parallel p_{\tilde{\Theta}}(x_i|y)\right) \\ &= \sum_{i=1}^{n} \sum_{k=1}^{K} p(y=k) \mathcal{KL}\left(p_{\Theta}(x_i|y=k) \parallel p_{\tilde{\Theta}}(x_i|y=k)\right). \end{aligned} \tag{6.15}$$

Note that a symmetric KL divergence, $\overline{\mathcal{KL}}(\Theta \parallel \tilde{\Theta})$ is employed here since $\mathcal{KL}(\Theta \parallel \tilde{\Theta}) \not\equiv \mathcal{KL}(\tilde{\Theta} \parallel \Theta)$ in general.

Algorithm 4 summaries the proposed approach. Similar to most stream-based active learning approaches [55,100], the computational cost of the proposed method is low thus suitable for real-time processing. Specifically, lines 2-14 of Algorithm 4 take an average time of 0.026 seconds with Matlab implementation on a single-core 2.8 GHz machine.

Algorithm 4: Stream-based active unusual event detection.

Input: Data stream $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_t, \ldots)$, an initial classifier $\mathcal{C}_0$ trained with a small set of labelled samples from known classes

Output: A set of labelled samples $\mathcal{S}$ and a classifier $\mathcal{C}$ trained with $\mathcal{S}$

```
Set S_0 = a small set of labelled samples from known classes ;
for t from 1,2,... until the data stream runs out do
    Receive x_t;
    Compute p_t^l (Equation (6.3)) ;
    Compute p_t^u (Equation (6.10)) ;
    Select query criterion by sampling a ~ Mult(w), assign p_t^query based on the selected
    criterion ;
    if p_t^query ≥ Th then
        Request y_t and set S_t = S_{t-1} ∪ {(x_t, y_t)} ;
        Obtain classifier C_{t+1} by updating classifier C_t with {(x_t, y_t)} ;
        Update query criteria weights w (Equation (6.12)) ;
    else
        S_t = S_{t-1};
    end
end
Unusual event is detected if p(y = unusual|x) is higher than Th_unusual;
```

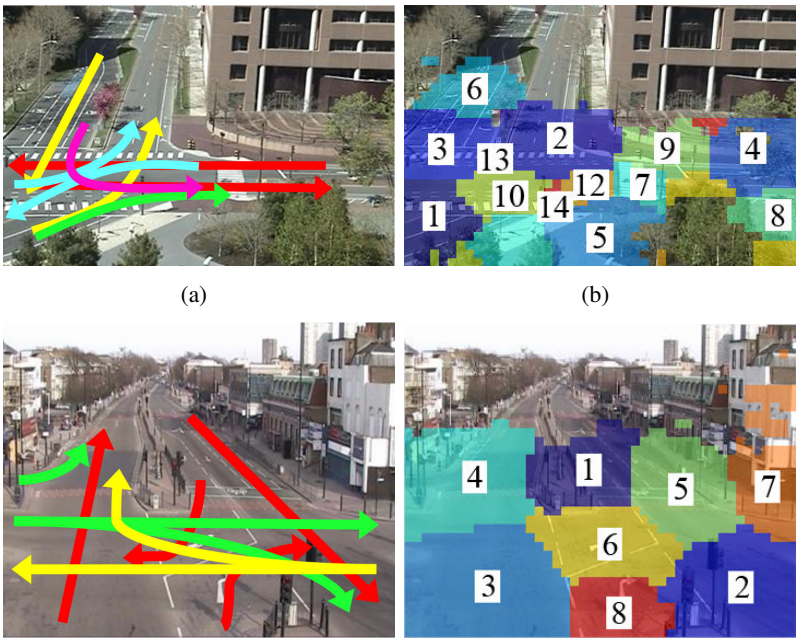
6.5 Experiments

6.5.1 Datasets and Settings

Two public video datasets[1] captured at busy public scenes were employed in experiments of this work.

MIT traffic dataset [231] - This dataset with an approximate length of 1.5 hours (168822 frames), is recorded at 30 fps and scaled to a frame size of 360×240. The traffic is controlled with traffic lights and dominated by five different traffic flows (Figure 6.3(a)). The scene decomposition result is given in Figure 6.3(b), showing the fourteen regions discovered.

[1]Processed data with ground truth are available for download at: `http://www.eecs.qmul.ac.uk/~ccloy/files/accv_2010_dataset.zip`.



(a) (b)

(c) (d)

Figure 6.3: Dominant traffic flows observed in MIT traffic dataset (a) and QMUL junction dataset (c) are treated as normal event classes. Different classes of traffic flow are coded with different colours (see Table 6.1). The scene decomposition results of both datasets according to the spatial distribution of activity patterns are shown in (b) and (d), respectively.

QMUL junction dataset [104, 134, 141] - The length of the video is approximately 60 minutes (89999 frames) captured with 360×288 frame size at 25 fps. The traffic is regulated by traffic lights and dominated with three traffic flows as shown in Figure 6.3(c). The scene decomposition result is depicted in Figure 6.3(d), showing the eight regions discovered.

Both datasets feature complex activities exhibited by multiple objects. In particular, behaviours and the correlations among vehicles are determined by not only the traffic light cycles, but also traffic volume and driving habits of drivers. For instance, vehicles waiting in region 6 of QMUL junction dataset can perform horizontal turning whenever there is a gap in vertical flow. This type of activity is more frequent in MIT traffic dataset, in which vehicles are allowed to do turning between gaps of traffic flows. As a consequence, the traffic phases of MIT traffic dataset are less distinctive visually and become harder to model compared to QMUL junction dataset.

Ground truth - The videos were segmented into non-overlapping clips of 50 frames long each, resulting 1800 clips and 3376 clips for QMUL junction dataset and MIT traffic dataset respectively. Each clip was manually labelled into different event classes as listed in Table 6.1. The ground truth is used as feedback returned to a classifier when it requests for labels during active

learning process[2]. It is also employed for comparison during testing phase.

Class	No. of clips (% from total)	Description
		MIT Traffic Dataset
1	874 (25.89)	Horizontal traffic flow (red arrows in Figure 6.3(a))
2	1249 (37.00)	Vertical traffic flow (yellow arrows in Figure 6.3(a))
3	376 (11.14)	Right-turn from zone 1 toward zone 4 (green arrow in Figure 6.3(a))
4	185 (5.48)	Left-turn from zone 3 toward zone 4 (magenta arrow in Figure 6.3(a))
5	517 (15.31)	Turning from left-exit toward zone 2, turning from zone 9 to zone 1 (cyan arrows in Figure 6.3(a))
6	75 (2.22)	[Unusual] Left-turn from zone 1 to left-exit
7	79 (2.34)	[Unusual] Turning right from zone 7 to zone 2
8	21 (0.62)	[Unusual] U-turn at zone 7
		QMUL Junction Dataset
1	1078 (59.89)	Vertical traffic flow (red arrows in Figure 6.3(c))
2	323 (17.94)	Rightward traffic flow (yellow arrows in Figure 6.3(c))
3	355 (19.72)	Leftward traffic flow (green arrows in Figure 6.3(c))
4	29 (1.61)	[Unusual] Illegal u-turns from zone 1 to zone 4 via zone 6
5	3 (0.17)	[Unusual] Emergency vehicles entered the junction using an improper lane of traffic
6	12 (0.67)	[Unusual] Traffic interruptions by fire engines

Table 6.1: Ground truth of MIT traffic dataset and QMUL junction dataset.

Settings - The clips (see Table 6.1) were randomly partitioned into training/test sets with equal size. Different partitions were used in different runs in the experiments. In this study, all experimental results were averaged over 30 runs.

This study followed similar experimental setting reported by Ho and Wechsler [100]. In particular, if a model did not request for any labels after observing a sufficiently large number of samples (100 in this study), the query threshold Th (preset to 0.5) was reduced to Th$'$ where Th$'$ was the largest p^{query} computed since the last query. A budget constraint, *i.e.* the number of samples a classifier can request on the data stream was specified as 250. There are three free parameters in the proposed active learning approach, namely λ (Equation (6.6)), uncertainty weights γ_u (Equation (6.10)), and update coefficient β (Equation (6.12)). Coarse parameter values were used without optimisation: $\lambda = 0.1$ for a weak prior, $\gamma_u = 0.9$ among normal classes, $\gamma_u = 10$ among normal-unusual classes, and $\beta = 0.9$ for slow adaptation rate. The number of

[2]In reality, these labels are assumed to be provided by human operators.

committee members for all QBC approaches was set to three in this study[3]. Initially, the classifier was given a sample from a random normal event class to start the learning process. In the QBC approaches, two random samples from different classes were needed.

6.5.2 Active Learning vs. Unsupervised Learning

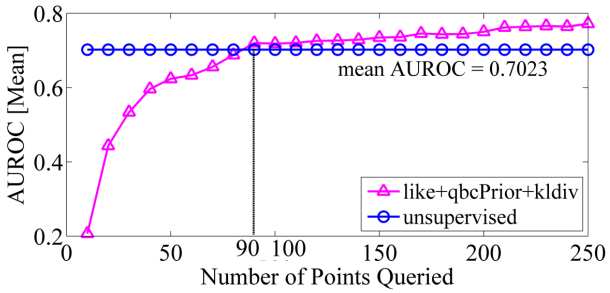


Figure 6.4: A comparison between the proposed active learning approach and an unsupervised learning approach. The results were averaged over 30 runs.

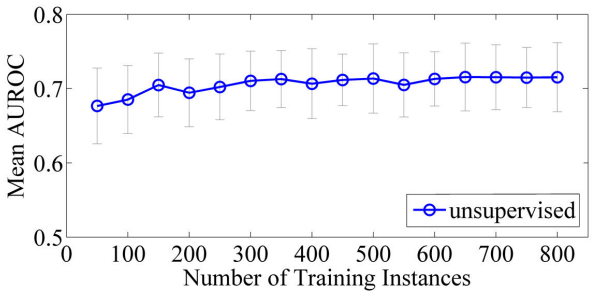
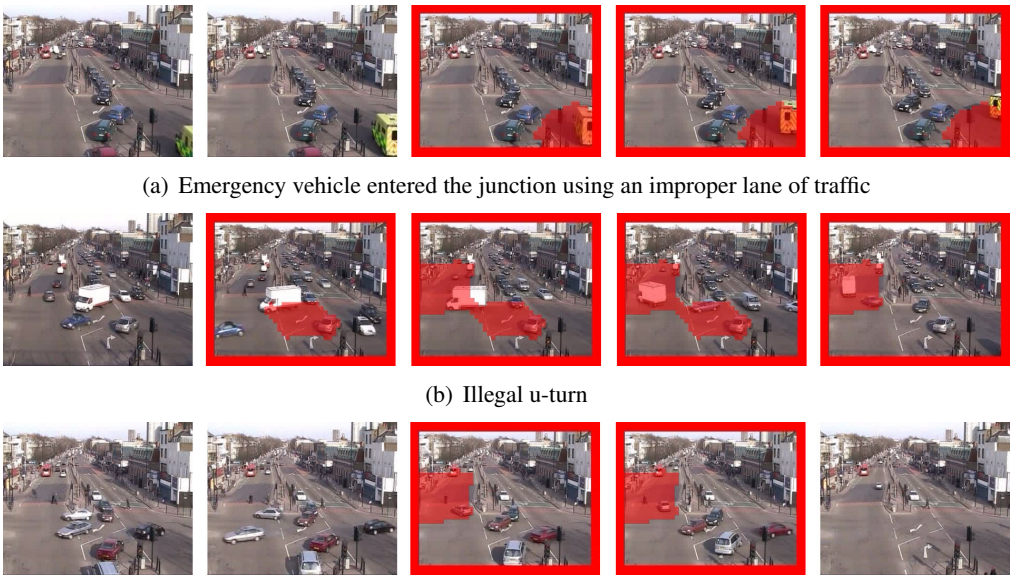


Figure 6.5: Unusual event detection performance of an unsupervised learning approach along with different numbers of training instances. The results were averaged over 30 runs.

The proposed method was first compared with unsupervised learning approach. To build an unsupervised model, a random set of 250 normal samples was first selected. The groups of these samples (*i.e.* the normal classes) were determined through the GMM clustering with automatic model selection based on Bayesian Information Criterion score [244]. The samples together with the predicted cluster labels were then employed to train a model described in Section 6.2. Note that the unsupervised learning strategy employed here is essentially similar in spirit to that reported in existing studies [104, 131, 155, 231]. For a fair comparison, same feature representation and model were used for both active learning and unsupervised learning strategies. As can be seen from Figure 6.4, the proposed method outperformed the fully unsupervised method given

[3]It is reported by McCallum *et al.* [152] that a committee size of three is sufficient and varying the size has little effect.



(a) Emergency vehicle entered the junction using an improper lane of traffic

(b) Illegal u-turn

(c) Illegal u-turn

Figure 6.6: Examples of true detection made by the proposed active learning method on QMUL junction dataset. Unsupervised method detected both clip (a) and clip (b) but missed the u-turn event in clip (c). The unusual regions are manually highlighted.

(a) (b) (c) (d)

Figure 6.7: Examples of false detection made by the unsupervised method on QMUL junction dataset. In contrast, the first three clips to the left were detected as normal using the proposed active learning method. The last clip that featured quiet scene was falsely detected as being unusual by both the unsupervised method and the active learning method.

as little as 90 samples annotated through active learning. Even if the number of unlabelled samples was increased to 800, the performance of the unsupervised model was still poorer to that achieved using the proposed approach as depicted in Figure 6.5. The results suggest that without exploiting human feedback, it was difficult to learn a satisfactory decision boundary even if a large amount of pure visual samples were employed.

Some example clips of unusual event detected using the proposed method are shown in Figure 6.6. As can be seen, illegal u-turn that was missed by the unsupervised model (Figure 6.6(c)) was detected by the active learning model. Apart from achieving more robust detection of subtle events, the active learning model also resulted fewer false detections in comparison to the unsupervised model since human feedback was taken into account for arbitrating such false alarms. For instance, normal events depicted in Figure 6.7(a-c) were falsely classified as being unusual

using the unsupervised model. These misclassified events corresponded to visually ambiguous events, *e.g*. vehicles using a gap in the middle of vertical traffic flow to make right and left turns at the same time interval (Figure 6.7(b)); a car making a left turn during a gap in the vertical traffic flow (Figure 6.7(c)). On the contrary, such ambiguous events had been picked up and incorporated as part of normal event class during active learning process. These testing clips were thus correctly detected as being normal using the proposed approach. Nevertheless, some events such as a quiet junction as shown in Figure 6.7(d) was wrongly detected as an unusual event using the proposed approach. Such a false alarm could be alleviated if more similar examples were available during active learning.

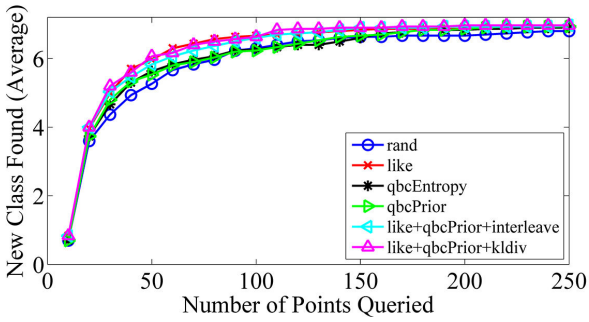
6.5.3 Active Learning vs. Random Sampling and other Active Learning Strategies

The following experiments compare the proposed approach with random sampling strategy and different active learning strategies. These strategies are summarised as follow:

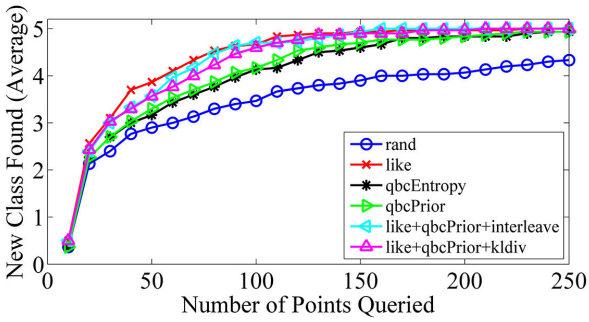
1. *rand* - random sampling strategy, *i.e*. samples are randomly chosen from the data stream
2. *like* - likelihood criterion as explained in Section 6.3.1
3. *qbcEntropy* - QBC approach with vote entropy measure [9]
4. *qbcPrior*[4] - the proposed QBC approach with a measure as described in Section 6.3.2
5. *like+qbcPrior+interleave* - combine both *like* and *qbcPrior* using interleave strategy, *i.e*. iterating different criteria during learning. This method is similar to the multi-criteria strategy proposed by Stokes *et al*. [209]
6. *like+qbcPrior+KLdiv* - combine both *like* and *qbcPrior* using the KL divergence-based strategy as described in Section 6.4

Different active learning strategies were evaluated according to: (1) how fast they can discover unknown classes (including normal and unusual event classes), and (2) how accurately the learned classifier can detect unusual events. The former case was measured based on the number of classes discovered vs. number of samples queried. The latter case was evaluated using AUROC computed in each active learning iteration against the number of queried samples. Receiver operating characteristic curve was obtained by varying $\mathrm{Th}_{\mathrm{unusual}}$.

[4]Matlab codes are available for download at: `http://www.eecs.qmul.ac.uk/~ccloy/files/qbcPrior.zip`.



(a) MIT traffic dataset



(b) QMUL junction dataset

Figure 6.8: Class discovery performance.

Discover unknown event classes - As can be seen from Figure 6.8, *like* showed the best performance in discovering unknown event classes in both datasets. The QBC approaches (*qbcEntropy* and *qbcPrior*) yielded slightly inferior result compared to *like* but still performed better than random sampling strategy. Specifically, with the introduction of prior (see Equation (6.10)) for dealing with the imbalanced data problem, the performance of the proposed *qbcPrior* is better to that of *qbcEntropy* (see Figure 6.8(b)). Both the proposed *like+qbcPrior+KLdiv* method and the alternative multi-criteria method *like+qbcPrior+interleave* showed comparable results and performed better than *qbcPrior* and *qbcEntropy* after combining with *like*.

To provide further insights on the performance difference between the *rand* method and the proposed *like+qbcPrior+KLdiv* method, queries raised by both approaches in the first 34 active learning rounds are given and compared in Figures 6.9 and 6.10, respectively. As can be observed, the amount of new event classes (both normal and unusual events) discovered by the *like+qbcPrior+KLdiv* method clearly outnumber the *rand* method. In particular, as shown in Figure 6.10, unusual events such as queries 17 and 32 (class 6 and 4 in QMUL junction dataset) were selected for labelling request based on uncertainty and likelihood criteria. This is in contrast to the *rand* method that selects queries without considering the influence and importance of a sample towards the learning process. A detailed examination on Figure 6.10 also reveals the

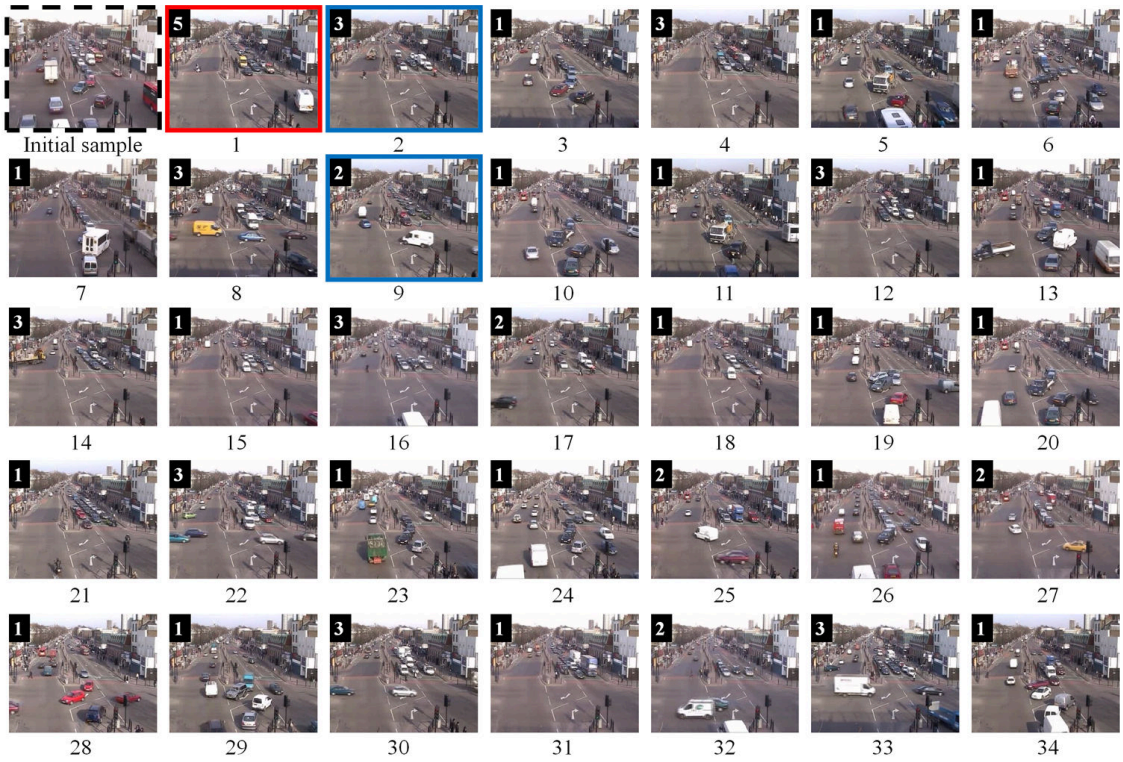


Figure 6.9: The figure shows the labelling requests (the key frame of each clip is depicted) that were made by the *rand* method in the first 34 active learning rounds on the QMUL junction dataset. The labels located at the bottom of each frame provide the order of the queries, whilst the labels on each frame's top-left corner show the class tag returned for respective request. The first frame with dashed border is an initial sample given to the model to initialise the active learning process. A query with blue border represents a new normal class discovered, whilst a query with red border denotes a new unusual class found during the learning process.

capability of the proposed *like+qbcPrior+KLdiv* method in complementing the current model with unexplored/unknown regions of existing classes. For instance, queries 21 to 25 were all belong to the vertical traffic flow class, but each encompassed unique activity patterns that were previously unknown to the current model.

Unusual event detection - Figure 6.11 shows the performance of different active learning strategies in detecting unusual events, measured as averaged AUROC over 30 runs. Overall, it can be seen that the detection performance of all active learning methods monotonically increase as more data is queried. Importantly, all methods significantly outperformed random sampling. In particular, it is observed that by incorporating prior constraint into uncertainty criterion (*qbcPrior*) yielded slightly better result compared to method without prior constraint (*qbcEntropy*). This is because that, without the prior constraint, *qbcEntropy* wasted some effort in refining boundary between normal classes, whilst *qbcEntropy* focussed on the uncertainty regions surrounding the unusual event classes.

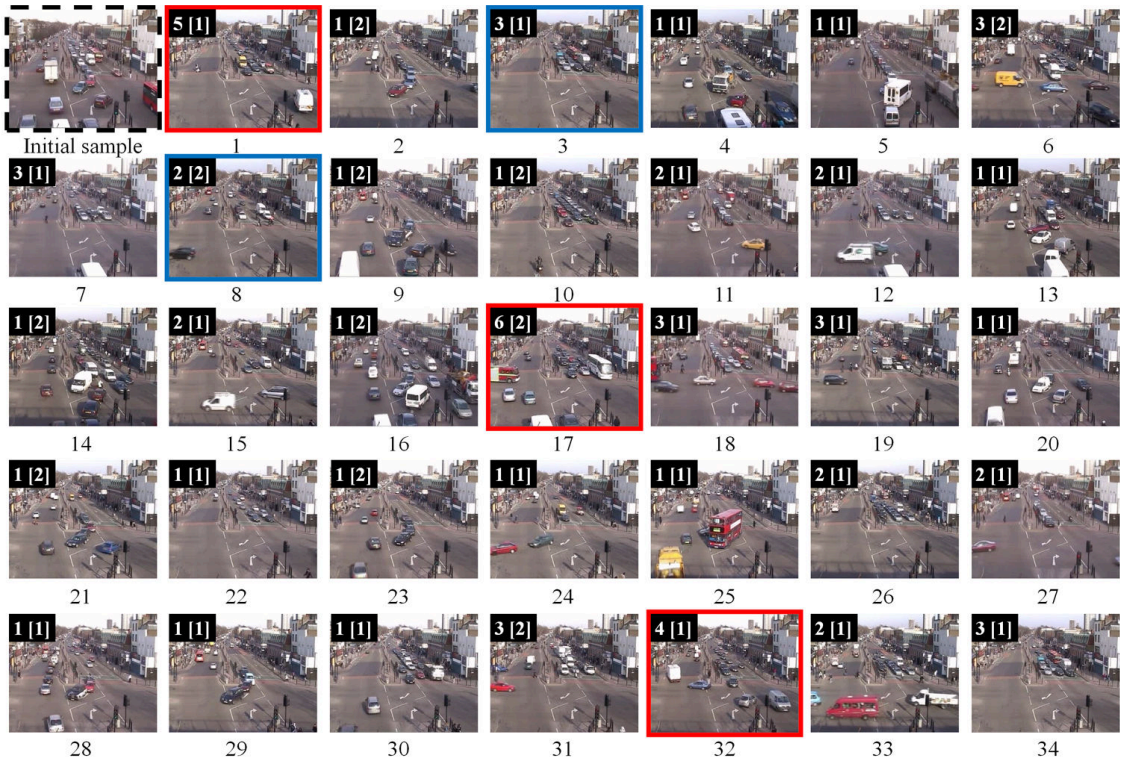
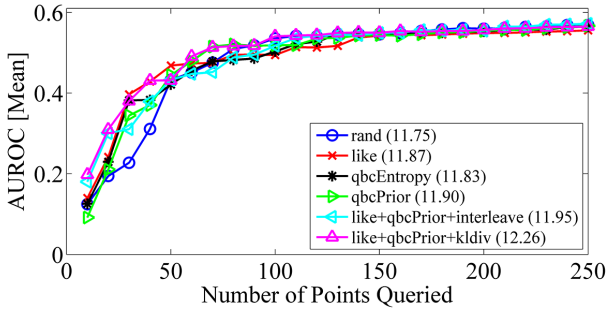


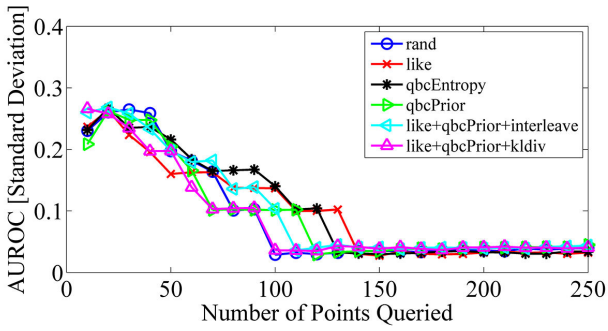
Figure 6.10: The figure shows the labelling requests (the key frame of each clip is depicted) that were made by the *like+qbcPrior+KLdiv* method in the first 34 active learning rounds on the QMUL junction dataset. Similar figure labelling scheme used in Figure 6.9 is employed here. An additional number inside a bracket located at the frame's top-left corner denotes the criterion that was used to pick the query (1 = likelihood criterion, 2 = uncertainty criterion). Note that *like+qbcPrior+KLdiv* selected some interesting and ambiguous clips for labelling request. In contrast, *rand* (Figure 6.9) select clips in a random manner without considering the influence and importance of these clips toward the learning process.

In both datasets, as can be seen from Figure 6.11(a) and 6.11(c), *like+qbcPrior+KLdiv* yields the best performance. Adaptive selection of multiple criteria leads to reliable and good performance with the proposed method *like+qbcPrior+KLdiv* outperforming the alternative approach *like+qbcPrior+interleave*. The reliability of the multi-criteria methods is also reflected by the smaller variance across multiple trials shown in Figure 6.11(b) and 6.11(d).

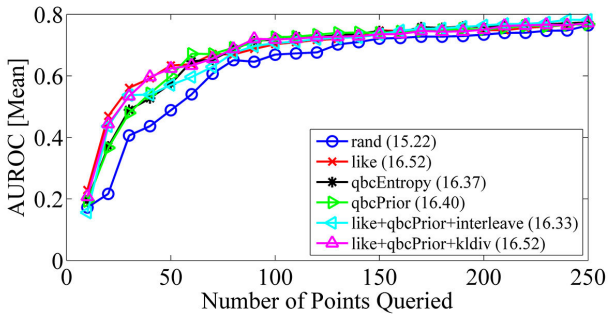
Adaptive selection of criteria - In contrast to iterative strategy reported by Stokes *et al*. [209], the proposed strategy assigns weights adaptively to different criteria at different stages of active learning based on the KL divergence of a model (see Section 6.4). For the MIT traffic dataset (Figure 6.12(a)), the likelihood criterion was more frequently selected than the uncertainty criterion before the number of queried samples reached 150. This observation suggests that when the visual distinctiveness between event classes were less obvious (see Section 6.5.1), the proposed method was capable of selecting the right criterion and avoid uncertainty criterion that may keep



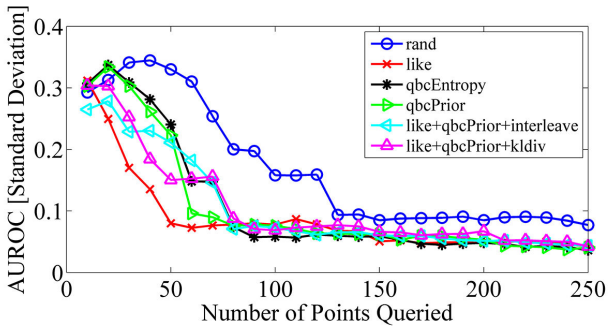
(a) MIT traffic dataset



(b) MIT traffic dataset

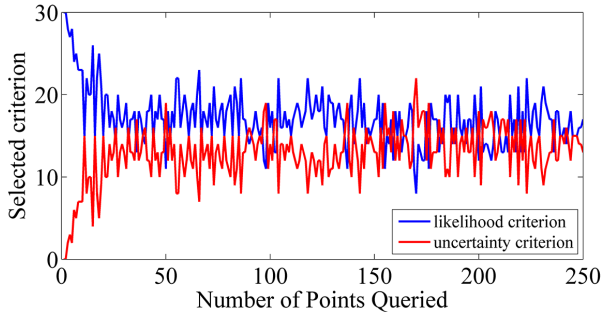


(c) QMUL junction dataset

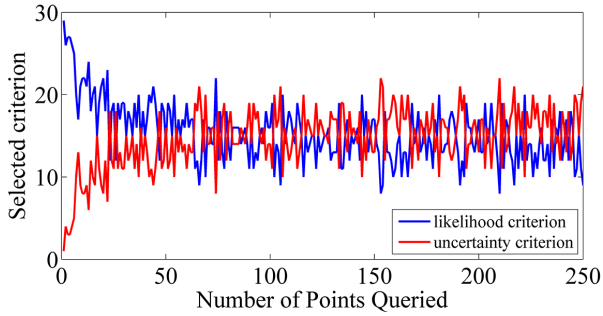


(d) QMUL junction dataset

Figure 6.11: Unusual events detection performance. Numbers shown in the brackets within graph legend are area under the mean AUROC of different approaches.



(a) MIT traffic dataset



(b) QMUL junction dataset

Figure 6.12: Selected criterion over 30 runs.

querying uncertain points located at highly overlapped area of class boundary, which are less useful for improving the detection performance. On the other hand, in the QMUL junction dataset (Figure 6.12(b)), likelihood criterion dominated at the beginning, since it discovered unknown events that caused greater change in parameter values to the classifier compared to uncertainty criterion. The model eventually switched from likelihood criterion to uncertainty criterion (after 80 samples were queried) to refine the classification boundary when the exploratory learning was no longer fruitful (see Figure 6.8(b), approximately 90% of total event classes were discovered after 80 samples).

6.6 Summary

This chapter has presented an investigation into the use of active learning to exploit human feedback for on-line unusual event detection. The premise behind the active learning approach is to select ambiguous points, which are either located close to the normal/unusual boundary or having low density in the current model, for validation and refutation by human operators.

Experimental results have demonstrated that the proposed approach is capable of yielding more robust and accurate detection of subtle unusual events in public space, as compared to

a random sampling strategy and conventional unsupervised learning strategies, by exploiting a small cost of human supervision through active learning. In general, the proposed stream-based multi-criteria approach is shown to be capable of balancing different query criteria for joint unknown event discovery and decision boundary refinement. It therefore results in a more reliable detection performance than using single criterion alone, and it outperforms an existing multi-criteria strategy [209] applied in a stream-based manner. In addition, by introducing a prior to deal with imbalanced data, the re-formulated QBC criterion improves the performance. However, it is still inconclusive that the adaptive selection scheme could always select the best suited criterion in all cases. For instance, as can seen from Figure 6.8(b), the class discovery performance of *like+qbcPrior+KLdiv* was inferior to that obtained by using *like* criterion alone, especially at the beginning of the active learning cycle when only a limited number of points were queried. The result implies that the selection scheme failed to select the *like* criterion in the test. More experiments on different dataset are needed to evaluate the capability of the proposed adaptive selection approach.

Chapter 7

Conclusion and Future Work

This thesis has set out to explore the possibility of using visual context and human feedback for activity understanding and unusual event detection in surveillance videos. In particular, the thesis is geared towards solving two problems: (1) activity analysis in public scenes monitored by multiple disjoint camera views and (2) detection of unusual events in crowded public scenes.

These problems are nontrivial due to inherent visual ambiguities and uncertainties owing to inter-camera visual variations, unknown time gaps between camera views, low-quality videos captured in crowded scene, inevitable sensory noise, as well as rarity and unpredictability of unusual events. As concluded in Chapter 2, the available literature strongly suggests that there is still a considerable scope for improving activity analysis on public scene surveillance videos by (1) learning multi-camera visual context and (2) incorporating human feedback into activity modelling.

7.1 Multi-Camera Activity Analysis with Visual Context Learning

This thesis has presented an alternative approach to learn the multi-camera visual context without performing either feature matching or object tracking. Specifically, a new approach has been proposed in Chapter 3 for learning pairwise correlations among partial observations of activities observed from multiple disjoint cameras separated with unknown and arbitrary time gaps. Subsequently, Chapter 4 has described an approach for modelling time delayed dependencies between distributed local activities by formulating a Time Delayed Probabilistic Graphical Model (TD-PGM), based on which a framework for global anomaly detection in multiple disjoint cameras

is developed. The time delayed dependencies among activities across camera views are globally optimised using a novel two-stage structure learning method. In order to adapt to unavoidable visual context changes in unconstrained environments, Chapter 5 further extends the two-stage structure learning method to permit time and memory efficient incremental learning on both parameters and structure of the TD-PGM.

The proposed approaches have shown superior performance as compared to existing techniques in various activity analysis tasks through multi-camera visual context learning. For instance, in Chapter 3, pairwise time delayed correlations were utilised as contextual information in person re-identification task to resolve ambiguities that arise due to similar visual features presented by multiple objects and feature variations caused by different poses, camera viewing angles and illumination changes. In activity-based temporal segmentation task, by exploiting the pairwise time delayed correlations and utilising the visual evidence collected from different views, the proposed multi-view activity modelling technique was shown to be more robust to noise and visual ambiguities than modelling activities separately within individual camera views. In Chapters 4 and 5, time delayed dependencies between regional activity patterns were deployed as visual context for global unusual event detection. Superior detection performance on subtle unusual events was achieved despite ambiguities and uncertainties caused by severe inter-object occlusions and temporal discontinuity owing to unknown inter-camera gaps.

Let us now discuss the areas that can be improved on. With regard to multi-camera dependency or correlation learning, there are several possible extensions:

1. *Spurious dependency* - Unexpected high dependencies were found between some regions even though their regional activities were not dependent on each other, *e.g.* $72 \rightarrow 20$ (see Figure 4.3). These unexpected dependencies may be caused by noise or constant crowdedness in two region pairs, which lead to spurious peaks that do not necessarily correspond to the true time delayed dependencies. A possible way to filter out these false dependencies is by analysing the shape of the xCCA or TDMI functions. The rational of carrying out the shape analysis is that a pair of connected regions would typically produce a function with a bell-shaped curve, whilst a function for two independent regions often exhibits a more random shape with multiple peaks.

2. *Multi-mode time delay and correlation* - It is assumed in the proposed approach that there is only one popular delay time between two regions. In most cases, this assumption is

valid because objects with different speeds (such as cars and pedestrians) appear in different regions, and their activities will thus be separated into different regions using the scene decomposition method. However, there are still cases where objects with different speeds and directions appear in the same location, *e.g*. middle of a traffic intersection. This work did not consider modelling multiple delay modes because the features employed do not capture motion speed and direction due to videos with low temporal and spatial resolution. Nonetheless, if object speed and direction can be measured given videos with higher frame rate, it is possible to extend the proposed approach to capture multiple delay and correlation modes. Specifically, one could decompose different directions into different bins (*e.g*. direction 001 = 0° - 15°, direction 010 = 15° - 30°, *etc*.). For each decomposed direction, xCCA or TDMI analysis can then be performed to model the correlation or dependency surface of different speeds and time delays. Alternatively, one could introduce a hidden node between two regions to learn the multi-mode time delay distribution in a probabilistic manner.

3. *More features* - Whilst this work has demonstrated the effectiveness of using simple pixel-based representation on challenging surveillance videos, including more sophisticated features is expected to improve the time delayed correlation and dependency analysis provided that the features can be computed reliably. In particular, motion and direction features can be employed to facilitate the estimation of multi-mode time delays and correlations as explained above. Object appearance features can be combined with correlation analysis to further strengthen the estimation of inter-camera transition statistics, *e.g*. using similarity of object appearance across camera views to filter out spurious cross-correlation values [167].

4. *Exploiting person re-identification result* - After one performing person re-identification based on the time delayed correlations estimated, one could exploits the object correspondence to improve subsequent time delayed correlation estimation, *e.g*. adjusting time delay estimation using the time gaps between object correspondences.

From the point of view of the TD-PGM and two-stage structure learning algorithm, there are two directions in which to extend the techniques:

1. *Cyclic dependency* - The TD-PGM assumes acyclicity and unidirectional dependency, thus

it is not capable of learning cross-camera activity with cyclic dependencies, *e.g.* roundabout's traffic. This can be resolved by using graphical model that encodes cyclic probabilistic dependencies such as a dependency network [96]. This however comes with a price of having more expensive inference that requires an approximate algorithm such as Gibbs sampling.

2. *Weak supervision* - To facilitate the learning of a better network structure, it is possible to introduce weak supervision into the learning process. In particular, an expert can provide partial knowledge in the form of an ordering of variables $\prec$ or more specifically a prior probability that a variable X_j being a parent of X_i, denoted by $p(X_j \rightarrow X_i | \prec)$. A set of these prior probabilities can then be utilised as a medium to enforce weak supervision during the scoring phase in the structure learning. Typically, one has to specify a new parameter to balance the degree of human intervention and data-driven process to generate a network structure.

7.2 Learning from Human Feedback via Active Learning

Apart from multi-camera visual context learning, this thesis has also demonstrated the importance of learning from human feedback in detecting subtle unusual events. In particular, unlike existing techniques that perform passive mining on unusual events, a novel approach is formulated in Chapter 6 to actively learn from human feedback on ambiguous points to incrementally improve the unusual event detection performance.

The proposed active learning approach has several features as follows. Firstly, it is devised as a stream-based solution to enable real-time response. Secondly, it is capable of selecting different criterion at each time step for dynamic exploration and exploitation according to degree of influence of each criterion to the current model. Thirdly, it uses a prior to favour unusual event classes to give a more balanced query selection. Experimental results have shown that the proposed strategy outperformed existing supervised and unsupervised learning methods in detecting subtle unusual events in public scenes.

There are three primary areas into which the active learning approach can be extended:

1. *Binary feedback* - In this study, an oracle (*e.g.* surveillance operator) is expected to provide exact class label in response to each request of the active learner. This labelling strategy can be labour-intensive from a user interaction point of view for the following reasons:

- The effort and time for labelling increase along with the number of classes.
- The interaction may be prone to mistakes in annotation, *i.e*. the annotator may confuse with large amount of different activity classes.
- It is nontrivial to extend to multi-oracle or distributed labelling since consistent multi-class labelling is demanded.

In practice, a system would be more user-friendly if it accepts binary feedback during the active learning process, such as yes/no, normal/unusual or interesting/uninteresting responses instead of precise class tags. This can be achieved in several ways, for example:

- After being provided with a binary label, an active learner could further classify the events into finer classes automatically (*e.g*. from 'normal' label to more specific classes such as vertical, rightward, and leftward traffic flows). This feature is desired for its user-friendliness but the implementation can be nontrivial. Specifically, apart from the active learning process, a model is expected to perform additional stage of classification, in which it needs to assign a coarsely-labelled event with an accurate predictive label for incremental learning. Accurate and finer class prediction may not be easily achievable given limited training data at the beginning of stream-based learning. Moreover, an early-stage classification error may propagate to subsequent active learning phase, leading to poor learning performance.
- Another possible active learning strategy requires only yes/no answer from a user. Specifically, in addition to the ambiguous query point selected following a set of predefined query criteria, the active learning algorithm also chooses a sample point from a pool of labelled training data based on the estimated class probabilities of the query point. The query-sample pair is then presented to a user: a 'yes' is given if there is a match between the pair, and 'no' if otherwise. This feedback is then exploited to train a multi-class classifier. This strategy has been applied successfully on object recognition domain [122] but not yet explored within video surveillance context.

2. *Cost-sensitive active learning* - Current approach assumes that same amount of effort and time for annotation are required for all samples. This assumption may be unrealistic since different video clips might exhibit varying difficulty owning to different number of objects and activity complexity. Hence, some video clips may demand longer time for labelling.

It would be useful if an active learning algorithm could foresee the possible cost of annotation, therefore maximising the classifier's objective function without overspending a budgeted cost. The cost of annotation can be computed automatically based on the complexity of a scene, *e.g*. the number of objects that co-exist in the scene. It is worth pointing out that cost-sensitive active learning has been proposed for the applications on object recognition and content-based retrieval [226, 227] but not yet investigated for video-based learning.

3. *Active learning applied to multiple cameras context* - The proposed active learning approach has only been validated using single-view surveillance videos. Nevertheless, it is possible to extend the current strategy to multiple cameras context. In particular, one could first employ the method described in Chapter 4 to estimate a model B that represents the dependencies between inter-camera activity patterns. Then, a class node C can be added to B such that each node in B has C as a parent. The resultant graph structure is commonly known as augmented naïve Bayesian classifier [71, 129]. Given the induced graph structure, active learning can then be performed following the method described in Chapter 6.

Current video surveillance technologies mostly suffer high false alarm rate, sensitivity to visual context changes due to hard-wired rules, and poor scalability to crowded public scenes. This thesis has presented several techniques that may help in addressing these problems, *e.g*. context modelling, non-trajectory based representation, global activity inference, and human-in-the-loop active learning. The hypotheses set up at the beginning of this thesis have been verified on videos captured from busy public scenes such as underground stations and traffic intersections. In particular, the thesis has shown that the learning of time delayed correlation and dependency between regional activities can benefit activity understanding and unusual event detection in multiple disjoint cameras with non-overlapping views. In addition, the thesis has demonstrated that learning from human feedback via active learning could achieve better unusual event detection performance compared to unsupervised learning strategy. It could also yield a comparable event classification performance with fewer labelled samples compared to random sampling strategy. Performing adaptive selection of different query criteria can facilitate the joint discovery of un-

known events (exploration) and classification boundary refinement (exploitation) to certain extent. However, it is inconclusive that the proposed adaptive selection scheme can select consistently the best criterion at different learning phases for different datasets. More experiments are required to test this hypothesis. Lastly, the thesis has shown that non object-centred representation (*i.e*. without explicit object segmentation and tracking) is useful for activity understanding and unusual event detection in crowded spaces captured by low-quality video data.

Appendix A

Maximum Likelihood Estimation of Dependence Tree

Chow and Liu [47] propose a method to estimate the maximum likelihood of a dependence tree based on the mutual information between variables. The method is employed in the two-stage structure learning described in Section 4.1.3. The proof of the algorithm is included here.

Considering a tree network $\mathcal{T}$ with n variables, $\mathbf{X} = \{X_i \mid i = 1, 2, \ldots, n\}$ and the specific value taken by X_i is denoted as x_i. $\mathbf{Pa}(X_i)$ represents the set of parents of X_i, whereas $\mathbf{pa}(X_i)$ is an instantiation of $\mathbf{Pa}(X_i)$. The log-likelihood of m independent samples, $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_j, \ldots, \mathbf{x}_m\}$ with $\mathbf{x}_j = \{x_{i,j} \mid i = 1, 2, \ldots, n\}$ from the distribution of the tree is given as:

$$\begin{aligned} \log p(\mathbf{x}_1, \ldots, \mathbf{x}_m \mid \mathcal{T}) &= \sum_{j=1}^{m} \log p(\mathbf{x}_j) \\ &= \sum_{j=1}^{m} \sum_{i=1}^{n} \log p\left(x_{i,j} \mid x_{\psi(i),j}\right), \end{aligned} \tag{A.1}$$

where mapping function $\psi(i)$ with $0 \leq \psi(i) < n$ defines the tree network $\mathcal{T}$ so that $X_{\psi(i)} = \mathbf{Pa}(X_i)$ if $\psi(i) > 0$ and $X_{\psi(i)} = \mathbf{Pa}(X_i) = \emptyset$ if $\psi(i) = 0$.

Let $\left(x_i, x_{\psi(i)}\right)$ represents combinations of finite value set from X_i and $X_{\psi(i)}$. By representing the log-likelihood score (Equation (A.1)) in information theoretic notions, one have

$$
\begin{aligned}
\log p\left(\mathbf{x}_1,\ldots,\mathbf{x}_m \mid \mathcal{T}\right) &= \sum_{j=1}^{m}\sum_{i=1}^{n}\log p\left(x_{i,j} \mid x_{\psi(i),j}\right) \\
&= \sum_{i=1}^{n}\sum_{\left(x_i,x_{\psi(i)}\right)}\log\left(\frac{N_{x_i|x_{\psi(i)}}}{N_{x_{\psi(i)}}}\right)^{N_{x_i|x_{\psi(i)}}} \\
&= -m\sum_{i=1}^{n}\left[-\sum_{\left(x_i,x_{\psi(i)}\right)}\frac{N_{x_i|x_{\psi(i)}}}{m}\log\left(\frac{N_{x_i|x_{\psi(i)}}}{N_{x_{\psi(i)}}}\right)\right] \\
&= -m\sum_{i=1}^{n}\mathrm{H}_{\mathcal{X}}\left(X_i \mid X_{\psi(i)}\right) \\
&= m\sum_{i=1}^{n}\mathrm{I}_{\mathcal{X}}\left(X_i; X_{\psi(i)}\right) - m\sum_{i=1}^{n}\mathrm{H}_{\mathcal{X}}\left(X_i\right), \qquad \text{(A.2)}
\end{aligned}
$$

since the empirical conditional entropy of X_i given $X_{\psi(i)}$, $\mathrm{H}_{\mathcal{X}}\left(X_i \mid X_{\psi(i)}\right)$ is written as

$$
\begin{aligned}
\mathrm{H}_{\mathcal{X}}\left(X_i \mid X_{\psi(i)}\right) &= -\sum_{\left(x_i,x_{\psi(i)}\right)}\frac{N_{x_i|x_{\psi(i)}}}{m}\log\left(\frac{N_{x_i|x_{\psi(i)}}}{N_{x_{\psi(i)}}}\right) \\
&= -\left[\mathrm{I}_{\mathcal{X}}\left(X_i; X_{\psi(i)}\right) - \mathrm{H}_{\mathcal{X}}\left(X_i\right)\right], \qquad \text{(A.3)}
\end{aligned}
$$

where $\mathrm{I}_{\mathcal{X}}\left(X_i; X_{\psi(i)}\right)$ denotes the mutual information between X_i and $X_{\psi(i)}$.

The equality in Equation (A.2) suggests that maximising the log-likelihood of the tree is equivalent to search for a tree structure that minimises the conditional entropy or maximising the mutual information. Specifically, note that the second term of the last line in Equation (A.2) is independent of the tree $\mathcal{T}$. Therefore, in order to maximise $\log p\left(\mathbf{x}_1,\ldots,\mathbf{x}_m \mid \mathcal{T}\right)$, one need to maximise only the first term. To search for a tree such that the first term is maximised, one can follow the Prim's algorithm [183].

Appendix B

Sampling Dirichlet Random Vector via Gamma Generator

In Chapter 6, the parameters of a new committee member in the Query-by-Committee (QBC) algorithm are generated by sampling from Dirichlet distribution via the Gamma generator. The proof of the algorithm is described here.

A multinomial distribution over counts m_k for a K-state discrete variable is given as:

$$\text{Mult}\,(m_1, m_2, \ldots, m_K \mid \boldsymbol{\theta}, N\} = \binom{N}{m_1, m_2, \ldots, m_K} \prod_{k=1}^{K} \theta_k^{m_k}, \tag{B.1}$$

where N is the total number of observations and each of the elements in $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)$ gives the probability of the variable taking state k. These probabilities are subject to the constraints $0 \leq \theta_k \leq 1$ and $\sum_k \theta_k = 1$.

The conjugate prior distribution for the parameters $\{\theta_k\}$ is the Dirichlet distribution, denoted $(\theta_1, \ldots, \theta_K) \sim \text{Dir}\,(\alpha_1, \ldots, \alpha_K)$, which can be written as:

$$\begin{aligned} \text{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) &\propto \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \\ &= \frac{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1}, \end{aligned} \tag{B.2}$$

where $\boldsymbol{\alpha}$ denotes $(\alpha_1, \ldots, \alpha_K)$ and $\alpha_k \in \mathbb{R}^+$ are parameters of the Dirichlet distribution. Here

$\Gamma(x)$ denotes the Gamma function, which is defined as:

$$\Gamma(x) = \int_0^\infty \mathrm{e}^{-u} u^{x-1}\, \mathrm{d}u. \tag{B.3}$$

It is possible to generate $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)$ as a Dirichlet random vector by transforming a sequence of Gamma random variables [60]. Specifically, K independent Gamma random samples $\varepsilon_1, \ldots, \varepsilon_K$ are first drawn from Gamma distributions each with density:

$$\mathrm{Gam}(\varepsilon_k \mid \alpha_k) = \frac{1}{\Gamma(\alpha_k)} \mathrm{e}^{-\varepsilon_k} \varepsilon_k^{\alpha_k - 1}, \tag{B.4}$$

where α_k represents a shape parameter of the Gamma distribution. The Dirichlet random vector can then be obtained by setting

$$\theta_k = \frac{\varepsilon_k}{\sum_{k=1}^K \varepsilon_k}. \tag{B.5}$$

Proof. Let $\varepsilon_k \sim \mathrm{Gam}(\varepsilon_k \mid \alpha_k)$ as defined in Equation (B.4). The joint probability density function (pdf) of $\varepsilon_1, \ldots, \varepsilon_K$ is

$$f(\varepsilon_1, \ldots, \varepsilon_K \mid \alpha_1, \ldots, \alpha_K) = \prod_{k=1}^K \frac{1}{\Gamma(\alpha_k)} \mathrm{e}^{-\varepsilon_k} \varepsilon_k^{\alpha_k - 1}. \tag{B.6}$$

Through the change of variable, set

$$\gamma = \sum_{k=1}^K \varepsilon_k,\ \theta_1 = \frac{\varepsilon_1}{\gamma},\ \theta_2 = \frac{\varepsilon_2}{\gamma},\ \ldots,\ \theta_{K-1} = \frac{\varepsilon_{K-1}}{\gamma}, \tag{B.7}$$

which can be transformed into

$$\varepsilon_1 = \gamma\theta_1,\ \varepsilon_2 = \gamma\theta_2,\ \ldots,\ \varepsilon_{K-1} = \gamma\theta_{K-1},\ \varepsilon_K = \gamma\left(1 - \sum_{k=1}^{K-1} \theta_k\right). \tag{B.8}$$

The Jacobian determinant of the transformation is

$$
\begin{aligned}
\text{Jacobian}\left(\frac{\varepsilon_1,\varepsilon_2,\ldots,\varepsilon_K}{\gamma,\theta_1,\theta_1,\ldots,\theta_{K-1}}\right) &= \begin{vmatrix} \frac{\partial\varepsilon_1}{\partial\gamma} & \frac{\partial\varepsilon_1}{\partial\theta_1} & \frac{\partial\varepsilon_1}{\partial\theta_2} & \cdots & \frac{\partial\varepsilon_1}{\partial\theta_{K-1}} \\ \frac{\partial\varepsilon_2}{\partial\gamma} & \frac{\partial\varepsilon_2}{\partial\theta_1} & \frac{\partial\varepsilon_2}{\partial\theta_2} & \cdots & \frac{\partial\varepsilon_2}{\partial\theta_{K-1}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial\varepsilon_{K-1}}{\partial\gamma} & \frac{\partial\varepsilon_{K-1}}{\partial\theta_1} & \frac{\partial\varepsilon_{K-1}}{\partial\theta_2} & \cdots & \frac{\partial\varepsilon_{K-1}}{\partial\theta_{K-1}} \\ \frac{\partial\varepsilon_K}{\partial\gamma} & \frac{\partial\varepsilon_K}{\partial\theta_1} & \frac{\partial\varepsilon_K}{\partial\theta_2} & \cdots & \frac{\partial\varepsilon_K}{\partial\theta_{K-1}} \end{vmatrix} \\
&= \begin{vmatrix} \theta_1 & \gamma & 0 & \cdots & 0 \\ \theta_2 & 0 & \gamma & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \theta_{K-1} & 0 & 0 & \cdots & \gamma \\ 1-\sum_{k=1}^{K-1}\theta_k & -\gamma & -\gamma & \cdots & -\gamma \end{vmatrix} \\
&= \gamma^{K-1}. \qquad \text{(B.9)}
\end{aligned}
$$

From Equations (B.6) and (B.8), it follows that the joint pdf of $\gamma,\theta_1,\theta_2,\ldots,\theta_{K-1}$ is

$$
\begin{aligned}
& f\left(\gamma,\theta_1,\ldots,\theta_{K-1} \mid \alpha_1,\ldots,\alpha_K\right) \\
&= \frac{\left(\mathrm{e}^{-\gamma}\right)\left(\gamma^{\sum_{k=1}^{K}\alpha_k-1}\right)}{\prod_{k=1}^{K}\Gamma(\alpha_k)}\left[\theta_1^{\alpha_1-1}\theta_2^{\alpha_2-1}\ldots\theta_{K-1}^{\alpha_{K-1}-1}\left(1-\sum_{k=1}^{K-1}\theta_k\right)^{\alpha_K-1}\right]. \qquad \text{(B.10)}
\end{aligned}
$$

The Dirichlet distribution can be obtained by integrating out γ as follows:

$$
\begin{aligned}
& \int_0^\infty f\left(\gamma,\theta_1,\ldots,\theta_{K-1} \mid \alpha_1,\ldots,\alpha_K\right)\mathrm{d}\gamma \\
&= \frac{\int_0^\infty\left(\mathrm{e}^{-\gamma}\right)\left(\gamma^{\sum_{k=1}^{K}\alpha_k-1}\right)\mathrm{d}\gamma}{\prod_{k=1}^{K}\Gamma(\alpha_k)}\left[\theta_1^{\alpha_1-1}\,\theta_2^{\alpha_2-1}\,\ldots\,\theta_{K-1}^{\alpha_{K-1}-1}\left(1-\sum_{k=1}^{K-1}\theta_k\right)^{\alpha_K-1}\right] \\
&= \frac{\Gamma\left(\sum_{k=1}^{K}\alpha_k\right)}{\prod_{k=1}^{K}\Gamma(\alpha_k)}\left[\theta_1^{\alpha_1-1}\theta_2^{\alpha_2-1}\ldots\theta_{K-1}^{\alpha_{K-1}-1}\left(1-\sum_{k=1}^{K-1}\theta_k\right)^{\alpha_K-1}\right], \qquad \text{(B.11)}
\end{aligned}
$$

where $\int_0^\infty\left(\mathrm{e}^{-\gamma}\right)\left(\gamma^{\sum_{k=1}^{K}\alpha_k-1}\right)\mathrm{d}\gamma$ is replaced with $\Gamma\left(\sum_{k=1}^{K}\alpha_k\right)$ according to Equation (B.3).

Finally, the following Dirichlet distribution is obtained according to Equation (B.11)

$$
\left(\theta_1,\ldots,\theta_K\right)\sim\mathrm{Dir}\left(\alpha_1,\ldots,\alpha_K\right), \qquad \text{(B.12)}
$$

where $\theta_K = 1-\sum_{k=1}^{K-1}\theta_k$.

Bibliography

[1] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(3):555–560, 2008.

[2] H. Aghajan and A. Cavallaro, editors. *Multi-Camera Networks: Principles and Applications*. Elsevier, 2009.

[3] M. Albanese, R. Chellappa, V. Moscato, A. Picariello, V. S. Subrahmanian, P. Turaga, and O. Udrea. A constrained probabilistic Petri net framework for human activity detection in video. *IEEE Transactions on Multimedia*, 10(6):982–996, 2008.

[4] S. Ali and M. Shah. A Lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6, 2007.

[5] S. Ali and M. Shah. Floor fields for tracking in high density crowd scenes. In *European Conference on Computer Vision*, pages 1–24, 2008.

[6] E. L. Andrade, S. Blunsden, and R. B. Fisher. Hidden Markov models for optical flow analysis in crowds. In *International Conference on Pattern Recognition*, pages 460–463, 2006.

[7] E. L. Andrade, S. Blunsden, and R. B. Fisher. Modelling crowd scenes for event detection. In *International Conference on Pattern Recognition*, pages 175–178, 2006.

[8] N. Anjum and A. Cavallaro. Trajectory association and fusion across partially overlapping cameras. In *International Conference of Advanced Video and Signal Based Surveillance*, pages 201–206, 2009.

[9] S. Argamon-Engelson and I. Dagan. Committee-based sample selection for probabilistic classifiers. *Journal of Artificial Intelligence Research*, 11:335–360, 1999.

[10] C. Auliac, V. Frouin, X. Gidrol, and F. d'Alch Buc. Evolutionary approaches for the reverse-engineering of gene regulatory networks: a study on a biologically realistic dataset. *BMC Bioinformatics*, 9(91):1–14, 2008.

[11] M. Bar. Visual objects in context. *Nature Reviews: Neuroscience*, 5:617–629, 2004.

[12] Y. Baram, R. El-Yaniv, and K. Luz. Online choice of active learning algorithms. *Journal of Machine Learning Research*, 5:255–291., 2004.

[13] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[14] M. Beal, Z. Ghahramani, and C. Rasmussen. The infinite hidden Markov model. In *Advances in Neural Information Processing Systems*, 2002.

[15] S. S. Beauchemin and J. L. Barron. The computation of optical flow. *ACM Computing Surveys*, 27(3):433–466, 1995.

[16] I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The alarm monitoring system: A case study with two probabilistic inference techniques for belief networks. In *European Conference on Artificial Intelligence in Medicine*, pages 247–256, 1989.

[17] Y. Benezeth, P.-M. Jodoin, V. Saligrama, and C. Rosenberger. Abnormal events detection based on spatio-temporal co-occurrences. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2458–2465, 2009.

[18] B. Benfold and I. Reid. Guiding visual surveillance by tracking human attention. In *British Machine Vision Conference*, 2009.

[19] A. Bhattacharyya. On a measure of divergence between two statistical populations defined by probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35:99–109, 1943.

[20] I. Biederman. *On the Semantics of a Glance at a Scene*, chapter 8, pages 213–263. Perceptual Organization, 1981.

[21] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, 2007.

[22] D. Blei and J. Lafferty. *Topic models. In Text Mining: Theory and Applications*. Taylor and Francis, 2009.

[23] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

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

[25] O. Boiman and M. Irani. Detecting irregularities in images and in video. *International Journal of Computer Vision*, 74(1):17–31, 2007.

[26] M. Brand. Understanding manipulation in video. In *International Conference on Automatic Face and Gesture Recognition*, pages 94–99, 1996.

[27] M. Brand, N. Oliver, and A. Pentland. Coupled hidden Markov models for complex action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 994–999, 1997.

[28] M. D. Breitenstein. *Visual Surveillance - Dynamic Behavior Analysis at Multiple Levels*. PhD thesis, ETH Zurich, 2009.

[29] M. D. Breitenstein, H. Grabner, and L. V. Gool. Hunting Nessie - real-time abnormality detection from webcams. In *IEEE International Workshop on Visual Surveillance*, 2009.

[30] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Online multi-person tracking-by-detection from a single, uncalibrated camera. Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence.

[31] M. D. Breitenstein, E. Sommerlade, B. Leibe, L. V. Gool, and I. Reid. Probabilistic parameter selection for learning scene structure from video. In *British Machine Vision Conference*, 2008.

[32] G. J. Brostow and R. Cipolla. Unsupervised Bayesian detection of independent motion in crowds. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 594–601, 2006.

[33] V. Bruce, M. A. Georgeson, and P. R. Green. *Visual Perception: Physiology, Psychology and Ecology*. Psychology Press, 4 edition, 2003.

[34] W. Buntine. Theory refinement on Bayesian networks. In *Uncertainty in Artificial Intelligence*, pages 52–60, 1991.

[35] H. Buxton and S. Gong. Visual surveillance in a dynamic and uncertain world. *Artificial Intelligence*, 78(1-2):431–459, 1995.

[36] Q. Cai and J. K. Aggarwal. Tracking human motion in structured environments using a distributed-camera system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1241–1247, 1999.

[37] Y. Caspi and M. Irani. A step towards sequence-to-sequence alignment. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 682–689, 2000.

[38] N. Cebron and M. R. Berthold. Active learning for object classification: from exploration to exploitation. *Data Mining and Knowledge Discovery*, 18(2):283–299, 2008.

[39] T.-H. Chang and S. Gong. Tracking multiple people with a multi-camera system. In *IEEE Workshop Multi-Object Tracking*, pages 19–26, 2001.

[40] K.-W. Chen, C.-C. Lai, Y.-P. Hung, and C.-S. Chen. An adaptive learning method for target tracking across multiple cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[41] T. P. Chen, H. Haussecker, A. Bovyrin, R. Belenov, K. Rodyushkin, A. Kuranov, and V. Eruhimov. Computer vision workload analysis: Case study of video surveillance systems. *Intel Technology Journal*, 9(2):109–118, 2005.

[42] X. Chen, G. Anantha, and X. Lin. Improving Bayesian network structure learning with mutual information-based node ordering in the K2 algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 20(5):628–640, 2008.

[43] X. Chen, G. Anantha, and X. Wang. An effective structure learning method for constructing gene networks. *Bioinformatics*, 22(11):1367–1374, 2006.

[44] Z. Cheng, D. Devarajan, and R. J. Radke. Determining vision graphs for distributed camera networks using feature digests. *EURASIP Journal on Applied Signal Processing*, 2007(1):220–220, 2007.

[45] D. M. Chickering. Learning equivalence classes of Bayesian-network structures. *Journal of Machine Learning Research*, 2:445–498, 2002.

[46] D. M. Chickering, D. Geiger, and D. Heckerman. Learning Bayesian networks: Search methods and experimental results. In *International Workshop on Artificial Intelligence and Statistics*, pages 112–128, 1995.

[47] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.

[48] N. Cohen, J. Gatusso, and K. MacLennan-Brown. *CCTV Operational Requirements Manual - Is your CCTV system fit for purpose?* Home Office, version 4 (55/06) edition, 2006.

[49] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 142–149, 2000.

[50] G. F. Cooper. A simple constraint-based algorithm for efficiently mining observational databases for causal relationships. *Data Mining and Knowledge Discovery*, 1(2):203–224, 1997.

[51] G. F. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9(4):309–347, 1992.

[52] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. The MIT Press, 3 edition, 2009.

[53] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2006.

[54] F. Cozman, I. Cohen, and M. Cirelo. Semi-supervised learning of mixture models. In *International Conference on Machine Learning*, pages 99–106, 2003.

[55] I. Dagan and S. Engelson. Committee-based sampling for training probabilistic classifiers. In *International Conference on Machine Learning*, pages 150–157, 1995.

[56] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 886–893, 2005.

[57] C. P. de Campos, Z. Zeng, and Q. Ji. Structure learning of Bayesian networks using constraints. In *International Conference on Machine Learning*, pages 113–120, 2009.

[58] H. Dee and D. Hogg. Detecting inexplicable behaviour. In *British Machine Vision Conference*, pages 477–486, 2004.

[59] H. M. Dee and S. A. Velastin. How close are we to solving the problem of automated visual surveillance? *Machine Vision and Applications*, 19(5–6):329–343, 2008.

[60] L. Devroye. *Non-Uniform Random Variate Generation*. Springer-Verlag, 1986.

[61] R. L. Dobrushin. Prescribing a system of random variables by conditional distributions. *Theory of Probability and Its Applications*, 15:458–486, 1970.

[62] Y. Du, F. Chen, and W. Xu. Human interaction representation and recognition through motion decomposition. *IEEE Signal Processing Letters*, 14(12):952–955, 2007.

[63] Y. Du, F. Chen, W. Xu, and Y. Li. Recognizing interaction activities using dynamic Bayesian network. In *International Conference on Pattern Recogntion*, pages 618–621, 2006.

[64] T. Duong, H. Bui, D. Phung, and S. Venkatesh. Activity recognition and abnormality detection with the switching hidden semi-Markov model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 838–845, 2005.

[65] D. Eaton and K. Murphy. Bayesian structure learning using dynamic programming and MCMC. In *Uncertainty in Artificial Intelligence*, pages 101–108, 2007.

[66] T. Ellis, D. Makris, and J. Black. Learning a multi-camera topology. In *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 165–171, 2003.

[67] S. Ertekin, J. Huang, L. Bottou, and C. L. Giles. Learning on the border: Active learning in imbalanced data classification. In *Conference on Information and Knowledge Management*, pages 127–136, 2007.

[68] I. Everts, N. Sebe, and G. Jones. Cooperative object tracking with multiple PTZ cameras. In *International Conference on Image Analysis and Processing*, pages 323–330, 2007.

[69] M. Farenzena, L. Bazzani, A. Perina, M. Cristani, and V. Murino. Person re-identification by symmetry-driven accumulation of local features. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2360 –2367, 2010.

[70] A. M. Fraser and H. L. Swinney. Independent coordinates for strange attractors from mutual information. *Physical Review*, 33(2):1134–1140, 1986.

[71] N. Friedman and M. Goldszmidt. Building classifiers using Bayesian networks. In *AAAI Conference on Artificial Intelligence*, pages 1277–1284, 1996.

[72] N. Friedman and M. Goldszmidt. Sequential update of Bayesian network structure. In *Uncertainty in Artificial Intelligence*, pages 165–174, 1997.

[73] N. Friedman and D. Koller. Being Bayesian about network structure. In *Uncertainty in Artificial Intelligence*, pages 201–210, 2000.

[74] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7(3-4):601–620, 2000.

[75] N. Friedman, I. Nachman, and D. Peér. Learning Bayesian network structure from massive datasets: The sparse candidate algorithm. In *Uncertainty in Artificial Intelligence*, pages 206–215, 1999.

[76] Frost & Sullivan. Video surveillance software emerges as key weapon in fight against terrorism. Press release - http://www.frost.com/.

[77] Frost & Sullivan. Eyes on the network - Understanding the shift toward network-based video surveillance in Asia, 2007.

[78] Z. Fu, W. Hu, and T. Tan. Similarity based vehicle trajectory clustering and anomaly detection. In *International Conference on Image Processing*, pages 602–605, 2005.

[79] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions of Information Theory*, 21:32–40, 1975.

[80] A. Galata, N. Johnson, and D. Hogg. Learning variable length Markov models of behaviour. *Computer Vision and Image Understanding*, 81(3):398–413, 2001.

[81] Z. Ghahramani. Unsupervised learning. In *Advanced Lectures on Machine Learning*, pages 72–112. Springer-Verlag, 2004.

[82] N. Gheissari, T. B. Sebastian, J. Rittscher, and R. Hartley. Person reidentification using spatiotemporal appearance. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1528–1535, 2006.

[83] A. Gilbert and R. Bowden. Incremental, scalable tracking of objects inter camera. *Computer Vision and Image Understanding*, 111(1):43–58, 2008.

[84] W. Gilks, S. Richardson, and D. Spiegelhalter, editors. *Markov Chain Monte Carlo in Practice*. Chapman and Hall, 1 edition, 1995.

[85] P. M. Gill, A. Spriggs, J. Allen, M. Hemming, P. Jessiman, D. Kara, J. Kilworth, R. Little, and D. Swain. Control room operation: findings from control room observations. Home office online report 14/05, Home Office, 2005.

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

[87] S. Gong and T. Xiang. Scene event recognition without tracking. *Acta Automatica Sinica*, 29(3):321–331, 2003.

[88] V. Gouaillier and A.-E. Fleurant. Intelligent video surveillance: Promises and challenges. Technological and commercial intelligence report, CRIM and Technôpole Defence and Security, 2009.

[89] D. Gray and H. Tao. Viewpoint invariant pedestrain recognition with an ensemble of localized features. In *European Conference on Computer Vision*, pages 262–275, 2008.

[90] M. W. Green. The appropriate and effective use of security technologies in U.S. schools. Technical Report NCJ 178265, Sandia National Laboratories, 1999.

[91] P. Gurdjos and P. Sturm. Methods and geometry for plane-based self-calibration. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 491–496, 2003.

[92] G. Haller. Distinguished material surfaces and coherent structures in three-dimensional fluid flows. *Physica D*, 149(4):248–277, 2001.

[93] R. Hamid, A. Johnson, S. Batta, A. Bobick, C. Isbell, and G. Coleman. Detection and explanation of anomalous activities - representing activities as bags of event n-grams. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1031–1038, 2005.

[94] R. Hamid, S. Maddi, A. Bobick, and M. Essa. Structure from statistics - unsupervised activity analysis using suffix trees. In *IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[95] D. J. Hand and K. Yu. Idiot's bayes: Not so stupid after all? *International Statistical Review*, 69(3):385–398, 2001.

[96] D. Heckerman, D. M. Chickering, C. Meek, R. Rounthwaite, and C. Kadie. Dependency networks for inference, collaborative filtering, and data visualization. *Journal of Machine Learning Research*, 1:49–75, 2000.

[97] D. Heckerman, D. Geiger, and D. M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243, 1995.

[98] G. Heitz, S. Gould, A. Saxena, and D. Koller. Cascaded classification models: Combining models for holistic scene understanding. In *Advances in Neural Information Processing Systems*, 2008.

[99] E. Herskovits and G. Cooper. Kutató: An entropy-driven system for construction of probabilistic expert systems from databases. In *Uncertainty in Artificial Intelligence*, pages 117–128, 1990.

[100] S.-S. Ho and H. Wechsler. Query by transduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1557–1571, 2008.

[101] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1/2):177–196, 2001.

[102] S. Hongeng, F. Bremond, and R. Nevatia. Representation and optimal recognition of human activities. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 818–825, 2000.

[103] S. Hongeng, R. Nevatia, and F. Brémond. Video-based event recognition: Activity representation and probabilistic recognition methods. *Computer Vision and Image Understanding*, 96(2):129–162, 2004.

[104] T. Hospedales, S. Gong, and T. Xiang. A Markov clustering topic model for mining behaviour in video. In *IEEE International Conference on Computer Vision*, pages 1165–1172, 2009.

[105] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

[106] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, and S. Maybank. Principal axis-based correspondence between multiple cameras for people tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):663–671, 2006.

[107] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man, And Cybernetics*, 34(3):334–352, 2004.

[108] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank. A system for learning statistical motion patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1450–1464, 2006.

[109] T. Huang and S. J. Russell. Object identification in a bayesian context. In *International Joint Conferences on Artificial Intelligence*, pages 1276–1283, 1997.

[110] S. S. Intille and A. F. Bobick. A framework for recognizing multi-agent action from visual evidence. In *AAAI Conference on Artificial intelligence*, pages 518–525, 1999.

[111] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *European Conference on Computer Vision*, pages 343–356, 1996.

[112] L. Itti and C. Koch. Computational modelling of visual attention. *National Review Neuroscience*, 2(3):194–203, 2001.

[113] Y. A. Ivanov and A. F. Bobick. Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):852–872, 2000.

[114] A. K. Jain and R. C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.

[115] O. Javed, Z. Rasheed, K. Shafique, and M. Shah. Tracking across multiple cameras with disjoint views. In *IEEE International Conference on Computer Vision*, pages 952–957, 2003.

[116] O. Javed, K. Shafique, and M. Shah. Appearance modeling for tracking in multiple non-overlapping cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 26–33, 2005.

[117] O. Javed and M. Shah. *Automated Multi-camera Surveillance: Theory and Practice*. Springer, 2008.

[118] F. Jiang, Y. Wu, and A. K. Katsaggelos. A dynamic hierarchical clustering method for trajectory-based unusual video event detection. *IEEE Transactions on Image Processing*, 18(4):907–913, 2009.

[119] P.-M. Jodoin, J. Konrad, and V. Saligrama. Modeling background activity for behavior subtraction. In *International Conference on Distributed Smart Cameras*, pages 1–10, 2008.

[120] N. Johnson and D. C. Hogg. Learning the distribution of object trajectories for event recognition. *Image and Vision Computing*, 14(8):609–615, 1996.

[121] A. J. Joshi, F. Porikli, and N. Papanikolopoulos. Multi-class active learning for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2372–2379, 2009.

[122] A. J. Joshi, F. Porikli, and N. Papanikolopoulos. Breaking the interactive bottleneck in multi-class classification with active selection and binary feedback. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2995–3002, 2010.

[123] M. Kalisch and P. Buhlmann. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research*, pages 613–636, 2007.

[124] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[125] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Gaussian processes for object categorization. *International Journal of Computer Vision*, 88(2):169–188, 2009.

[126] A. Kapoor and E. Horvitz. On discarding, caching, and recalling samples in active learning. In *Uncertainty in Artificial Intelligence*, pages 209–216, 2007.

[127] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. In *IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[128] M. Kendall and J. K. Ord. *Time Series*. Edward Arnold, 1990.

[129] E. J. Keogh and M. J. Pazzani. Learning augmented Bayesian classifiers: A comparison of distribution-based and classification-based approaches. In *International Workshop on Artificial Intelligence and Statistics*, pages 225–230, 1999.

[130] S. Khan and M. Shah. Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1355–1360, 2003.

[131] J. Kim and K. Grauman. Observe locally, infer globally: a space-time MRF for detecting abnormal activities with incremental updates. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2928, 2009.

[132] L. Kratz and K. Nishino. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1446–1453, 2009.

[133] H. Kruegle. *CCTV Surveillance: Video Practices and Technology*. Butterworth-Heinemann, 2006.

[134] D. Kuettel, M. D. Breitenstein, L. V. Gool, and V. Ferrari. What's going on? Discovering spatio-temporal dependencies in dynamic scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1951–1958, 2010.

[135] W. Lam. Bayesian network refinement via machine learning approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):240–251, 1998.

[136] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958, 2009.

[137] G. Lavee, E. Rivlin, and M. Rudzsky. Understanding video events: A survey of methods for automatic interpretation of semantic occurrences in video. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 39(5):489–504, 2009.

[138] C. K. Lee, M. F. Ho, W. S. Wen, and C. L. Huang. Abnormal event detection in video using N-cut clustering. In *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pages 407–410, 2006.

[139] L. Lee, R. Romano, and G. Stein. Monitoring activities from multiple video streams: establishing a common coordinate frame. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):758–767, 2000.

[140] J. Li, S. Gong, and T. Xiang. Global behaviour inference using probabilistic latent semantic analysis. In *British Machine Vision Conference*, pages 193–202, 2008.

[141] J. Li, S. Gong, and T. Xiang. Scene segmentation for behaviour correlation. In *European Conference on Computer Vision*, pages 383–395, 2008.

[142] Z. Li, J. Liu, and X. Tang. Constrained clustering via spectral regularization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 421–428, 2009.

[143] T. W. Liao. Clustering of time series data - a survey. *Pattern Recognition*, 38(11):1857–1874, 2005.

[144] C. C. Loy, T. Xiang, and S. Gong. Incremental global activity dependency modelling. Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence.

[145] C. C. Loy, T. Xiang, and S. Gong. Stream-based active unusual event detection. In *Asian Conference on Computer Vision*, 2010.

[146] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of Imaging Understanding Workshop*, pages 121–130, 1981.

[147] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos. Anomaly detection in crowded scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[148] D. Makris and T. Ellis. Learning semantic scene models from observing activity in visual surveillance. *IEEE Transactions on Systems, Man, and Cybernetics*, 35(3):397–408, 2005.

[149] D. Makris, T. Ellis, and J. Black. Bridging the gaps between cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 205–210, 2004.

[150] D. Marinakis and G. Dudek. Self-calibration of a vision-based sensor network. *Image and Vision Computing*, 27(1-2):116–130, 2009.

[151] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[152] A. K. McCallum and K. Nigam. Employing EM in pool-based active learning for text classification. In *International Conference on Machine Learning*, pages 350–358, 1998.

[153] E. M. McCreight. A space-economical suffix tree construction algorithm. *Journal of the ACM*, 23(2):262–272, 1976.

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

[155] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behaviour detection using social force model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 935–942, 2009.

[156] G. A. Miller. The magical number seven plus or minus two: some limits on our capacity for processing information. *Psychological Review*, 63(2):81–97, Mar 1956.

[157] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006.

[158] R. J. Morris and D. C. Hogg. Statistical models of object interaction. *International Journal of Computer Vision*, 37(2):209–215, 2000.

[159] K. P. Murphy. Active learning of causal Bayes net structure. Technical report, University of California, 2001.

[160] K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California at Berkeley, Computer Science Division, 2002.

[161] A. Naftel and S. Khalid. Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space. *Multimedia Systems*, pages 227–238, 2006.

[162] J. C. Nascimento, M. A. T. Figueiredo, and J. S. Marques. Semi-supervised learning of switched dynamical models for classification of human activities in surveillance applications. In *IEEE International Conference on Image Processing*, pages 197–200, 2007.

[163] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856, 2001.

[164] H. T. Nguyen, Q. Ji, and A. W. M. Smeulders. Spatio-temporal context for robust multitarget tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1):52–64, 2007.

[165] N. T. Nguyen, D. Q. Phung, S. Venkatesh, and H. H. Bui. Learning and detecting activities from movement trajectories using the hierarchical hidden Markov model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 955–960, 2005.

[166] S. H. Nielsen and T. D. Nielsen. Adapting Bayes network structures to non-stationary domains. *International Journal of Approximate Reasoning*, 49(2):379–397, 2008.

[167] C. Niu and E. Grimson. Recovering non-overlapping network topology using far-field vehicle tracking data. In *International Conference of Pattern Recognition*, pages 944–949, 2006.

[168] C. Norris, M. McCahill, and D. Wood. The growth of CCTV: a global perspective on the international diffusion of video surveillance in publicly accessible space. *Surveillance and Society*, 2(2/3):110–135, 2004.

[169] K. Okuma, A. Taleghani, N. D. Freitas, J. Little, and D. Lowe. A boosted particle filter: Multitarget detection and tracking. In *European Conference on Computer Vision*, pages 28–39, 2004.

[170] N. Oliver, E. Horvitz, and A. Garg. Layered representations for human activity recognition. In *IEEE International Conference of Multimodal Interfaces*, pages 3–8, 2002.

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

[172] J. Orozco, S. Gong, and T. Xiang. Head pose classification in crowded scenes. In *British Machine Vision Conference*, 2009.

[173] J. Owens and A. Hunter. Application of the self-organizing map to trajectory classification. In *IEEE International Workshop on Visual Surveillance*, pages 77–83, 2000.

[174] S. Palmer. The effects of contextual scenes on the identification of objects. *Memory and Cognition*, 3:519–526, 1975.

[175] S. Park and M. M. Trivedi. A two-stage multi-view analysis framework for human activity and interactions. In *IEEE Workshop on Motion and Video Computing*, 2007.

[176] J. Pearl. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann Publishers Inc., 1988.

[177] D. Pelleg and A. Moore. Active learning for anomaly and rare-category detection. In *Advances in Neural Information Processing Systems*, pages 1073–1080, 2004.

[178] M. Piccardi. Background subtraction techniques: a review. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 3099–3104, 2004.

[179] C. Piciarelli and G. Foresti. On-line trajectory clustering for anomalous events detection. *Pattern Recognition Letters*, 27:1835–1842, 2006.

[180] J. Pilet, C. Strecha, and P. Fua. Making background subtraction robust to sudden illumination changes. In *European Conference on Computer Vision*, pages 567–580, 2008.

[181] D. Pollard. *A User's Guide to Measure Theoretic Probability*. Cambridge Series in Statistical and Probabilistic Mathematics, 2001.

[182] R. Poppe. A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990, 2010.

[183] R. C. Prim. Shortest connection networks and some generalizations. *Bell System Technical Journal*, 36:1389–1401, 1957.

[184] J. C. Principe. *Information Theoretic Learning: Renyi's Entropy and Kernel Perspectives*. Springer, 1 edition, 2010.

[185] B. Prosser, S. Gong, and T. Xiang. Multi-camera matching under illumination change over time. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications*, 2008.

[186] B. Prosser, S. Gong, and T. Xiang. Multi-camera matching using bi-directional cumulative brightness transfer functions. In *British Machine Vision Conference*, 2008.

[187] B. Prosser, W. Zheng, S. Gong, and T. Xiang. Person re-identification by support vector ranking. In *British Machine Vision Conference*, 2010.

[188] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted Gaussian mixture models. In *Digital Signal Processing*, 2000.

[189] I. Rish. An empirical study of the naive bayes classifier. In *International Joint Conferences on Artificial Intelligence - Workshop on Empirical Methods in Artificial Intelligence*, 2001.

[190] N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *International Conference on Machine Learning*, pages 441–448, 2001.

[191] D. Russell and S. Gong. Minimum cuts of a time-varying background. In *British Machine Vision Conference*, pages 809–818, 2006.

[192] S. Russell and P. Norv. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2 edition, 1998.

[193] C. Sacchi, C. Regazzoni, G. Gera, and G. Foresti. A neural network-based image processing system for detection of vandal acts in unmanned railway environments. In *International Conference on Image Analysis and Processing*, pages 529–534, 2001.

[194] I. Saleemi, K. Shafique, and M. Shah. Probabilistic modeling of scene dynamics for applications in visual surveillance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(8):1472–1485, 2009.

[195] O. Schwartz, A. Hsu, and P. Dayan. Space and time in visual context. *Nature Reviews Neuroscience*, 8:522–535, 2007.

[196] G. Schwarz. Estimating the dimension of a model. *The Annals of Mathematical Statistics*, 6(2):461–464, 1978.

[197] B. Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison, 2010.

[198] B. Settles, M. Craven, and S. Ray. Multiple instance active learning. In *Advances in Neural Information Processing Systems*, 2007.

[199] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Conference on Learning Theory*, pages 287–294, 1992.

[200] Y. A. Sheikh and M. Shah. Trajectory association across multiple airborne cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):361–367, 2008.

[201] V. Shet, D. Harwood, and L. Davis. Multivalued default logic for identity maintenance in visual surveillance. In *European Conference on Computer Vision*, pages 119–132, 2006.

[202] Y. Shi, A. Bobick, and I. Essa. Learning temporal sequence model from partially labeled data. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1631–1638, 2006.

[203] Y. Shi, Y. Huang, D. Minnen, A. Bobick, and I. Essa. Propagation networks for recognition of partially ordered sequential action. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 862–869, 2004.

[204] R. Sillito and R. Fisher. Semi-supervised learning for anomalous trajectory detection. In *British Machine Vision Conference*, 2008.

[205] M. Singh and M. Valtorta. An algorithm for the construction of Bayesian network structures from data. In *Uncertainty in Artificial Intelligence*, pages 259–265, 1993.

[206] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. The MIT Press, 2 edition, 2000.

[207] C. Stauffer and W. E. L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–757, 2000.

[208] C. Stauffer and K. Tieu. Automated multi-camera planar tracking correspondence modeling. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 259–266, 2003.

[209] J. W. Stokes, J. C. Platt, J. Kravis, and M. Shilman. ALADIN: Active learning of anomalies to detect intrusions. Technical report, Microsoft Research, 2008.

[210] D. Sun, S. Roth, J. P. Lewis, and M. J. Black. Learning optical flow. In *European Conference on Computer Vision*, pages 83–97, 2008.

[211] K. Sung, Y. Hwang, and I. Kweon. Robust background maintenance for dynamic scenes with global intensity level changes. In *International Conference on Ubiquitous Robots and Ambient Intelligence*, pages 759–762, 2008.

[212] C. T. Symons, N. F. Samatova, R. Krishnamurthy, B. H. Park, T. Umar, D. Buttler, T. Critchlow, and D. Hysom. Multi-criterion active learning in conditional random fields. In *IEEE International Conference on Tools with Artificial Intelligenc*, pages 323–331, 2006.

[213] D. M. J. Tax. *One-class classification; concept-learning in the absence of counter-examples*. PhD thesis, Delft University of Technology, June 2001.

[214] C. C. Loy, T. Xiang, and S. Gong. Modelling activity global temporal dependencies using time delayed probabilistic graphical model. In *IEEE International Conference on Computer Vision*, pages 120–127, 2009.

[215] C. C. Loy, T. Xiang, and S. Gong. Multi-camera activity correlation analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1988–1995, 2009.

[216] C. C. Loy, T. Xiang, and S. Gong. Time-delayed correlation analysis for multi-camera activity understanding. *International Journal of Computer Vision*, 90(1):106–129, 2010.

[217] K. Tieu, G. Dalley, and W. E. L. Grimson. Inference of non-overlapping camera network topology by measuring statistical dependence. In *IEEE International Conference on Computer Vision*, pages 1842–1849, 2005.

[218] M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11(2):443–482, 1999.

[219] S. Tong and D. Koller. Active learning for parameter estimation in Bayesian networks. In *Advances in Neural Information Processing Systems*, pages 647–653, 2000.

[220] A. Torralba. Contextual influences on saliency. *Neurobiology of Attention*, 2005.

[221] S. D. Tran and L. S. Davis. Event modeling and recognition using Markov logic networks. In *European Conference on Computer Vision*, pages 610–623, 2008.

[222] B. Triggs. Camera pose and calibration from 4 or 5 known 3D points. In *IEEE International Conference on Computer Vision*, volume 1, pages 278–284, 1999.

[223] I. Tsamardinos, L. E. Brown, and C. F. Aliferis. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78, 2006.

[224] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities - a survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1473–1488, 2008.

[225] A. van den Hengel, A. Dick, and R. Hill. Activity topology estimation for large networks of cameras. In *IEEE Conference on Advanced Video and Signal Based Surveillance*, 2006.

[226] S. Vijayanarasimhan and K. Grauman. What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2262–2269, 2009.

[227] S. Vijayanarasimhan, P. Jain, and K. Grauman. Far-sighted active learning on a budget for image and video recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3035–3042, 2010.

[228] C. Vogler and D. Metaxas. A framework for recognizing the simultaneous aspects of American sign language. *Computer Vision and Image Understanding*, 81(3):358–384, 2001.

[229] M. Wang, Z. Chen, and S. Cloutier. A hybrid Bayesian network learning method for constructing gene networks. *Computational Biology and Chemistry*, 31:361–372, 2007.

[230] X. Wang, X. Ma, and E. Grimson. Unsupervised activity perception by hierarchical Bayesian models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[231] X. Wang, X. Ma, and W. E. L. Grimson. Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(3):539–555, 2009.

[232] X. Wang, K. Tieu, and E. Grimson. Learning semantic scene models by trajectory analysis. In *European Conference on Computer Vision*, pages 110–123, 2006.

[233] X. Wang, K. Tieu, and W. E. L. Grimson. Correspondence-free activity analysis and scene modeling in multiple camera views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):56–71, 2010.

[234] M. J. V. Wert, T. S. Horowitz, and J. M. Wolfe. Even in correctable search, some types of rare targets are frequently missed. *Attention, Perception, & Psychophysics*, 71:541–553, 2009.

[235] J. M. Wolfe, T. S. Horowitz, and N. M. Kenner. Rare items often missed in visual searches. *Nature*, 435:439–440, 2005.

[236] J. M. Wolfe, T. S. Horowitz, M. J. Van Wert, N. M. Kenner, S. S. Place, and N. Kibbi. Low target prevalence is a stubborn source of errors in visual search tasks. *Journal of Experimental Psychology: General*, 136(4):623–638, 2007.

[237] C. R. Wren, D. C. Minnen, and S. G. Rao. Similarity-based analysis for large networks of ultra-low resolution sensors. *Pattern Recognition*, 39(10):1918–1931, 2006.

[238] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *International Journal of Computer Vision*, 75(2):247–266, 2007.

[239] S. Wu, B. E. Moore, and M. Shah. Chaotic invariants of Lagrangian particle trajectories for anomaly detection in crowded scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2054–2060, 2010.

[240] T. Xiang and S. Gong. Beyond tracking: modelling activity and understanding behaviour. *International Journal of Computer Vision*, 67(1):21–51, 2006.

[241] T. Xiang and S. Gong. Incremental visual behaviour modelling. In *IEEE International Workshop on Visual Surveillance*, pages 65–72, 2006.

[242] T. Xiang and S. Gong. Activity based surveillance video content modelling. *Pattern Recognition*, 41(7):2309–2326, 2008.

[243] T. Xiang and S. Gong. Incremental and adaptive abnormal behaviour detection. *Computer Vision and Image Understanding*, 111(1):59–73, 2008.

[244] T. Xiang and S. Gong. Video behaviour profiling for anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):893–908, 2008.

[245] B. Xie, V. Ramesh, and T. Boult. Sudden illumination change detection using order consistency. *Image and Vision Computing*, 22(2):117–125, 2004.

[246] M. Yang, Y. Wu, and G. Hua. Context-aware visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(7):1195–1209, 2008.

[247] Y. Yang, J. Liu, and M. Shah. Video scene understanding using multi-scale analysis. In *IEEE International Conference on Computer Vision*, 2009.

[248] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Journal of Computing Surveys*, 38(4):1–45, 2006.

[249] L. Zelnik-Manor and M. Irani. Statistical analysis of dynamic actions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1530–1535, 2006.

[250] L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering. In *Advances in Neural Information Processing Systems*, pages 1601–1608, 2004.

[251] E. E. Zelniker, S. Gong, and T. Xiang. Global abnormal behaviour detection using a network of CCTV cameras. In *IEEE International Workshop on Visual Surveillance*, 2008.

[252] D. Zhang, D. Gatica-Perez, S. Bengio, and I. McCowan. Semi-supervised adapted HMMs for unusual event detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 611–618, 2005.

[253] D. Zhang, D. Gatica-Perez, S. Bengio, I. McCowan, and G. Lathoud. Modeling individual and group actions in meetings with layered HMMs. *IEEE Transactions on Multimedia*, 8(3):509–520, 2004.

[254] Z. Zhang, K. Huang, and T. Tan. Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes. In *International Conference on Pattern Recognition*, pages 1135–1138, 2006.

[255] T. Zhao and R. Nevatia. Tracking multiple humans in complex situations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:1208–1221, 2004.

[256] W. Zheng, S. Gong, and T. Xiang. Associating groups of people. In *British Machine Vision Conference*, 2009.

[257] W.-S. Zheng, S. Gong, and T. Xiang. Quantifying contextual information for object detection. In *IEEE International Conference on Computer Vision*, 2009.

[258] H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 819–826, 2004.

[259] H. Zhou and D. Kimber. Unusual event detection via multi-camera video mining. In *IEEE International Conference on Pattern Recognition*, pages 1161–1166, 2006.

[260] X. Zhou and T. Huang. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems*, 8:536–544, 2003.

[261] X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2007.

[262] Z. Zivkovic and F. van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 27(7):773–780, 2006.

[263] X. Zou, B. Bhanu, and A. Roy-Chowdhury. Continuous learning of a multilayered network topology in a video camera network. *Journal on Image and Video Processing*, 2009:1–19, 2009.