Subspace discovery for video anomaly detection

Tziakos, Ioannis

Subspace discovery for video anomaly detection

A thesis presented to the University of London

by

Ioannis Tziakos

for the degree of

Doctor of Philosophy

in

Electronic Engineering

School of Electronic Engineering and Computer Science,
Queen Mary, University of London,
Mile End Road, London, E1 4NS.

December, 2010

I confirm that the work presented in this thesis is my own and the work of other persons is appropriately acknowledged.

Sincerely yours,

Ioannis Tziakos

Abstract

In automated video surveillance anomaly detection is a challenging task. We address this task as a novelty detection problem where pattern description is limited and labelling information is available only for a small sample of normal instances. Classification under these conditions is prone to over-fitting. The contribution of this work is to propose a novel video abnormality detection method that does not need object detection and tracking. The method is based on subspace learning to discover a subspace where abnormality detection is easier to perform, without the need of detailed annotation and description of these patterns. The problem is formulated as one-class classification utilising a low dimensional subspace, where a novelty classifier is used to learn normal actions automatically and then to detect abnormal actions from low-level features extracted from a region of interest. The subspace is discovered (using both labelled and unlabelled data) by a locality preserving graph-based algorithm that utilises the Graph Laplacian of a specially designed parameter-less nearest neighbour graph.

The methodology compares favourably with alternative subspace learning algorithms (both linear and non-linear) and direct one-class classification schemes commonly used for off-line abnormality detection in synthetic and real data. Based on these findings, the framework is extended to on-line abnormality detection in video sequences, utilising multiple independent detectors deployed over the image frame to learn the local normal patterns and infer abnormality for the complete scene. The method is compared with an alternative linear method to establish advantages and limitations in on-line abnormality detection scenarios. Analysis shows that the alternative approach is better suited for cases where the subspace learning is restricted on the labelled samples, while in the presence of additional unlabelled data the proposed approach using graph-based subspace learning is more appropriate.

Acknowledgements

This work was funded by a grant jointly provided by the Engineering and Physical Sciences Research Council and BT Research & Venturing. It was performed under the supervision of Prof. Andrea Cavallaro, whose guidance was fundamental throughout this work and the value of whose vigilant direction, support and help was enormous. Valuable support and advice was provided by Dr. Li-Qun Xu. Great help, especially in proof-reading, was provided by other supervisees of Prof. Cavallaro. Valuable advice on parts of the work was provided by anonymous reviewers of the 7th Advanced Video and Signal-Based Surveillance (AVSS) conference, the Pattern Recognition Letters and Neurocomputing journals.

To my family,

those who were,

those who are,

those who will be...

Contents

Associated Publications

The following publications have been produced in association with this thesis:

Journal Papers

[J1] I. Tziakos, A. Cavallaro, and L.-Q. Xu, "Video event segmentation and visualisation in non-linear subspace", *Pattern Recognition Letters*, vol. 30, iss. 2, pp. 123–131, 2009.

[J2] I. Tziakos, A. Cavallaro, and L.-Q. Xu, "Event monitoring via local motion abnormality detection in non-linear subspace", *Neurocomputing*, Special Issue on Subspace learning, vol. 73, iss. 10-12, pp. 1881-1891, 2010.

Conference Papers

[C1] I. Tziakos, A. Cavallaro, and L.-Q. Xu, "Local abnormality detection in video using subspace learning", *in Proc. of the IEEE International Conference on Advanced Video and Signal-Based Surveillance*, Boston, MA, USA, 29 August - 1 September 2010, pp. 519 - 525.

Chapter 1

Introduction

1.1 Motivation

In recent years we have witnessed an increase in the deployment of large networks of CCTV cameras in key infrastructure sites such as underground transportation and airports. In these sites, it is desirable to detect events and behaviours that take place in the scene. This is a challenging task. Camera networks generate a large amount of data that needs to be filtered by a human operator, in order to identify instances that need further attention. Under these operating conditions issues like fatigue, operational stress and change blindness occur that greatly affect the ability of the human operator to detect events [1].

Machine vision is focused on providing the CCTV camera system with additional intelligence and help security personnel by filtering out unnecessary and redundant information. A challenging security task is to perform abnormal event detection and detect rare and unexpected actions. For example, the motion of a person walking against the usual flow of people in an airport site is marked abnormal because it is statistically infrequent. Based on the context, these actions might be dangerous or even illegal. Thus, automatic detection of unexpected action is a desired task. Machine vision systems can provide real-time alarms and warnings for security or medical personnel to act and intervene. In the long term, statistical information gathered can prove useful to redesign safety procedures and improve security services.

However, although the problem of abnormality detection in video sequences has a simple definition, the solution is not an easy task. Related work addresses the abnormality

detection problem as novelty detection. Features extracted from a video sequence are used to model the normal (usual) behaviour that takes place in front of the camera and the trained classifiers rank new data as novel based on whether these instances have been seen during training. A common approach aims to track the moving objects in the scene and detect abnormal trajectories [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]. More recent research is geared towards using low-level cues (*e.g.* motion) when trajectories are unreliable [13, 14, 15]. While abnormality detection frameworks cover a wide diversity of methodologies, each of them suffers from a number of limitations and, as a whole, they fail to address issues due to camera positioning, crowded scenes, non-linearity, high dimensionality and complexity.

This thesis specifically addresses the on-line novelty detection problem where labelling information is available only for a small sample of normal patterns. The problem is addressed as unbalanced one-class classification. Given a set of low-level features extracted from video where a part is declared *normal*, the aim is to learn the patterns of usual behaviour (motion) and classify *abnormal* instances based on their statistical properties (rarity). Training under these conditions is prone to over-fitting and the curse of dimensionality [16], which leads to degraded performance. This thesis extends video abnormality detection by pursuing a subspace learning framework based on graph-based dimensionality reduction. The abnormality detection method utilises the low-dimensional subspace to provide motion video abnormality detection while keeping the running and deployment complexity low.

1.2 Problem statement

Our objective is to deploy an on-line novelty detection framework for single camera abnormality detection, with a limited number of parameters to be defined by the user. We consider the abnormality detection problem under the following conditions: *(i)* normal instances are common; *(ii)* no information on abnormal events is available; *(iii)* no a priori model of normality is available; and *(iv)* object tracking data is not available or it is unreliable.

The abnormality detection problem is thus stated as an one-class classification problem. Given a video sequence $\mathcal{P} = \{I_t \mid t = 1, \dots T\}$ composed of a set of images captured over time t, features are extracted so that each frame I_t is represented by a multi-dimensional

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

Figure 1.1: Motion patterns (a) that exist in the labelled samples (first 7500 frames) of the real world sequence S5 and sample events that do not exist in training: (b,c) people entering the platform and (d) person walking in the corridor right to left.

observation feature vector $\mathbf{o}_t \in \mathbb{R}^l$. We form the set $\mathcal{O} = \{\mathbf{o}_t \mid t = 1, \dots T\}$ and a video clip that contains commonly observed action patterns is labelled by the operator as *normal* (i.e. $\mathcal{X} = \{\mathbf{o}_t \mid t = T_1, \dots T_2, \mathbf{o}_t \text{ is } normal\}$ where T_1, T_2 are the boundaries of the annotated video clip). The abnormality detection framework F needs to learn the classes of normal behaviour based on the provided labelled and unlabelled information and then classify a new observation $\mathbf{o}_{\tilde{t}}$ at time $\tilde{t}$. The problem is described as novelty detection that classifies $\mathbf{o}_{\tilde{t}}$ into the *normal* and *abnormal* classes,

$$\mathsf{F}_{\mathcal{O},\mathcal{X}}(\mathbf{o}_{\tilde{t}}) = \begin{cases} 0 & \text{if } \mathbf{o}_{\tilde{t}} \text{ is } normal \\ 1 & \text{if } \mathbf{o}_{\tilde{t}} \text{ is } abnormal \end{cases} . \tag{1.1}$$

In a real-world scenario the operator provides data from a camera and labels a small part as *normal*. Since normal actions are only defined by example, anything that does not comply with the norm is considered abnormal. For example, given a scene of an underground train station and the knowledge that actions that take place in the first 7500 frames are *normal*; actions that are not contained in the training samples should be labelled *abnormal* (Fig. 1.1). However, detailed information about the characteristics, such as type and speed, of the normal or abnormal actions is not provided. Under these conditions abnormality detection is a very challenging task.

1.3 Major contributions

- A graph-based low-dimensional representation method (described in Section 3.3) tuned for video event projection without explicitly using foreground object detection or

object tracking. Given a region of interest (ROI) in the scene, which depicts a single object, we investigate and propose an algorithm that discovers a low-dimensional subspace suitable for abnormal event visualisation. The setup compares favourably in terms of complexity and performance to alternative linear and non-linear techniques in projecting common and uncommon events so that they can be visually identified [J1].

- An off-line abnormality detection framework for local motion video abnormality detection based on a graph-based low-dimensional representation (described in 3.3.4). Using motion vectors and assuming that the detector size (region) is no smaller than the average size of the moving body parts (*e.g.* head and torso), the work compares low-level features and presents the advantages that subspace learning can give to a low complexity novelty detector against alternative more complex classifiers [J2].

- An on-line abnormality detection framework (described in Section 4.7) for motion video abnormality detection utilising a grid of local detectors with similar assumption as in [J2]. The method uses a graph-based unsupervised subspace learning method to allow out-of-sample extension for on-line motion abnormality detectors. The algorithm is compared with an alternative linear subspace based approach under a variety of subspace training conditions using both labelled and unlabelled samples to identify advantages and limitations in real-world scenarios [C1].

1.4 Organisation of the thesis

This thesis is structured as follows: Chapter 2 provides a review of video abnormality detection techniques, related background on subspace learning and recent work in non-linear graph-based methods. It identifies the issues and limitations while highlighting a number of open challenges that exist in the related works. Chapter 3 describes the proposed method for subspace learning and presents the framework for local motion-based video abnormality detection. In particular, the self-tuned neighbourhood graph which allows subspace learning suitable for novelty detection applications is given, followed by the description of the off-line and on-line novelty detectors. Discussion on deployment and parameter selection is also provided. Chapter 4 presents results on off-line and on-line

abnormality detection using single and multi-object datasets. The method is compared with alternative subspace learning algorithms and the use of different low-level features is investigated. Furthermore, the off-line local detector is evaluated against popular novelty detection algorithms. Finally, the on-line framework is compared with alternative linear subspace learning method to establish performance, suitability and training limitations under challenging real-world scenarios. Chapter 5 concludes the thesis and discusses future directions.

Chapter 2

Background

2.1 Introduction

There has been an increased interest in research tackling the video abnormality detection problem. In related works, *Abnormal* or *unusual* events refer to events in the video that differ from the activities that have already been observed. The task is formalised as novelty detection with restrictions on prior knowledge. While the problem of abnormality detection in video sequences has a simple definition, solving the problem is not an easy task. In unconstrained crowded environments with articulated objects, behaviours cannot be easily modelled and unexpected events are rare and, by their nature, cannot necessarily be described fully. In contrast, footage of normal actions is available in abundance. Such restrictions hinder the deployment of novelty detection frameworks in real world scenarios because they eventually require a large amount of detailed annotated data (which is usually expensive and error prone) for model definition and training. Furthermore, the rarity of *abnormal* samples proves to be an important issue in the estimation of optimal working parameters. The generic framework is composed of two separate pipelines (Fig. 2.1). The training pipeline uses the labelling information from the operator and features extracted from the training sequence to model the normal action patterns and train the novelty classifier. The detection pipeline uses a similar feature extraction module to produce a feature vector so that the novelty classifier can label for abnormality based on the previously learnt models.

The next section provides a more detailed description of video abnormality detection

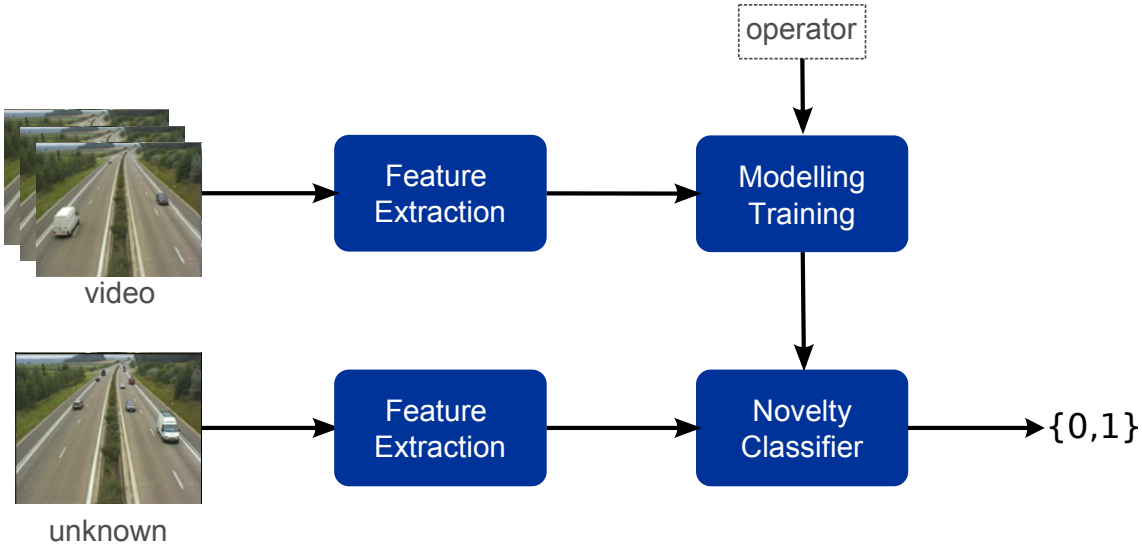


Figure 2.1: Generic framework for video abnormality detection. Features are extracted from the video sequences to model the normal instances (annotated by the operator) and train the novelty classifier to detect the abnormal events.

based on its correspondence with the novelty detection problem and discusses the challenges and the methodologies proposed in the literature to tackle this problem.

2.2 Video abnormality detection

Anomaly or novelty detection is an important signal analysis task that aims to classify data into known (regular) and novel (irregular) classes. These classes are usually defined based on modelling the statistical properties of patterns in complex signals. Chandola et al. [17] provide a comprehensive discussion and review of the generic novelty detection problem and how it is addressed in various research fields. Following their analysis, the main characteristics of the problem are: *(i)* the *nature of data*; *(ii)* the *anomaly type*; *(iii)* the *training labels*; and *(iv)* the *final output*. Using this decomposition we find correspondences and associate the video abnormality detection problem with the generic novelty detection framework.

2.2.1 Anomaly detection

In video abnormality detection, the *nature of the data* (features) can take a wide range of types. These range from high-level information such as trajectories of moving objects in the

scene to blob features (object detection) and even low-level cues such as appearance (pixels) and motion vector fields. This selection provides the first filtering process applied to the video sequence and reduces the types of abnormalities that can be detected by the system. For example, the use of motion vectors would make detection of uncommon clothing (*e.g.* yellow jacket) in a scene impossible. The challenge is to find the best compromise between filtering the noise and keeping the information necessary for describing the model of *normal* behaviour. However, and especially for stream-like data, it is important for the framework to understand and compensate for the temporal correlation between the feature vectors.

According to Chandola et al. [17] there can be three *types of anomalies* in the generic problem of novelty detection: *(i) point*, *(ii) contextual*, and *(iii) collective*. These are defined below, with the addition of *global* and *local* anomalies which are specific to the video abnormality detection problem.

Point anomalies are defined when the feature vector or action pattern does not appear on the training set (*e.g.* a person jumping on the rails of a train, or walking against the flow in a one-way corridor). The majority of related work in abnormality detection solves the problem of point anomalies which is easier to model from the data.

Contextual anomalies are present when the data is only anomalous in a specific context. In surveillance, such a context is usually the time of day (*e.g.* a person running inside a tube station out-of-hours, in contrast to a person running during rush-hour). Contextual anomalies usually require additional information about the structure of the scene and function (*e.g.* traffic-lights) and are specific to the particular application.

Collective anomalies or conditional anomalies are composite anomalies that mainly appear when the data are in sequence and a correlation exists between the vectors in time. The various parts of which they are composed of are not necessarily anomalous but their order is abnormal. In video sequences these anomalies are commonly temporal anomalies (*e.g.* a person waiting for the elevator who does not use it when it arrives). By their definition, collective anomalies describe abnormalities that exist in a variety of temporal scales. However, a feature

transformation of the sequence features (*e.g.* packing in spatio-temporal windows) can transform the problem into point anomaly detection. The side-effect in these cases is that the complexity of the patterns and the dimensionality of the feature vector rapidly increase.

Global and *local anomalies* are defined based on spatial criteria. *Global* anomalies are expressed (or take place) when all the motion in the scene is anomalous (*e.g.* a crowd in panic in a corridor). *Local* anomalies are spatially restricted in a local region. They can be expressed as contextual anomalies when normal behaviour is only inferred based on the region (spatial context) of the scene (*e.g.* normal motion on the street is different from normal motion on the pavement).

The *training labels* of the abnormality detection problem are provided during training in order to infer that a specific instance (video clip) is abnormal. The *labels* could be provided as prior information about the type and structure of the annotated events. However, such high quality annotation is very rare and expensive. In practice, annotation is provided only for a small part of the video, where the operator has evaluated that all the actions taking place in the scene "look normal" without any additional description. Furthermore, there are no guarantees that the provided annotation is comprehensive enough to incorporate the complete corpus of possible behaviours that take place in a scene. The lack of detailed information is the primary reason that related work has been mainly applied in *supervised* or *semi-supervised* modes. The challenge is to maximise the *transfer of information* from the feature vectors to the model utilising both the provided *labelled* and *unlabelled* samples.

Finally, *the output* of an abnormality detection method is commonly a score (scalar value) that measures abnormality. However, the application requires a binary label, thus the abnormality score is thresholded in to the abnormality label. Given the limited information available, abnormality detection cannot provide clear semantic labels of activity (*i.e.* actions are not allocated an activity tag such as jumping or running). This point clearly separates these frameworks from works that aim at specific event detection and are expected to provide a more detailed class label as an output.

The following subsections discuss related work in video abnormality detection. As presented in Table 2.1, video abnormality detection methods use a large variety of tools to model the patterns that exist in the training feature vectors. This proliferation of methods

makes categorisation and comparison of the abnormality detection frameworks a difficult task, yet some aspects of the methods can be compared and criticised. The methods are grouped based on the type of abnormality they detect (*global* or *local*) and the features they use to achieve the task (*i.e.* trajectories, object detections, and low-level features). This taxonomy is followed, in this chapter from high-level towards low-level features in order to provide an overview and critique of related work that motivates our proposed framework for addressing the identified challenges.

2.2.2 Trajectory-based methods

Trajectories are produced by detecting moving objects in the scene and tracking them over time. The majority of the methods that use object tracking are suitable for both global and local abnormality detection. Modelling of global and local behaviour is performed hierarchically. Feature vectors are extracted from normal trajectories to model the local behaviour of the moving objects in a local region. Global patterns are modelled based on the learnt local behaviour models.

Johnson and Hogg [2] suggest a two-layer system to model normal actions. They convert trajectories to a sequence of 4-D vectors describing the position and speed of an object at regular spatio-temporal intervals. Modelling involves a set of neural networks (in sequential topology) to perform vector quantisation (VQ), define prototypes and learn the spatial and temporal distributions of the sequence of the feature vectors. New trajectories are characterised as abnormal based on the learnt probability densities. A similar modelling method is proposed by Mecocci et al. [3] who use a 5-D feature vector by adding the object size/perimeter ratio.

Basharat et al. [11] use a 5-D feature vector describing time, location and size of the associated object for every pixel in the normal trajectories. Their modelling is based on Gaussian Mixture Models (GMMs). Each GMM is associated with every pixel in the image and learn the possible transitions (*i.e.* paths) that can take place based on the normal trajectories. Local and global abnormality is inferred by finding the probability of an observed object transitions. Sillito and Fisher [12] propose a semi-supervised on-line learning method (also based on GMM), where a novelty classifier is incrementally updated to detect abnormality based on the feedback of the human operator. This method is only

	Tools	Local/Global	Reference
Trajectories	Neural Net	local/global	Johnson and Hogg [2]
			Mecocci et al. [3]
	HMM models	local/global	Chan et al. [7]
			Hu et al. [9]
			Jiang et al. [8]
		global	Izo and Grimson [5]
	Spectral clustering	global	Porikli and Haga [6]
	Trees	local/global	Piciarelli and Foresti [4]
	Mix. of Gaussians	local/global	Basharat et al. [11]
			Salas et al. [10]
		global	Sillito and Fisher [12]
Objects	MOHMMs	global	Xiang and Gong [18, 19, 20]
	Adapted HMMs		Zhang et al. [21]
	Tracking		Cui et al. [22]
	SVM		Sudo et al. [23]
	LSA	local	Li et al. [24]
	Concurrence Matrix	local	Li et al. [25]
	Histogram		Russell and Gong [26, 27]
	Spectral clustering	global	Zhong et al. [28]
			Hamid et al. [29]
Appearance	infinite-HMMs	local/global	Pruteanu-Malinici and Carin [30]
	HMMs		Kratz and Nishino [31]
	HMMs		Ermis et al. [32]
	Database search		Boiman and Irani [33]
Motion	MOHMMs	global	Andrade et al. [15]
	Histograms	local/global	Adam et al. [13]
	MRF		Kim and Grauman [14]

Table 2.1: Related work in video abnormality detection. (Key: HMM - Hidden Markov Model, MOHMM - Multi Observation HMM, SVM - Support Vector Machine, LSA - Latent Semantic Analysis, and MRF - Markov Random Field)

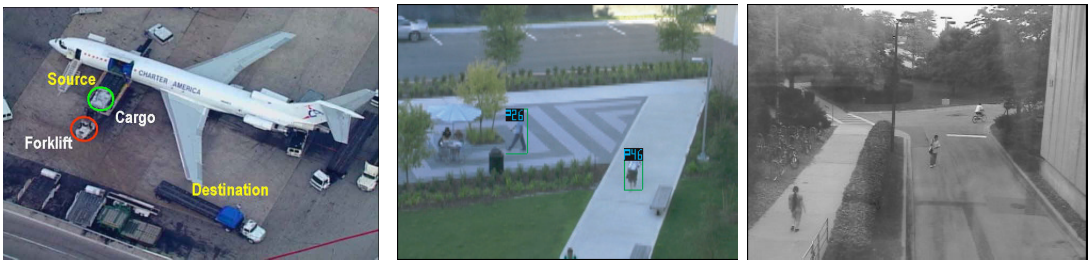
able to detect global abnormality since it represents the complete trajectory with a single feature vector, containing the coefficient values of a polynomial fitted on the trajectories.

While most of the trajectory-based methods assume that the trajectories are already available, Hu et al. [9] propose a complete system to extract and cluster trajectories in a hierarchical way, and detect abnormal events. In their work, the trajectories are re-sampled and normalised to the same length and clustered using the fuzzy k-means algorithm. The results allow the original set of trajectories to be separated into spatial classes of intermediate trajectories. A second layer of clustering takes place on each spatial class using temporal information to group the main motion patterns in the scene. A chain of Gaussians is trained on each pattern. Anomaly detection is based on the probability that a new trajectory belongs to one of the learnt patterns. Hidden Markov Models (HMMs) [34] were used to learn the main trajectory classes in [8]; where they are paired with hierarchical clustering to automatically find the number of states and learn the dominant paths in the scene. Hu et al. [9] present results on a sparsely occupied scene of pedestrians moving at a distance. For simple scenes in a stationary environment, such as an airport cargo terminal, tracks can be converted into semantic primitives and HMMs can be trained in order to detect rare events [7].

Hybrid approaches combine HMMs with spectral clustering. Porikli and Haga [6] train one HMM per trajectory (modelling position, direction and speed) to define a similarity measure between trajectories that is independent of their length. Using the HMM model similarity, Porikli and Haga propose a spectral clustering method based on conformity scores, that provides unsupervised off-line abnormal trajectory detection. More recently, Izo and Grimson [5] use spectral clustering in an iterative way so that on-line abnormality detection is possible. The similarity function in this case is defined based on a weighted sum of trajectory attributes. They adopt the Normalised Cuts [35] algorithm and cluster the data in a low-dimensional representation. New instances are embedded in the low dimensional space based on the Nyström approximation. However, in order to reduce the computational cost the method uses a smaller sample set (landmarks) to find the out-of-sample extension on the embedding space. After the embedding, new instances are validated against the learnt clusters for abnormality.

A different approach uses tree structures to represent and cluster trajectories [4]. Each

(a)



(b) (c)

Figure 2.2: Examples of scenes with low probability of dynamic occlusions where trajectory-based methods present results: (a) distant rigid objects [6, 9, 5], (b) top view [7], and (c) low object density [11, 8]. (Note: images extracted from [6, 9, 5, 7, 11, 8])

node describes a cluster of similar local paths. During training the tree is populated and clusters the normal patterns that exist in the trajectories. Given the information in the tree, "abnormal" are trajectories that have low probability. When additional information about the scene is available, Salas et al. [10] suggest using a combination of appearance and motion features to track rigid objects at crossroads where vehicle movement is controlled by traffic lights. Based on the status of the controller of the traffic light system, contextual abnormality is detected (e.g. wrong turns and red light violation) when a track follows a path is not expected during that phase of the system.

Research that relies on trajectories has been restricted in scenes which have a small number of moving objects (Fig. 2.2). Such a restriction is mainly due to the low performance of object association under crowded and low-contrast conditions. Such conditions cause misdetections due to partial and dynamic occlusions and thus tracking is unreliable [36]. Furthermore, since tracking is commonly supported by a number of modules (*e.g.* background subtraction, object detection), the modelling is very sensitive to errors that appear earlier in the pipeline.

In terms of computational complexity, the trajectory based methods, utilising HMMs

and neural networks are fast and have been known to work in real time (given that the cost of multi-object tracking is not included). However, since the complexity of trajectories in crowded scenes is high, the need for extended amounts of annotated training samples increases. As a result, the deployment cost of the framework is prohibitive and requires numerous error-free manually selected trajectories for training to avoid degradation of the model. Spectral methods are less sensitive to noisy training data but their computational complexity is high. Thus they are more suitable for off-line detection.

The next subsection discusses methods that avoid the tracking and association modules and directly use output from the object detection module.

2.2.3 Object-based methods

Several approaches attempt to detect abnormal events using features extracted by the object detection module. We consider object-based features as mid-level since there is a good amount of post-processing (often involving heuristics) that takes place before the individual object blobs can be defined. Only point abnormalities are detected since there is no temporal information in the object detection. It is however common to add and pack together observations over time intervals, to incorporate in the extracted features the temporal evolution of the object's motion in the scene. Such feature "packing" allows collective abnormalities that conform to a specific time scale to be discovered.

Xiang and Gong [18] use object detection to extract Pixel Change History maps (PCH) and represent each blob as a 7-D vector. The feature vectors are clustered into *event instances* based on fitting a GMM over the training features. The video is divided into clips that are detected based on frames of inactivity. The *events* that take place in each clip compose an *action pattern set*. Each *pattern set* is then modelled by an adapted Multi-Observation Hidden Markov Models (MOHMM) to provide a similarity measure with which they can be clustered into classes. Using the class information the action pattern sets are now re-modelled using a Mixture of MOHMMs for each class. The process is unsupervised since it automatically assigns the abnormal label to the classes that have a low membership count. The framework also allows for the model to be adapted on-line as more data become available. It is demonstrated on indoor sequences with low crowd density, but is only able to detect global abnormal events. In an alternative implementation [20, 19], they suggest an

on-line algorithm to the sequence that does not relay on frames of inactivity. Furthermore, they describe a method to cluster the *action pattern sets* using an eigen-decomposition. In addition to the indoor scene used in [18], results are also demonstrated on a scene from an aircraft ramp area where small rigid objects are in motion.

Cui et al. [22] also choose to extract change-relevant pixel statistical features. They use Pixel Change Frequency (PCF) and Pixel Change Retainment (PCR), which describe each pixel's change frequency and duration. The detections are grouped together over a temporal window to produce a set of visual events (blobs) and luminance histograms are used to describe each blob. Using additional knowledge about the normal event classes the visual events are separated and preprocessed by Principal Component Analysis (PCA) [37] to produce a low-dimensional subspace for each class. The classes of events are then modelled using an extended Sequential Monte Carlo tracker framework which tracks the sequence of classes (events) based on probabilistic manifolds. Abnormality is evaluated based on the Expected Log Likelihood (ELL) of the current state provided by the tracker.

Zhang et al. [21] propose a semi-supervised incremental learning algorithm that uses object features packed over regular time intervals. The high-dimensionality of the feature vectors is again addressed by PCA. Modelling is based on iterative adapted HMMs (iHMMs) that are initially trained on labelled normal samples. These models are then incrementally adapted based on the unlabelled training samples. This semi-supervised approach allows better initialisation of the pattern modules, since the classes are provided by the operator. However, the use of PCA places a number of restrictions on the characteristics of the feature vectors (see section 2.3).

A linear subspace is also utilised by Sudo et al. [23] and is coupled with an incremental SVM [38] novelty classifier to detect abnormality from change detection features. Features are packed using a sliding window and subspace learning is performed through an incremental approach (iPCA) which adapts the subspace based on the new samples. The one-class SVM is incrementally updated, using the new principal components, to detect global abnormal events. Unfortunately classifiers such as the SVM are based on the assumption that normal and abnormal sets have similar sizes, which conflicts with the fact that the problem of abnormality detection is by definition unbalanced and abnormal events are rare.

For batch-mode (off-line) unsupervised detection of uncommon events, Zhong et al. [28] extract object features from non-overlapping clips from a video sequence in a high-dimensional space and co-embed them, into a low-dimensional space, with prototypes provided by k-means. Detection requires an additional clustering step to find the classes with total low inter-cluster similarity and to label them as abnormal. When contextual information is available, object detections can be directly converted to semantic tags and the video abnormality problem is transformed into identifying uncommon sequences of tags. An example application is presented by Hamid et al. [29], who use spectral clustering on event-motifs (semantic tags) corresponding to object detections in a utility room (kitchen).

Contextual local abnormality is detected by Li et al. [25, 24], who cluster blob features from short clips and use the learnt classes to the scene into regions of similar activity. For each clip the prototypes of *atomic blob events* are discovered (k-means). The *atomic events* in the clips are further clustered to define global behavioural patterns. Regions where the same behaviour exists are defined using image segmentation applied on feature vectors describing the patterns that take place at each pixel position. The most relevant blob features in each region are selected based on entropy and the new feature space is clustered to acquire regional behavioural patterns. Modelling is based on hierarchical probabilistic Latent Semantic Analysis (pLSA) [24]. The first layer models the local (region based) topics and the second layer learns the global topics based on the models of the first layer. However, the method is restricted to scenes with a clear phase structure in the regions (*e.g.* traffic light junctions) which is required to be known à priori. A method that can be applied in generic scenes is given in [25], where the regional and global behavioural patterns on the training set are modelled using a concurrence matrix. The abnormality label is then inferred by identifying unexpected pairs of *atomic events*.

In similar scenarios with clear phase structure, Russell and Gong [26, 27] propose on-line methods based on discovering the fundamental frequency of the feature change cycle over a spatio-temporal grid. In [27] the aspect ratio of the detected objects is used to form a spatio-temporal histogram for each spatio-temporal volume. The *fundamental frequency* (cycle) of each spatial grid position is discovered using the training clips. Next, the local activity is modelled by an average spatio-temporal histogram of the features within the cycle. The model is updated on-line to keep track of the *fundamental frequency* changes.

(a) (b) (c)

Figure 2.3: Example of scenes where object-based method are able to detect *global* events under perspective distortion: (a) Corridor [18, 19] , (b) Poker game [21, 28] and (c) nursing home [21]. (Note: images extracted from [18, 19, 21, 28])

Abnormal objects are detected when their aspect ratio does not conform to the learnt model. In [26] the feature space is replaced by a 5-D histogram (*i.e.* 2-D optical flow + 3-D colour intensity) extracted from detected objects to represent a multi-dimensional signal. The model update is controlled by a Phase Loop Locked (PLL) topology which removes jitter noise in the estimation of the fundamental frequency.

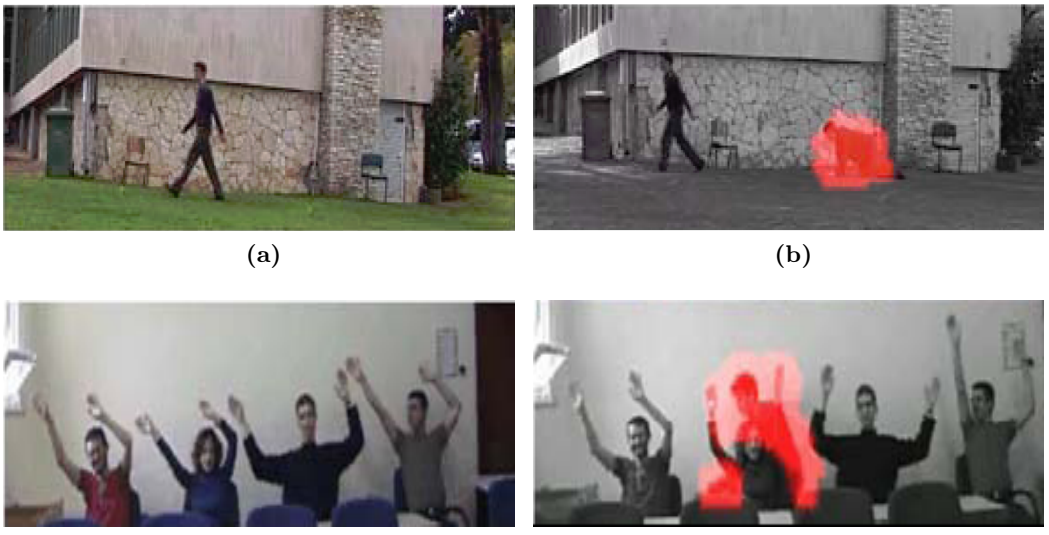
In contrast to the trajectory-based approaches, object-detection based methods present results in indoor scenes with perspective distortion (Fig. 2.3). However, under heavy perspective and crowded scenarios (*e.g.* underground stations, airports) the probability of dynamic occlusions is very high and thus the important task of object separation is very challenging. Yet, without proper object separation inferring abnormality on a local region is difficult. This is the primary reason why a number of methods that provide local abnormality detection [26, 24, 25, 27, 29] are demonstrated on sequences where object separation is more reliable (Fig. 2.4).

2.2.4 Low-level feature-based methods

In crowded scenes the interactions among multiple objects moving in an unrestricted manner create complex action patterns. Several actions take place at the same time and as a result spatially local abnormal instances are difficult to identify. In crowded scenes object detection cannot separate the different moving objects and low-level features are preferred to describe the behaviour patterns. Common features used for modelling are appearance, pixel change and motion related features.

(a) **(b)**

Figure 2.4: Examples of scene that allow local abnormality detection in object-based frameworks: (a) top view [29] and (b) rigid distant objects [26, 24, 25, 27]. (Note: images extracted from the corresponding publication)



(a) **(b)**

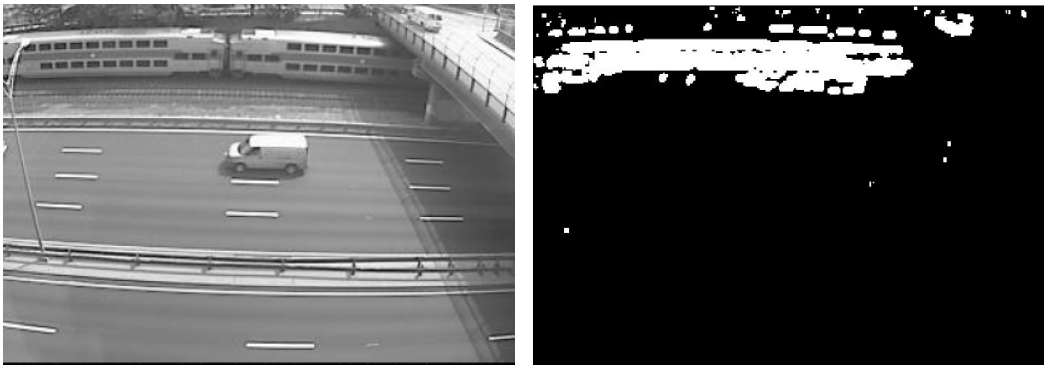
(c) **(d)**

Figure 2.5: Sample frames from the work of Boiman and Irani [33], applied in scenes with low perspective distortion. (left column) normal instances and (right column) detection of regions where action is not present in the database. (Note: images extracted from [33])

Using appearance Boiman and Irani [33] extract numerous spatio-temporal ensembles (cuboids), overlaid on each frame. The ensemble is divided into multiple small patches in various scales and a local descriptor is calculated from each patch. The ensemble descriptors along with their position relative to the centre of the ensemble provide a database of normal patterns. Abnormal events are detected when they cannot be inferred from the existing patterns in the database. While local abnormality is detected the global nature of the method will not describe in a specific manner the patterns that are dominant (normal) in the various regions of the scene. Such a setup assumes that spatially local actions are global, meaning that an action is normal or abnormal independent of the position it appears in the scene. The method is demonstrated in scenes with no perspective distortion (Fig. 2.5) and is able to detect local abnormalities in a variety of crowd densities. The sophisticated feature descriptor used requires a large number of manually set parameters and calculation over multiple temporal scales to capture behaviour reliably. Thus, due to the high complexity and memory requirements (over 100,000 descriptors per minute of video), it does not scale well to larger training samples.

While appearance based features are easy to acquire, their use in video abnormality detection is problematic. A common issue raised is due to the diversity in the appearance of the objects. It is common to have objects with different colours perform the same action or behaviour. As a result, when we need to represent a specific behaviour in a comprehensive way, we need to have examples of all possible colour appearances of the objects. Furthermore, appearance is very sensitive to illumination changes. Conversion to grayscale might help reduce the effect but still it is expected that the high variability in appearance will cause problems in modelling the normal patterns for novelty detection.

To reduce the dependency on appearance, features based on change detection are utilised. Ermis et al. [32] use the change rate in each pixel, captured over a fixed size temporal window, as a one-dimensional time series. The novel idea is to consider the normal change rate as a *static* motion pattern for each pixel. Novelty detection is then addressed as background subtraction where the *static* pattern is treated as the background and the detections are the abnormal instances. Low-dimensional representations of change rate patterns from all the pixels in the scene are provided by *dimensionality reduction* based on random projections and then are clustered using K-means. Finally, a histogram is constructed to

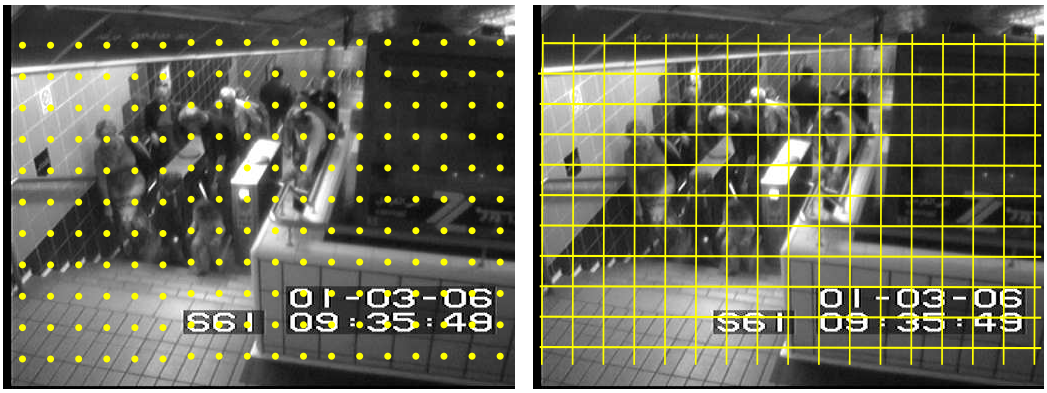


(a) **(b)**

Figure 2.6: Sample frames from the work of Ermis et al. [32], applied in scenes with rigid objects and distant cameras: (a) Original view, (b) output by removing pixels with normal change rate. (Note: images extracted from [32])

model each discovered normal class. The likelihood of the observed change rate in a pixel labels the change pattern for abnormality. Using the change rate of the pixels provides a degree of geometrical independence under perspective distortion and thus allows matching of behaviours in multi-camera scenarios [39]. However, the pixel change rates will not be able to properly describe instances for non-rigid objects and under dynamic occlusions. Furthermore, pixels are assumed to be independent or to only have linear correlation thus modelling is not expected to cope well with articulated objects. Therefore, they have been applied in scenes where the cameras are positioned far away from the moving objects and at angles that reduce dynamic occlusions (Fig. 2.6). The modelling methodology also suffers from the assumption that the change patterns are spatially universal (global) similar to the work of Boiman and Irani [33].

Another popular feature is motion, which provides a more reliable description against partial occlusions and is less sensitive to appearance changes. Andrade et al. propose a hybrid method that uses both change detection and optical flow. In their work [15], the video sequence is divided into clips and motion feature vectors are extracted from each frame. For each video clip, a MOHMM is trained on a low-dimensional representation produced by PCA. The video grouped are grouped through spectral clustering based on the similarity measure between the learnt models. A new set of MOHMMs is trained on the samples from each class. Detection is provided by thresholding the observation's likelihood against the learnt bank of models. The method assumes that the training data do not have instances of inactivity and considers only global abnormality detection. Furthermore,



(a) (b)

Figure 2.7: Multi-detector based setups for local abnormality detection. Example deployment of (a) single pixel motion based detectors [13] and (b) region based detectors [14].

results suggest that in real-world scenarios parameter selection is challenging.

To overcome complexity and better represent the spatial dependency of the action patterns, frameworks use simple low-level feature descriptors and introduce the concept of the *local detector*. The detector in this case is permanently associated with a spatial region of the scene. In the literature, local detectors are associated to a pixel or a specific region (Fig. 2.7a). Adam et al. [13] use a grid of pixel based local detectors. They estimate the motion vector on single pixels while taking into account the optical flow in the local pixel neighbourhood. The measurement from each point is evaluated against the histogram of previous results and ranked based on the probability of occurrence. The abnormality label for the frame is obtained through a voting schema of local detections over a number of previous frames. The concept of independent detectors allows for *pure* local abnormality detection but modelling can only consider one feature at a time (angle or magnitude). The approach has a low complexity, but abnormality is evaluated based on recent (temporal) motion history and thus detected abnormal events might not be consistent (*i.e.*, what is abnormal at a certain time instant might not be abnormal in the future). However, this memory characteristic makes the detection of abnormal contextual events (time of day) possible.

Another example of a multi-detector framework is presented in Pruteanu-Malinici and Carin [30]. The scene is divided into a grid of non-overlapping regions (Fig. 2.7b) and each detector is trained on appearance based features. The feature space is construc-

ted using Invariant Subspace Analysis (ISA) [40] which finds a low-dimensional subspace transformation and train infinite-HMMs for each block in the grid.

In a similar setup, Kim and Grauman [14] use optical flow calculated over different scales and construct a space-time Markov Random Field (MRF) [41] to infer abnormalities. The scene is divided into non-overlapping regions which themselves are further divided in to smaller sub-regions. The feature vector for each region is constructed by concatenating the set of 9-dimensional vectors (angle histogram plus speed) calculated over the sub-regions. The features in the training sample are clustered into "words" using Mixture of Probabilistic Principal Components Analysis (MPPCA) [42]. Detection takes place by constructing a space-time Markov Random Field (MRF) over the last 10 frames and finding the labelling that maximises the Maximum a Posteriori Probability (MAP). The MRF is constructed based on histograms of the discovered "words" to describe the typical local activities and their interaction. A frequency histogram of "words" (local activity) is associated with each node, and a co-occurrence histogram of "words" (activity interaction) is associated with each link of the MRF. The framework can be used in off-line (batch) mode. On-line application is possible, assuming that the posterior probability of each component is constant so that the MPPCA components can be updated with a small complexity cost. The detection results have improved compared to the work of [13], since it also utilises spatial restrictions to infer the abnormality labelling and a local method MPPCA to represent motion patterns. However, this modelling also suffers from the assumption that these patterns are universal in the scene (similar to Ermis et al. [32], Boiman and Irani [33]). Such an assumption is restrictive and undermines the concept of *local* abnormality.

Spatio-temporal restrictions among local detectors are also suggested in the work of Kratz and Nishino [31]. The scene is divided into a grid and spatio-temporal cuboids are extracted from each region. An action signature is constructed by statistically modelling the spatio-temporal gradients of pixels in the cuboids using a multi-variate Gaussian. Assuming that the local motion patterns do not deviate much across the scene, the signatures extracted from the complete video volume are clustered based on the symmetric Kullback-Leibler divergence [43] to separate the signatures in classes and find their prototypes. Each cuboid can now be represented as a distribution of prototypical signatures. Detection of abnormal instances is based on modelling the spatio-temporal dynamics of

the distribution of the prototypical classes of signatures that are associated with a specific location. The temporal dynamics are captured by training HMMs for each local region and the spatial correlation of the cuboids is captured using coupled HMMs on the adjacent regions. To infer the abnormality label for the scene, the confidence values of the spatial and temporal models are fused (weighted-sum). The method is demonstrated on extremely crowded scenes, yet a number of important parameters are fine tuned (*e.g.* the weights for the spatio-temporal abnormality ranking) based on the ground-truth.

Based on recent work *motion*, is more reliable for abnormality detection in crowded scenes with dynamic occlusions [13, 14, 31]. These methods consider a number of detectors associated with a specific region to provide local temporal abnormality detection. The results are later fused to further infer spatio-temporal abnormality in a hierarchical fashion. Results favour the use of mid-sized local regions per detector to capture the motion patterns of articulated objects [14, 31]. Nevertheless, key aspects of the novelty detection problem regarding the feature space properties and framework complexity have not been investigated in a comprehensive way. The next subsection will highlight these limitations and the challenges that exist in the literature of video abnormality detection.

2.2.5 Limitations and challenges

So far, in the literature, the focus is to identify appropriate and reliable features that describe the motion patterns in the scene (feature extraction module) and, based on these features, define a modelling strategy that can provide spatio-temporal abnormality detection for both local and global instances. An important aspect of the problem is how this *transfer of information* from the feature vectors to the model takes place as well as the cost of the practical application of the abnormality detection framework in real-world scenarios.

Practical application is greatly affected by the computational complexity. During training one can accept some delay, but real-world solutions demand real-time performance during on-line abnormality detection. Thus complex algorithms that require high computational power and a large amount of memory are not practical [33]. Alternatively, simple (low complexity) frameworks [13, 14, 31, 15] are more appropriate.

However, in most methods it is common to ignore the deployment complexity of a framework. This complexity is related to the cost of setting up, training and fine tuning

the various parameters of the work. In the literature these parameters are usually hidden and not fully explained (*e.g.* the number and size of the bins in the history histogram in [13]) or they are estimated retrospectively from the results and the ground truth [31]. It is usually considered that methods that have more user-defined parameters can be better fitted to a variety of problems. But this belief can be supported only when: *(i)* the annotated training data are in abundance; *(ii)* there is a clear connection between the parameters and their effects and; *(iii)* abnormal instances are available during training.

It can be argued, that custom made modelling strategies can be designed, when the operator provides additional information about the scene, such as: detailed annotation (*e.g.* separation of training into classes); contextual information; or already known abnormal events. However, the cost of such annotation is very high and hence such annotation is not always available. Furthermore, there is the possibility that this information is biased and does not actually represent the true structure of the patterns unless it is carefully selected and verified.

In real-world scenarios of abnormality detection, the limited amount of information about the normal patterns and lack of annotated abnormal instances during the training phase places limitations on the parameter estimation. Optimisation through methods of cross-validation is also not a practical solution for complex frameworks whose pipeline requires a plethora of manually tuned parameters in order to provide acceptable performance. Therefore, it is of interest to design frameworks that can work using a limited amount of annotation and extract all the possible information directly from labelled and unlabelled data.

The ability to work under such restrictions favours methods that focus on maximising the *transfer of information*. Yet in related work this process is usually hindered by weak assumptions on the nature and characteristics of the feature vectors. Most frameworks ignore issues that arise when manipulating high-dimensional data. Processing of such high-dimensional feature spaces is inherently difficult due to the *curse of dimensionality* [16]. The feature space is sparsely populated by the training vectors and pattern recognition algorithms fail to perform.

Furthermore, in the vast space of mathematically possible vectors that correspond to each feature space, those that are physically possible are only a small fraction. These

vectors are embedded into a low-dimensional manifold that is defined by the correlation between the free parameters of the system. Biological systems are known to exploit this correlation, which in the general case is *non-linear* [44].

Finally, non-linear dynamics can be introduced through the post-processing steps performed on the feature vectors, yet most methods are oblivious to the serious side-effects these steps can have on the data. For example a number of works in the literature, group or pack the observations over a number of frames (video clips) in order to construct feature vectors that capture the temporal evolution of the actions. Unfortunately, such a process changes the dynamics of the feature space in such as a way that clustering algorithms fail to capture the real pattern information. This is demonstrated in the work of Lin et al. [45]. Their experiments on clustering and learning patterns from time series data (stock market values) has shown that the results are the same and are independent of which dataset is used for training (even a random time-series gives similar results). The paradox is revealed when it is shown that sliding window sampling filters the patterns and forces them lie within an ellipse [46].

It is argued that if these challenges are not considered, no modelling can be expected to reach its full potential. Maximising the *transfer of information* is thus of equal importance to the modelling strategy. To this extent, subspace learning methods are usually employed to provide a low dimensional subspace which allows for better representation of the data and to compress information removing redundancy and noise. Furthermore, the possibility of using both annotated and non-annotated data can alleviate the need for detailed and costly information provided by the operator. However, in related work [23, 21, 22, 14, 30, 15], when a subspace learning algorithm is used: *(i)* this is performed assuming a linear correlation between the feature vectors (*e.g.* PCA); and *(ii)* the main focus is on reducing the complexity of the framework and not the *transfer of information*.

This section has presented and discussed recent work on the problem of video abnormality detection. The problem is addressed as novelty detection where the *normal* instances are in abundance and *abnormal* instances are rare. The methods are grouped into categories based on the features they utilise to infer abnormality and are discussed to identify their positive and negative points. In related work, *trajectory-based* methods attempt to exploit the high-quality information that is included in an object trajectory and detect both *local*

and *global* abnormal instances. However, in challenging, unrestricted, real-world scenarios, trajectories are not always reliable. To compensate, recent work uses mid-level features from object-detection blobs. Unfortunately, these methods still suffer problems when the scene is crowded and the probability of dynamic occlusions is high. Additionally, using object detection features makes *local* abnormality detection a very difficult task. To this extent, the current trend is to use low-level features, yet, a number of issues regarding the deployment complexity and the *transfer of information* have not been fully addressed. It is argued, that subspace learning can provide a path to address these issues thus maximising the *transfer of information.* The next section presents a brief review of commonly used subspace learning methods identifying the characteristics, advantages and limitations that their application involves.

2.3 Dimensionality reduction

The term *dimensionality reduction* is commonly used in the literature [47, 48, 49] to describe methods that aim to learn, from high-dimensional samples, a low-dimensional subspace wherein specific statistical properties can be well preserved. All methods assume that, due to redundancy and correlation between the feature dimensions, the feature vectors lie close to a low-dimensional manifold embedded in the original high-dimensional space.

Given the feature vector $\mathbf{o}_t \in \mathbb{R}^l$, the aim is to produce $\mathbf{y}_t \in \mathbb{R}^m$ that represents $\mathbf{o}_t$ into a low-dimensional space (*i.e.* $m \ll l$). Then dimensionality reduction is the mapping function D,

$$\mathbf{y}_t = \mathsf{D}(\mathbf{o}_t). \tag{2.1}$$

When the mapping can be expressed as a linear combination of the basis vectors the projection can be expressed as,

$$\mathbf{y}_t = \mathsf{D}(\mathbf{o}_t) = A^\top \mathbf{o}_t, \tag{2.2}$$

where A is an $l \times m$ matrix that provides the basis transformation from the original high-dimensional feature space to a low-dimensional subspace.

Dimensionality reduction methods are commonly divided into two categories: *linear*

	Non-linear	Global/Local	Criteria	Ref.
PCA	-	+/-	(max) variance	Shlens [37]
ICA	-	-/+	statistical independence	Comon [50]
MDS	-	+/-	(min) metric dis.	Cox and Cox [51]
Isomap	yes	+/-	(min) geodesic dis.	Tenenbaum et al. [48]
LLE	yes	+/+	neighbourhood graph	Roweis and Saul [49]
LE	yes	+/+	neighbourhood graph	Belkin and Niyogi [47]
MVU	yes	+/+	neighbourhood graph	Saul et al. [52]

Table 2.2: *Dimensionality reduction* methods and their main properties.

and *non-linear*. Linear methods can be expressed as a linear mapping (Eq. 2.2). Non-linear methods can be only expressed using Eq. 2.1. Apart from this categorisation, the algorithms mainly differ on the criteria they use to find the optimal projection (Table 2.2). These criteria enhance or suppress various characteristics and dynamics of the input feature vectors.

2.3.1 Linear methods

A common practice to compress and preprocesses the feature space is to use *linear dimensionality reduction*. A well-known method is Principal Components Analysis (PCA) expects that variables in the high-dimensional feature vectors are linearly correlated (*i.e.* there exist only first order dependence between the variables) and that the feature patterns follow approximately Gaussian distributions [37]. The PCA method aims at discovering a subspace projection (Eq. 2.2) that maximises the variance. Therefore, PCA will not be able to properly represent feature vectors where maximum variance is not important while being very sensitive to outliers.

In some cases where the number of dimensions is high and the complexity of solving the PCA eigen-problem increases, it is easier to compute the projection matrix using Multi-Dimensional Scaling (MDS) [51]. MDS uses a dissimilarity matrix between the feature vectors of the original feature vectors $\mathcal{O}$ to construct a low-dimensional representation $\mathcal{Y}$ that preserves the pair-wise distances. When the dissimilarity matrix is calculated based on the Euclidean distance, results are equivalent with PCA.

Another linear method is Independent Components Analysis (ICA) [50], which attempts to discover a linear transformation of a random vector that minimises the statistical

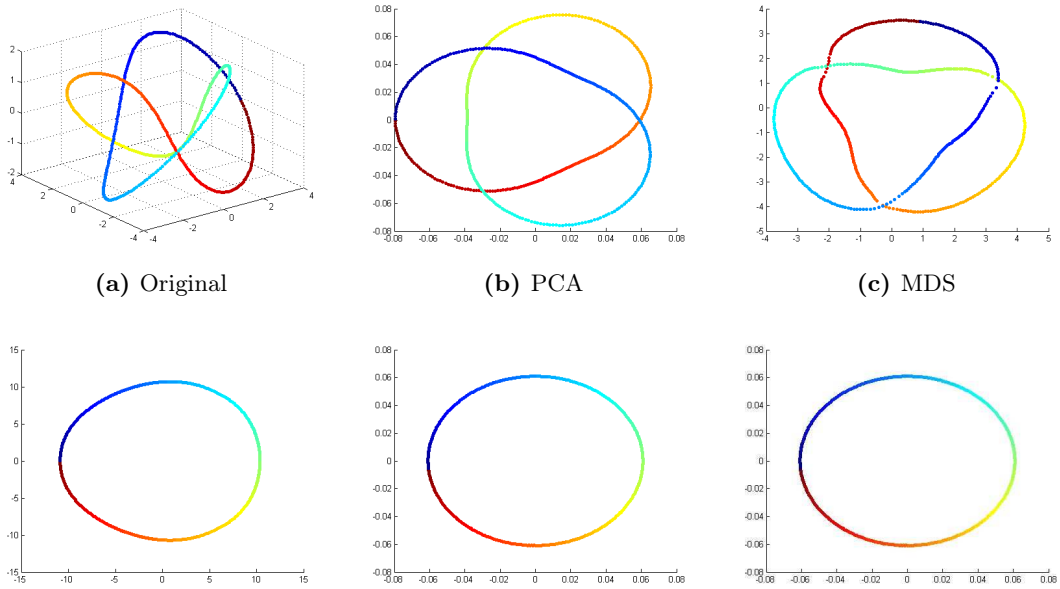
dependence between its components and can cope with patterns that follow non-Gaussian distributions. However ICA's performance is poor when the feature patterns actually follow a close-to-Gaussian distribution. Draper et al. [53] compare these two methods on the classification problem of face recognition where they conclude that based on the task, PCA is more suitable for global tasks (like face recognition) compared to ICA which is better suited for tasks that involve local patterns like expression recognition.

2.3.2 Non-linear methods

When feature vectors are not linearly correlated the mapping can be produced by non-linear dimensionality reduction algorithms which rely on metrics defined on a neighbourhood graph. Based on these metrics and under certain constraints, the mapping (Eq. 2.1) is produced by solving an eigenvalue problem to find the solution that minimises the projection error.

A popular method is Isomap [48], an extension to MDS where the distance metric is the geodesic distance defined on a neighbourhood graph constructed from the input data. Using this metric, Isomap can achieve adequate results on mapping data that lay on manifolds that have a non-linear global structure. Another global non-linear dimensionality reduction algorithm is Maximum Variance Unfolding (MVU) [52] (previously known as Semi-Definite Embedding), which solves the optimisation problem that maximises the distance between the unconnected nodes in a neighbourhood graph while preserving the distances along the edges and the angles between the edges. This optimisation problem is solved using Semi-Definite Programming (SDP) [54]. Locally Linear Embedding (LLE) [49] and Laplacian Eigenmaps (LE) [47] use neighbourhood graphs to embed data accounting for the local data structure around each point in the high dimensional space. Specifically, LE is based on the commute times between the graph nodes, which take into account all the paths from one node to another, and not just the shortest path, thus preserving local structure.

An example is presented in Fig. 2.8. The data consist of 539 points [55] sampled from a trefoil knot in three dimensions. The underlying manifold is an one-dimensional closed loop and is projected in two dimensions. The application of LE, Isomap and MVU produce the correct projection. On the other hand, PCA and MDS fail to account for non-linear correlation between the features in the input vectors.



(a) Original **(b)** PCA **(c)** MDS

(d) Isomap **(e)** MVU **(f)** LE

Figure 2.8: Example of *dimensionality reduction* of the trefoil manifold (a), from three dimensions to two, using linear (b,c), global non-linear (d,e) and local non-linear (f) dimensionality reduction algorithms.

However, Balasubramanian and Schwartz [56] claim that graph-based methods cannot be applied except for specific synthetic examples. More recently, Van der Maaten and Hinton [57] also challenge the performance of graph-based dimensionality reduction methods. The main critique is that they are unstable and cannot provide meaningful projections of real-world datasets. These issues exist because, in the non-linear dimensionality reduction literature [47, 48, 49] there exist a subtle assumption that the neighbourhood graph is approximating the assumed manifold structure where the feature points exist. Nevertheless, while the graph embedding problem is addressed by a number of methods, the graph construction problem has not yet been fully investigated.

2.3.3 Neighbourhood graphs

Commonly used graphs are the k and ε nearest neighbours (NN). The k-NN graph is based on the rule that each node is connected to at least k neighbourhood (closest) nodes sorted by a similarity measure (usually the Euclidean distance between vectors). As a set of connection rules node i is connected to node j if node j is among the k closest

neighbours of i or node i is among the k closest neighbours of j. The main advantages are that it usually provides a sparsely connected graph and the numerical eigen-solver executes faster. Alternatively, the rules for the ε-nearest neighbour graph dictate that we form a hypothetical hypersphere, with radius ε around each point and connect only those vectors that exist inside it. Thus node i is connected to node j if $d_{ij} \leq \varepsilon$ (where d_{ij} is the similarity between nodes i and j). This set of rules provides a good approximation of the relative neighbourhood around each node which is more intuitive from a geometrical perspective. Unfortunately, it is sensitive to the choice of parameters and therefore requires a careful selection to avoid distorted mappings.

Some research to improve the graph construction, and thus graph-based dimensionality reduction, tries to automate the process of selecting the neighbourhood size. Samko et al. [58] provide a heuristic method for finding the optimal global neighbourhood size, which requires an exhaustive search over the probable values. Unfortunately the cost of applying the non-linear dimensionality reduction algorithms in a batch, so that we can find the optimal parameters, becomes impractical as the number of feature vectors increases. Furthermore, it is not applicable to problems where prior information is limited (*i.e.*, novelty detection). Mekuz and Tsotsos [59] adapt the local a neighbourhood size using intrinsic dimensionality estimation. They select the local size of neighbourhood by fitting a tangent space at each point. The approach is complex and uses heuristics to converge into an acceptable result. Alternatively, one can use different graph construction rules which tend to be more stable and approximate the manifold better than the global neighbourhood size graphs (k-NN, ε-NN). Yang [60] looks into several approaches [61, 62, 63] to provide a better approximation of the data manifold. The resulting graphs have been demonstrated with new methods of calculating geodesic distances in the work of Meng et al. [64]. Nevertheless, these works [61, 62, 63] still depend on manually setting free parameters that can be verified by exhaustive search.

2.3.4 Proximity graphs

A plausible alternative to the parameter estimation is to use proximity graphs (also known as topological graphs), a family of graphs that describe the topological structure for point sets [65]. Common proximity graphs are: *(i)* the Delaunay Graph (DG); *(ii)* the Ga-

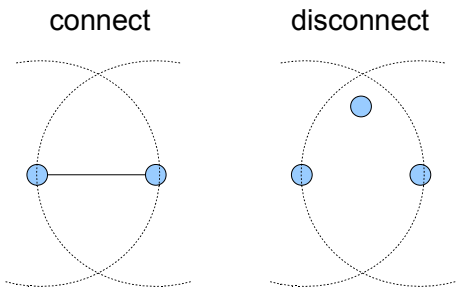


Figure 2.9: Connection example for the Relative Neighbour Graph: If there is no other node inside the lune between the nodes then a connection is established.

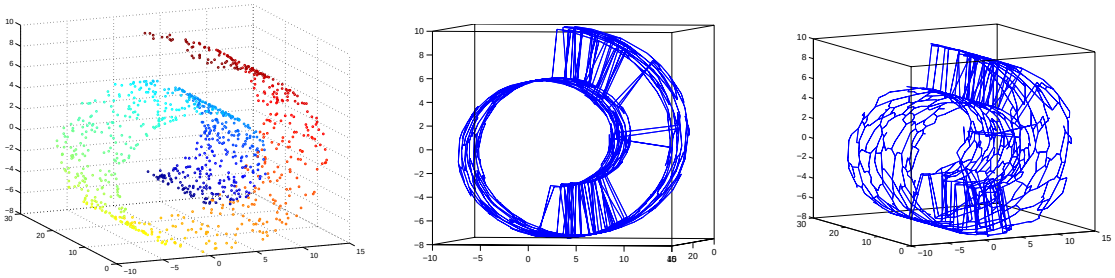
briel Graph (GG) [66]; and *(iii)* the Relative Neighbour Graph (RNG) [67]. Additionally, RNG and GG belong to the sub-family of the β-Skeleton Graphs [68]. These graphs G_{RNG}, G_{GG}, G_{DG} are related to each other by the sub-graph relation,

$$G_{RNG} \subseteq G_{GG} \subseteq G_{DG}. \tag{2.3}$$

where all the graphs are constructed by the same set of vectors.

The proximity graphs are unique and parameter-less, but they have a very high complexity. For example, the RNG is constructed by the rule that two nodes i, j are connected if and only if there is no other node exist between them closer than their pairwise distance $d_{i,j}$. Figure 2.9 provides an example of the application of the RNG rule in two dimensions where a connection between two points is allowed only when no other point exist in the overlapping area of two circular discs. The discs have a radius of $d_{i,j}$ and are centred at the two nodes i and j. To satisfy the RNG connection rule the brute force implementation in high-dimensional data (n vectors) requires $O(n^3)$ operations [67]. Furthermore, it can be demonstrated that under manifolds that fold on themselves (like the Swiss Roll) the output is not following the manifold (Fig. 2.10). Dimensionality reduction using the RNG graph is thus not advisable since the incorrect connections will distort the embedding. Since the RNG is a sub-graph of the GG and DG (*i.e.* the connection in RNG also exist in GG and DG), it is fair to conclude that GG and DG a not suitable for use in dimensionality reduction.

This section presented a brief overview of dimensionality reduction methods commonly used to discover a low-dimensional subspace. It is argued in the literature that non-linear methods are better suited to represent data without assumptions on linearity and



(a) (b) (c)

Figure 2.10: Graph construction for the Swiss roll manifold: (a) Original data, (b,c) different views of the relative Relative Neighbour Graph.

restriction on the pattern distribution. In contrast to the non-parametric linear algorithms, the application of graph-based dimensionality reduction is sensitive to the parameters used to construct the neighbourhood graph. However, methods that attempt to address this issue suffer from high-complexity and often require additional information to be available.

2.4 Summary

This chapter has provided a detailed literature review on video abnormality detection methods, highlighting advantages, limitations and open challenges. The problem is investigated under the novelty detection framework, where a normal event is mainly inferred from frequency of appearance, and rare events are abnormal. Features extracted from the video sequences are processed and utilised to model the normal instances based on information provided by the operator. The models are then used to detect abnormal action in the video. Video abnormality detection methods are required to overcome challenges such as partially annotated data, high-dimensionality and complexity in order to provide an abnormality label for *point*, *contextual*, *collective*, *local* and *global* anomalies that take place in a real-world scene. To achieve this goal, the state of the art utilises a large variety of methods and features to learn the normal action patterns and detect the abnormal ones.

Previous work on abnormal event detection used object detection and tracking/association features. Tracking features, in theory, provide the information needed to create detailed models of usual behaviour, but the various stages have been studied independently assuming that the input is accurate, with assumptions of feature independence and linearity. The matching between the different modules and filters is still an open problem and needs

careful manual tuning of several parameters in order to produce accurate results. To overcome these limitations, recent research is uses mid-level foreground-background segmentation, thus avoiding the tracking module. Nevertheless, such modules do not perform well under camera scenarios where dynamic occlusion is highly probable and they cannot always provide local abnormality detection in crowded scenes. Moving one step further, methods can remove the object detection module and extract low-level features from the image frames. Spatial (local) abnormality is inferred by training *local detectors* associated with sub-regions of the scene. The local temporal abnormality labels are later fused, inferring global spatio-temporal abnormality.

A number of challenges regarding deployment complexity and the *transfer of information* on the video abnormality problem have not been fully addressed. Methods contain hidden parameters that are difficult to define without additional knowledge of abnormal events and costly cross-validation techniques. This affects the practical application of the methods and challenges their performance under real-world scenarios where information is limited and abnormal events are unknown.

Under these restrictions the *transfer of information* to the detector from the training samples is a very challenging task, but this has not been properly investigated in related work. Intermediate processing steps may cause side-effects which distort the distribution of the action patterns. Furthermore, the high-dimensionality of the feature space and the effects of the *curse of dimensionality* have been addressed through linear dimensionality reduction methods without considering the strong assumptions and filtering that these methods impose on the data. It is thus argued that, in order to maximise the *transfer of information*, graph-based dimensionality reduction can provide a low-dimensional subspace that is better suited for abnormality detection. Nevertheless, graph-based dimensionality reduction is itself an open problem and its practical application in the framework of novelty detection is challenging.

The next chapter will describe the proposed approach to use graph-based subspace learning to find a low-dimensional representation of low-level features extracted from a local region of a video sequence. The proposed subspace method is used along low-complexity novelty classifiers to provide a local abnormality detector. Finally, a multi-detector framework for abnormality detection is described.

Chapter 3

Abnormal event detection

3.1 Introduction

This chapter describes a method for training and deploying an abnormality detection framework. The method uses graph-based dimensionality reduction to provide a low-dimensional representation of the feature vectors. The goal is to maximise the *transfer of information*, while keeping the computational and deployment complexity low. The framework is based on the *local detector* [13] principle (Fig. 3.1). To this extent, a set of regions $\mathcal{R} = \{\mathsf{R}_{i,j} \mid i = 1, \dots r_h;\ j = 1, \dots r_h\}$ are defined in the frame and a set of independent local detectors $\mathcal{H} = \{\mathsf{h}_{i,j} \mid i = 1, \dots r_h;\ j = 1, \dots r_h\}$ are associated with each region. Motion vectors are extracted and a low-dimensional representation is produced for each region based on graph-based dimensionality reduction. The independent detector (the novelty classifier) is utilising the learnt subspace to provide local abnormality detection. The abnormality label of each detector is later fused through a voting scheme that labels frames as abnormal based on the recent history of alarms accumulated by the set of detectors.

The main assumptions of the method are:

- *The video sequence is continuous* (*i.e.* it does not contain scene cuts). This is typical in footage from CCTV cameras that is recorded without editing.
- *The normal instances are the majority of the actions taking place in the scene.* This ensure that the problem is an unbalanced classification problem in line with the definition of abnormality detection.

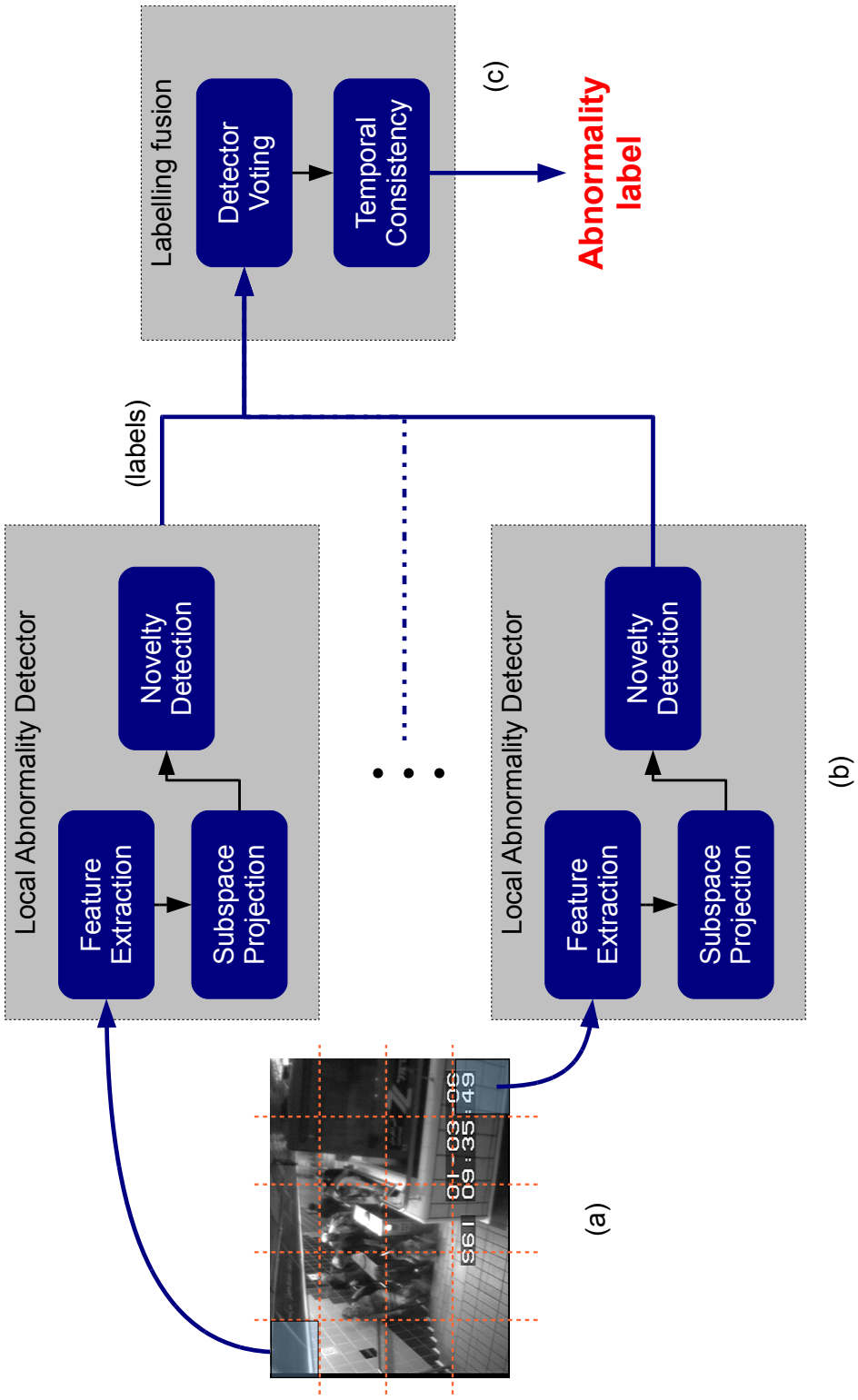


Figure 3.1: Schematic of the multi-detector framework for video abnormal event detection using subspace-aware novelty detectors (a) detector deployment, (b) single detector layout and (c) label fusion

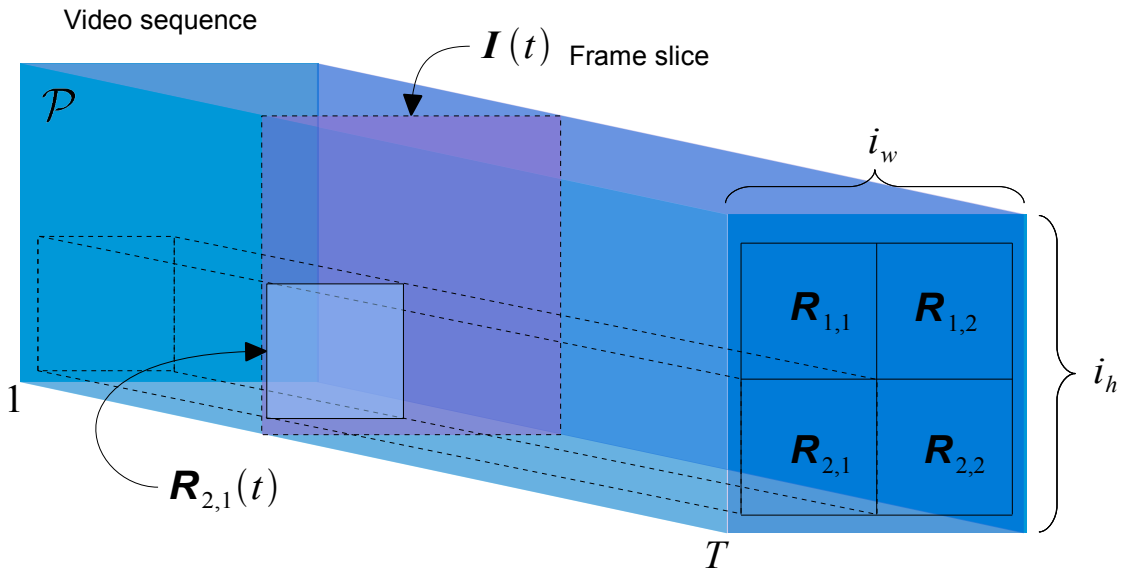


Figure 3.2: Schematic example of the region selection where a video sequence is divided into a 2×2 grid of (spatial) regions.

- *The training labelled frames do not contain any abnormal events.* This is a common assumption for *supervised* and *semi-supervised* novelty detection algorithms. (Note that this only refers to the training samples for the novelty detector and not the subspace learning algorithm, which is unsupervised)

There are no restrictions on the distribution of the vector patterns and no assumptions about the significance of the vector variance to separate the normal and abnormal classes. The next sections describe in detail the various steps for training and testing the proposed abnormality detection framework.

3.2 Feature extraction

Feature extraction is the first filtering process in abnormality detection. Selection of the appropriate feature space affects performance, computational complexity and ease of deployment in a fundamental way. The proposed framework is flexible and can use different low-level features. However, based on experiments in single object and multi-object sequences (see section 4.4) motion vectors have been shown to provide abnormality detection performance when compared to the use of pixel values or change detection masks.

In order to produce the high-dimensional feature vectors, we define I_t to be the image

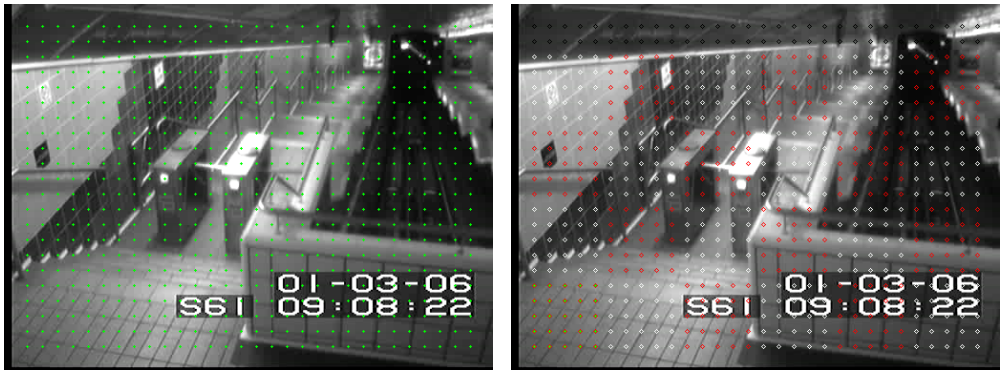
frame of size $i_h \times i_w$ (rows $\times$ columns) at time t of the video sequence and $I_t^C(i,j)$ the colour values of the (i,j) pixel, where $C \in \{R,G,B\}$ and $i = 1,\dots i_h$; $j = 1,\dots i_w$). We also define the grid $\mathcal{R}$ of non-overlapping spatial image regions $\mathsf{R}_{i,j} \in \mathcal{R}$ (where $i = 1,\dots r_h$ and $j = 1,\dots r_w$). Figure 3.2 provides a schematic example of the region placement with a 2×2 grid.

The *motion vectors* are calculated using the block matching technique implemented in the OpenCV library [69]. The image I_t is first converted from colour to grayscale I_t^Y based on,

$$I_t^Y(i,j) = 0.299 I_t^R(i,j) + 0.587 I_t^G(i,j) + 0.114 I_t^B(i,j), \quad \text{with } i = 1,\dots i_h \text{ and } j = 1,\dots i_w. \tag{3.1}$$

A two-dimensional blurring filter is applied on I_t^Y to produce the filtered image $\tilde{I}_t^Y$. The image $\tilde{I}_t^Y$ is divided into non-overlapping blocks b_w pixels (width) and b_h pixels (height). The grid of blocks is defined as $\mathcal{B}(t) = \{B_{i,j}(t) \mid i = 1,\dots \lfloor i_h/b_h \rfloor,\ j = 1,\dots \lfloor i_w/b_w \rfloor\}$ and each block $B_{i,j}(t)$ is a sub-image ($b_h \times b_w$ in pixels) of the grayscale image $\tilde{I}_t^Y$ whose centre pixel is $\tilde{I}_t^Y(\hat{i},\hat{j})$. The index is given by $(\hat{i},\hat{j}) = (\lfloor b_h * i + b_h/2 \rfloor,\ \lfloor b_w * j + b_w/2 \rfloor)$ where the indices i,j refer to $B_{i,j}(t)$. For each $B_{i,j}(t)$ the block matching algorithm searches the neighbourhood of the corresponding pixel position in $\tilde{I}_{t-1}^Y$ to find a block that is most similar to the $B_{i,j}(t)$ in $\tilde{I}_t^Y$. The search is limited to a square area around each corresponding pixel $\tilde{I}_{t-1}^Y(\hat{i},\hat{j})$ of size $2s_h \times 2s_w$. Where the values s_h and s_w define the maximum displacement that can exist due to motion. The metric used to measure similarity is the Sum of Absolute Difference (SAD) of the pixel values. The horizontal and vertical displacement (relative to the centre pixel) of the best match provides the associated motion vector $\mathbf{v}_{i,j}(t)$ of $B_{i,j}(t)$. The resulting motion vector field is $V_t = [\mathbf{v}_{i,j}(t)]_{v_h \times v_w}$ where $v_w = \lfloor i_w/b_w \rfloor$, $v_h = \lfloor i_h/b_h \rfloor$ and $i = 1,\dots v_h$, $j = 1,\dots v_w$. Typical values used for the calculation of the motion vectors in real-world scenes (such as in sequence S5) are: *(i)* $b_w = 19$; *(ii)* $b_h = 19$; *(iii)* $s_w = 20$; *(iv)* $s_h = 20$; and *(v)* window size 5×5 or 7×7 for the filtering of the grayscale image I_t^Y.

Since the search for the estimation of motion vectors at the edges of the image frame is not symmetrical, the value of these motion vectors is ambiguous and often incorrect.



(a) **(b)**

Figure 3.3: Example illustration of dividing the motion vector field into multiple non-overlapping regions: (a) original vector field and (b) region deployment and grouping, with 5×5 motion vectors per region. (Key: regions are depicted by alternating colour patterns)

To avoid introducing these errors into the feature space these edge vectors are removed. Additional errors commonly present in motion vectors calculated through block matching are: *(i)* short bursts of unexpected motion vectors caused by sudden camera motion or errors in the search algorithm; and *(ii)* small jitter of the motion vector due to the finite resolution of the motion vector calculation. Both types of noise need to be addressed before further processing. To remove the first type a median filter is used as it removes short lived spikes while preserving the shape of the signal. The second type of errors is better addressed through a moving average filter, which filters out the high-frequency jitter noise. These one-dimensional filters are applied over time on every component (horizontal and vertical) for each motion vector independently. The window size parameter for the successive filtering (first median then moving average) is estimated by visual inspection of the motion vectors on a small part of the training samples. The results should aim for smooth motion vectors relatively free from high-frequency noise and sudden spikes.

To produce the observation vectors in the regions, we divide the motion vector grid and group the motion vectors according to $\mathcal{R}$ (see example at Fig. 3.3). Each region $\mathsf{R}_{i,j} \in \mathcal{R}$ (where $i = 1, \ldots r_h$ and $j = 1, \ldots r_w$) at time t is associated with the enclosed $m_h \times m_w$ motion vectors from the motion vector field V_t,

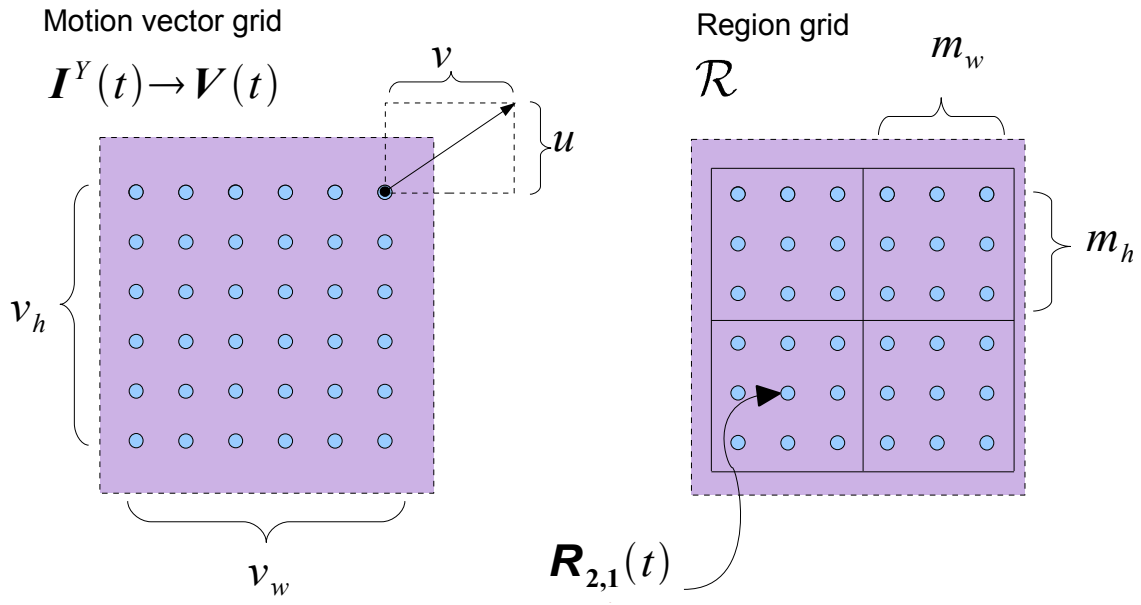


Figure 3.4: Schematic example of motion vector grid and the relationship between the enclosed motion vectors and the regions in the image frame. Each region is associated with the 9 (*i.e.* $m_h = 3$, $m_w = 3$) motion vectors that are enclosed.

$$\begin{pmatrix} \mathbf{v}_{m_h(i-1)+1,m_w(j-1)+1}(t) & \cdots & \mathbf{v}_{m_h(i-1)+1,m_w(j-1)+m_w}(t) \\ \vdots & \ddots & \vdots \\ \mathbf{v}_{m_h(i-1)+m_h,m_w(j-1)+1}(t) & \cdots & \mathbf{v}_{m_h(i-1)+m_h,m_w(j-1)+m_w}(t) \end{pmatrix},$$

where the indices in the matrix are relative to V_t. Figure 3.4 provides a schematic example of the motion vector grid and the division of motion vectors into regions. The observation vector $\mathbf{o}_t^{ij}$, that corresponds to the region $\mathsf{R}_{i,j}$ at time t (*i.e.* $\mathsf{R}_{i,j}(t)$), is constructed by concatenating the individual scalar values of the enclosed motion vectors into a single column vector. Thus, given that a motion vector is two-dimensional $\mathbf{v} = (v, u)$, where v and u are the horizontal and vertical displacement, the observation vector $\mathbf{o}_t^{ij}$ is,

$$\mathbf{o}_t^{ij} = \begin{pmatrix} v_{m_h(i-1)+1,m_w(j-1)+1}(t) \\ u_{m_h(i-1)+1,m_w(j-1)+1}(t) \\ v_{m_h(i-1)+1,m_w(j-1)+2}(t) \\ u_{m_h(i-1)+1,m_w(j-1)+2}(t) \\ \vdots \\ u_{m_h(i-1)+m_h,m_w(j-1)+m_w}(t) \end{pmatrix}. \tag{3.2}$$

According to the *local detector* principle [13] each region $\mathsf{R}_{i,j}$ is associated with a detector $\mathsf{h}_{i,j}$. We define the following sets of the observation vectors $\mathbf{o}_t^{ij}$ that are associated with each region $\mathsf{R}_{i,j}$:

$\mathcal{S}_{i,j}$ contains the feature vectors (both labelled and unlabelled) that are used in the dimensionality reduction process (where $\mathcal{S}_{i,j} \subseteq \mathcal{O}_{i,j}$) .

$\mathcal{X}_{i,j}$ contains the feature vector samples that are annotated as *normal* based on the information provided by the operator (where $\mathcal{X}_{i,j} \subseteq \mathcal{S}_{i,j}$). This information is only used during the novelty classifier training procedure.

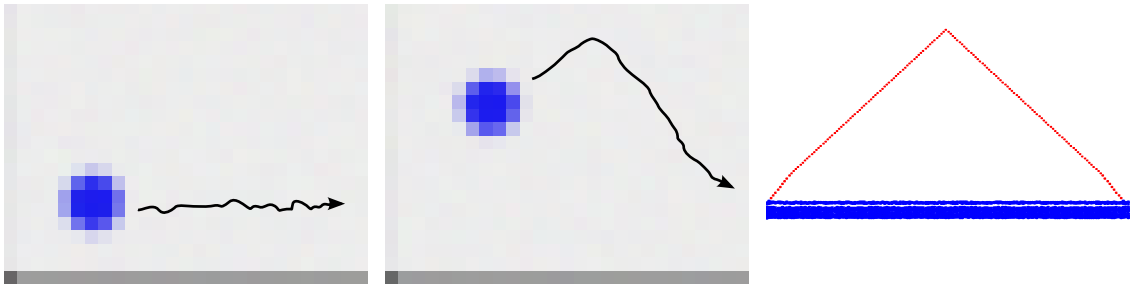
$\mathcal{Z}_{i,j}$ contains the set of unlabelled samples used during the evaluation of the abnormality detection framework.

For simplicity we represent the $\mathbf{o}_t^{ij}$ vectors that belong to the above sets with $\mathbf{s}_t^{ij}$, $\mathbf{x}_t^{ij}$ and $\mathbf{z}_t^{ij}$ respectively.

This section described and illustrated the feature space construction process to produce the original high-dimensional feature vectors for the multi-detector abnormality detection framework. The next section will describe the algorithm to produce a non-linear low-dimensional representation of these high-dimensional feature vectors.

3.3 Subspace learning

Subspace learning is the main module of the proposed abnormality detection framework. It provides a low-dimensional projection with emphasis on separating the normal (statistically common) from the abnormal (statistically uncommon) instances in the feature space. It addresses at the same time the high-complexity, non-linear and unbalanced nature of the problem. Since it is unsupervised, annotation of the data is not necessary; thus the provided subspace uses all available information while being robust to the (possible) existence of abnormal instances. This is achieved by using Laplacian Eigenmaps, a graph-based dimensionality reduction algorithm that uses a neighbourhood graph (graph embedding) to provide a low-dimensional space. When out-of-sample extension is required, an alternative method is utilised that approximates the subspace provided by Laplacian Eigenmaps and allows for new samples (vectors) to be projected in the subspace. To maximise the



(a) (b) (c)

Figure 3.5: Synthetic example sequence S0: (a) normal motion pattern, (b) abnormal motion pattern and (c) illustration of the path of all the motion instances.

transfer of information from the sequence to the projections, one has to select the proper parameters for the graph construction: *(i)* the neighbourhood size; and *(ii)* the weights for the edges of the graph.

Since each region $\mathsf{R}_{i,j}$ is processed independently and in order to allow for a simpler mathematical notation, the analysis of the graph-based dimensionality reduction will concentrate only on the observation vectors $\mathbf{o}_t$ (instead of $\mathbf{o}_t^{ij}$) of the generic region R (instead of $\mathsf{R}_{i,j}$) and of the associated detector h (instead of h_{ij}). In a similar manner the observation sets are $\mathcal{S}, \mathcal{X}, \mathcal{Z}$. Furthermore, the various stages of subspace learning are also illustrated using S0, a synthetic single object video sequence (2240 frames) that depicts single circular disks moving in the scene. Figure 3.5 presents the path of the normal events and the single abnormal instance. The colour, speed and size of the disks remains constant. White and Gaussian noise are added to the image frames and the disc motion respectively. We assume a single region R equal to the image frame I_t and the vector $\mathbf{o}_t$ is produced by concatenating the pixel values of the image matrix into a high-dimensional feature vector.

3.3.1 Temporal ε-graph

As discussed in section 2.2, the neighbourhood graph is crucial to the success of the graph-based subspace learning process. The graph neighbour parameters define what is considered to be *local*. An inappropriate selection of neighbourhood parameters produces a distorted embedding. Unfortunately, in several approaches where graph dimensionality algorithms are used, the *local* neighbourhood is predefined or estimated by trial and error schemes. Such approaches are not suitable for the application of subspace learning in real-world problems (*e.g.* video abnormality detection).

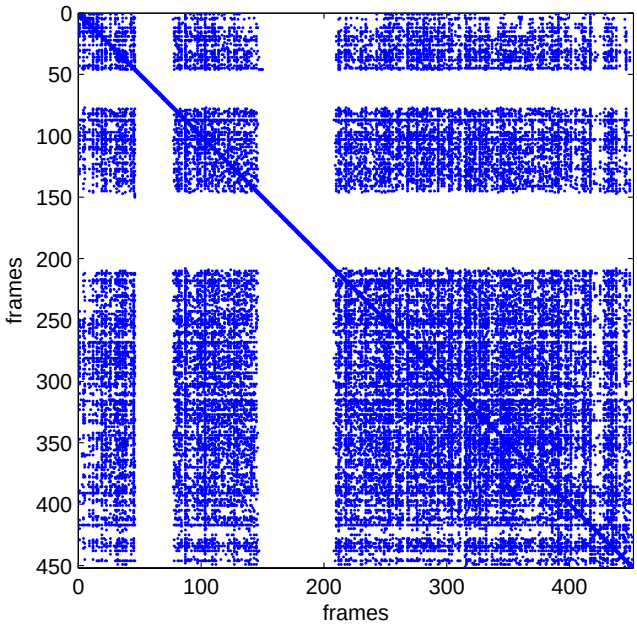


Figure 3.6: A sample matrix of a graph constructed with the temporal e-graph rules. Nodes (frames) are divided into popular and lonely. The regions of low connectivity (frames: 45–75 and 146–207) correspond to frames with slow changing actions.

In special cases, *e.g.* when the uncommon events are known to be composed of actions which evolve slowly in time, it is possible to define the *local* neighbourhood indirectly by exploiting prior knowledge on the patterns that are of interest, by incorporating that information into the graph connection rules. The graph that utilises this information is a variation of the ε-NN graph, where the neighbourhood size of each vector is defined by the similarity of temporally adjacent vectors. The temporal ε-graph, is thus formed in compliance to the following connection rules. Given the set $\mathcal{S}$ of $n_{\mathcal{S}}$ high-dimensional feature vectors:

1. Vector $\mathbf{s}_i$ is always connected to the next in time vector $\mathbf{s}_{i+1}$ with $i = 1 \dots (n_{\mathcal{S}} - 1)$

2. Based on the similarity d_{ij} between vectors $\mathbf{s}_i$ and $\mathbf{s}_j$ (where $j = 1 \dots n_S$ and $j \notin \{i, (i+1)\}$), an undirected connection between $\mathbf{s}_i$ and $\mathbf{s}_j$ is formed when $d_{ij} \leq d_{i(i+1)}$.

The nodes of the graph can be described as *popular* and *lonely* nodes (Fig. 3.6). Vectors nodes that have a large difference from adjacent (in time) nodes, have their neighbourhood threshold relaxed (popular). Vector nodes with small differences are hardly connected

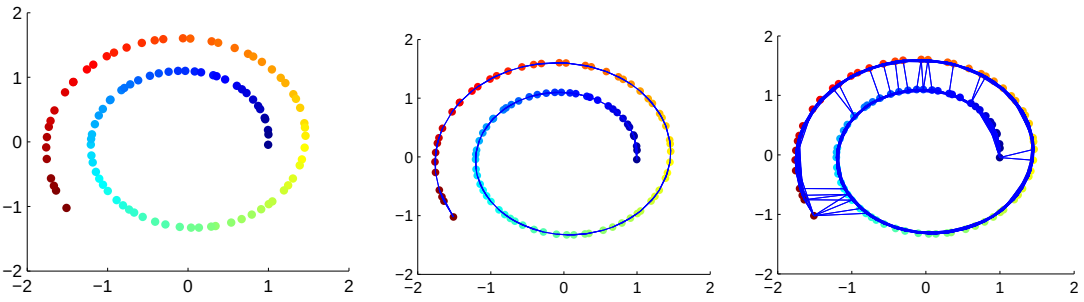


Figure 3.7: Points sampled from a hyperbolic spiral (a) and their k-nearest neighbours graphs for the (b) minimum graph and (c) the distorted graph with incorrect connections when the neighbourhood size is increased.

to any other nodes (lonely), except the previous and next in time vector from the video sequence. These rules will make events with slow changing features stand out in the projections (see section 4.3).

However, such a graph will not provide a good representation when the assumptions about the action patterns do not hold. When the *normal* patterns are described in an abstract manner (*i.e.* frame labels), a more generic approach is necessary.

3.3.2 Minimum k-NN

The proposed method for a generic graph is based on the k-nearest neighbours (k-NN) graph commonly used in the literature. This graph is built according to the rule that each node is connected to at least k neighbourhood (closest) nodes sorted by a similarity measure. Since the structure of the manifold is unknown, similarity is estimated using the Euclidean distance as an approximation. However, Euclidean distance is unreliable in high-dimensional spaces and deviates from the geodesic distance between points that exist on a low-dimensional manifold [48]. As a result, connections in the graph between vectors with large Euclidean distance do not follow the manifold surface and are incorrect (see Fig. 3.7(c)). In order to avoid such connections, we have to keep the graph neighbourhood inside a small area around the reference node where the error between the Euclidean and the geodesic distances is small (see Fig. 3.7(b)).

A good value for the size of the neighbourhood can be achieved by using the minimum k that provides a connected graph through the iterative process described below:

1. Calculate the similarity (distance) function for all the possible vector pairs between the $n_{\mathcal{S}}$ vectors in the training set $\mathcal{S}$. Given two randomly selected vectors $\mathbf{s}_i$, $\mathbf{s}_j$ with $i = 1, \ldots n_{\mathcal{S}}$ and $j = 1, \ldots n_{\mathcal{S}}$, the distance function is:

$$d_{i,j} = \mid \mathbf{s}_i - \mathbf{s}_j \mid . \tag{3.3}$$

   When duplicate vectors exist in the feature space, numerical calculations will give $d_{i,j} = 0$. These cases should be identified and specially treated by the nearest-neighbour search so that the connection is not lost between these vectors and is properly accounted for. A simple method (and the one followed in this work) is to replace the zero value with a very small (but representable) floating point number.

2. Construct the k-NN graph matrix G with duplicates. Given that $\mathcal{K}_i$ is a set of the k nearest neighbours of $\mathbf{s}_i$ and d_i^k is the similarity of the last (most distant) neighbour in $\mathcal{K}_i$, we populate the graph matrix with edges based on the following:

$$g_{i,j} = \begin{cases} d_{i,j} & \text{if } d_{i,j} \leq d_i^k \text{ or } d_{i,j} \leq d_j^k. \\ 0 & \text{otherwise.} \end{cases} \tag{3.4}$$

   This definition provides an extended undirected k-NN which takes into account that in $\mathcal{S}$ it is possible to have several vectors with the same distance from a reference vector. The probability of such duplicate similarity distance is especially high in discrete feature spaces (*e.g.* pixel values, change detection masks and motion vectors). Without this exception, the reference node could randomly connect to one of the candidate nodes and the resulting connectivity would not be unique and geometrically symmetrical (Fig. 3.8).

3. Select the minimum neighbourhood size by iteratively increasing the value k until the produced graph is a connected graph.

The resulting graph matrix represents a sparse undirected graph, which is composed of two type of edges, the *close-in-time* (close to the main diagonal) and the *far-in-time* edges. The *close-in-time* edges represent the evolution of the scene through time and they should exist since we assume that the video sequence evolves smoothly through time (*i.e.* there

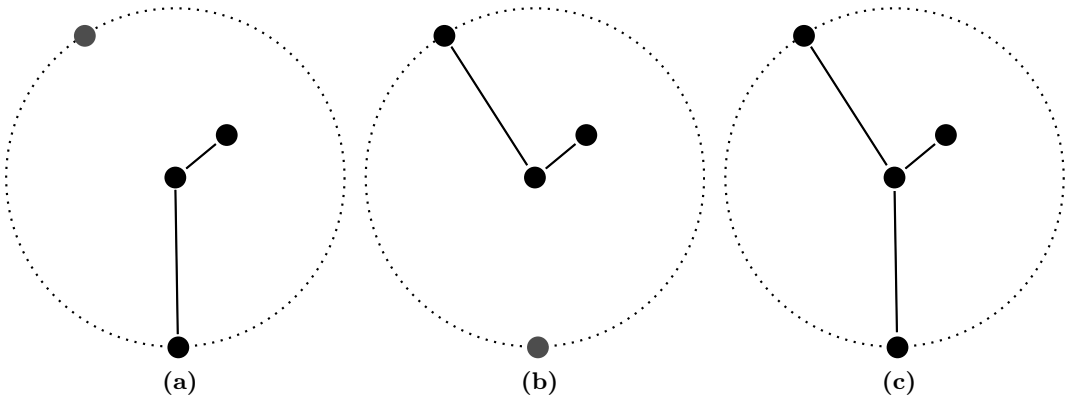


Figure 3.8: k-NN rules example ($k = 2$): (a,b) Original rules only one of the two equidistant nodes will be connected at random and (c) correct nearest neighbour graph using the proposed extension.

are no scene cuts in the video sequence). The *far-in-time* edges represent the similarity between the action patterns where temporal patterns observed at different times in the sequence are similar and thus connected. Figure 3.9 presents the connectivity structure of the minimum k-NN graph produced by using sequence S0. Since in S0 there exist multiple instances of the normal patterns, the graph depicts numerous *far-in-time* connections. Additionally, due to the fact that the actions in the sequence evolve smoothly the *close-in-time* connections are also present. However, as discussed later in subsection 4.3.5, when the main assumptions about the video sequence and the patterns do not hold, the two types of connections in the graph are not in balance and the acquired representation will not be useful for abnormality detection.

Due to the iterative nature of the algorithm to acquire the minimum-k-NN graph, the size of neighbourhood k is not restricted from reaching high values if the pattern classes are very far from each other. While the Laplacian Eigenmaps will be able to provide a projection even under these circumstances, the solution will be computationally impractical and equivalent to spectral clustering methods. To investigate these extreme cases, let us assume that we have a set of N multi-dimensional normal vectors $\mathcal{M}$ and a set of n abnormal vectors $\mathcal{A}$. We also assume that $\mathcal{M} \cap \mathcal{A} = \emptyset$ and $N \gg n$. We combine these vector sets ($\mathcal{M} \cup \mathcal{B}$) and construct the graph G based on the proposed rules. The minimum-k-NN algorithm will start from $k = 1$ and will continue to increase until $k = n-1$. At that point the sub-graph of the abnormal vectors $G_{\mathcal{A}} \subset G$ is fully connected, while the main graph remains unconnected. However, when $k = n$, the rules will force all of the vectors

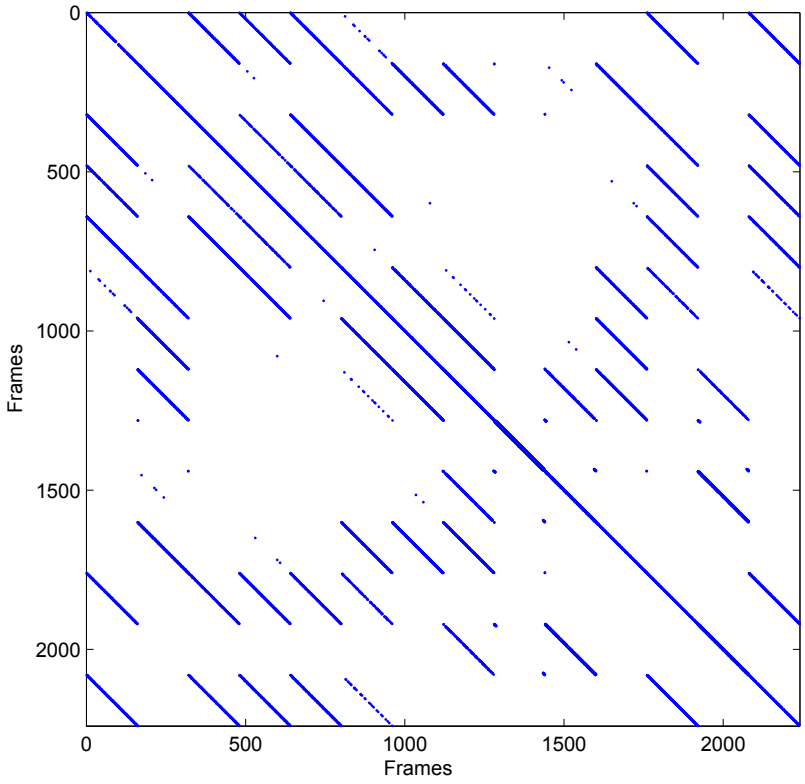


Figure 3.9: Illustration of the connection patterns in the minimum k-NN produced from sequence S0. The connections that are close to the main diagonal represent the time flow, while the off-diagonal connections describe similarity between patterns.

Algorithm 1 K-nearest neighbours with duplicates

function K-NN($data, K$) ▷ construct the K-nn graph from the distance matrix $data$
  $n \leftarrow$ number of rows in $data$
  Create an empty graph g
  for $i = 1$ to n **do** ▷ iterate for every row in the matrix
    $row \leftarrow data[i]$ ▷ get the i-th row
    $index \leftarrow$ argsort(row) ▷ get the index that would sort the row
    for $j = 2$ to $K + 1$ **do** ▷ connect the k nearest neighbours
      $g(i, index[j]) \leftarrow$ connected
      $g(index[j], i) \leftarrow$ connected ▷ undirected graph
    end for
    while $row[index[j + 1]] = row[index[k + 1]]$ **do** ▷ connect equidistant nodes
      $j \leftarrow j + 1$
      $g(i, index[j]) \leftarrow$ connected
      $g(index[j], i) \leftarrow$ connected ▷ undirected graph
    end while
  end for
  return g ▷ Return the graph
end function

Algorithm 2 minimum K-nearest neighbours

procedure MINKNN($data$) ▷ Find connected graph of the data with the minimum K
  $K \leftarrow 1$
  $g \leftarrow$ K-NN(data,K) ▷ Create graph
  while $g \neq connected$ **do**
    $K \leftarrow K + 1$ ▷ Increase neighbours
    $g_{previous} \leftarrow g$ ▷ Save previous graph
    $g \leftarrow K$-nn($data, K$) ▷ Create new graph
  end while
  $g \leftarrow g_{previous}$ ▷ Restore the last connected graph
  return g ▷ Return the graph
end procedure

in $\mathcal{A}$ to connect to the closest point in $\mathcal{M}$, making the graph connected. In practice, the minimum-k-NN becomes fully connected using just a few nearest neighbours. This happens because the features in a video sequence without scene cuts evolve smoothly over time.

Algorithm 1 describes the brute-force implementation of the extended k-NN graph construction. To automatically choose the number of neighbours, we follow the iterative process (Algorithm 2) that provides a connected graph with the minimum possible k. The bottlenecks in the process are the calculation of the distance matrix and, to a lower degree, the intermediate k-NN graph construction. The brute-force algorithm, while being easily parallelised, requires $O(ln^2)$ distance calculations (l is the original vector dimensionality and n is the number of vectors). However, recent work [70] has proved that it is possible to get approximate k-nearest neighbours with a lower than quadratic complexity $O(ln^\kappa)$ where κ is a small number close to 1.

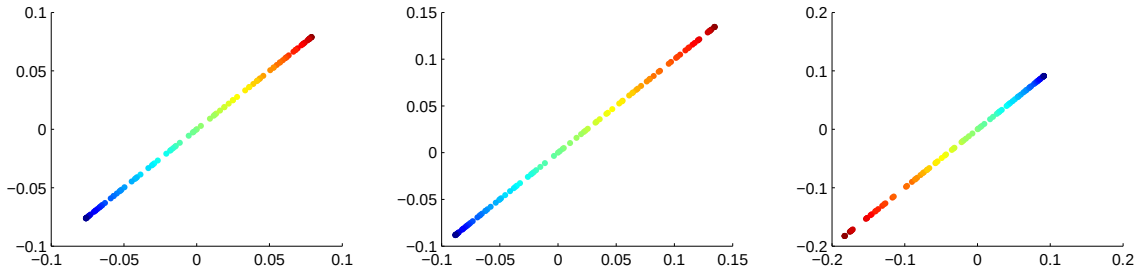
3.3.3 Graph weighting

The last step in the graph construction is to apply the appropriate weighting scheme that will produce the weighted graph matrix W. Given a graph G of $n_{\mathcal{S}}$, matrix W is based on the weighting function [47]:

$$w_{ij} = \begin{cases} e^{-\frac{g_{ij}^2}{\tau}} & \text{if } g_{i,j} > 0, \\ 0 & \text{otherwise.} \end{cases} \tag{3.5}$$

where $i = 1, \dots n_{\mathcal{S}}$ and $j = 1, \dots n_{\mathcal{S}}$. Parameter τ effectively separates the local neighbourhood into *close* and *distant* connections. The simple scheme is to use a binary representation and to assign "1" where there is an edge between two nodes and "0" otherwise. We can achieve this result by setting $\tau = \infty$ in Eq. 3.5. The value $\tau_\infty = \infty$ preserves the general information about the local structure in the proximity of each vector, but discards the relative importance among the neighbours. By setting the value of τ, we can scale the influence that neighbourhood (connected) nodes have. Values very close to zero result in a numerically unstable eigenvalue problem, and thus should be avoided. Values close to $+\infty$ dictate that only the graph structure is important for the embedding. This value is highly popular among several graph-based dimensionality reduction approaches that expect a global manifold to exist. It is also generally preferred as it has a small memory footprint and the eigenvectors converge faster to a solution. If we select a value between those extremes (*i.e.* $\{0, \infty\}$) we can exploit not only the fact that nodes are neighbours, but also how close they are to each other. In theory, the latter weighting strategy holds more information about the local neighbourhood. Both weighting schemes produce similar results when the number of nodes reaches infinity. However, the latter needs careful selection of the scale parameter τ to avoid errors in the numerical solution of the generalised eigenvalue problem.

Figure 3.10 gives an example for such values for the scale parameter using Laplacian Eigenmaps (LE) and projecting into one dimension. All values produce an unfolding of the global manifold. The colour sequence in the projections is the same as would be perceived if we were running along the spiral. As we move from larger to smaller values of τ, the grouping on the points in the line follows more closely the grouping that is apparent in



(a) $\sigma = +\infty$ **(b)** $\sigma = 0.20$ **(c)** $\sigma = 0.12$

Figure 3.10: Points sampled from a hyperbolic spiral (Fig.3.7a) and their LE projection into one dimension using the minimum connected graph with different value for the scale parameter τ. Smaller values provide more information about the grouping of the points that exist on the manifold. (Key: τ values are relative to the maximum edge length in the graph)

Figure 3.7(a). As a result the weighting of the graph values is very important for the discovery of clusters on the manifold.

In this work three different values for τ were used in the experiments:

- The binary scheme $\tau_\infty = \infty$. This value discards any similarity information between the feature vectors and only considers their connectivity.

- The value $\tau_\mu = \frac{\sum_{ij} g_{ij}^2}{n_e}$, where n_e is the number of vertices (edges) in the graph G. This value represents the average distance in the graph. It is used in our initial experiments as an estimate for τ to separate between the *close* and *distant* connections in the graph and thus improve the grouping of the patterns in the projections. However, the weighted graph based on τ_μ was found to cause issues with the numerical stability of the eigen-solver algorithm used in graph embedding (see subsection 4.3.3). In these cases, a better estimation τ_o is proposed.

- The value τ_o is provided by applying Otsu's method [71] on the histogram of the squared edges ($g_{ij}^2 > 0$) of the graph. Otsu's method is an unsupervised decision procedure which separates a set of values into two classes and aims to find a value τ (threshold) that minimises the inter-class variance. The procedure is commonly used for estimating the threshold that clusters grayscale images into two classes by using the intensity histogram. In this work, it is used to provide the value τ_o, based on the edge histogram of the neighbourhood graph, that would separate the edges into *close* and *distant* neighbours. Value τ_o is also the preferred value since it was

found to provide better stability for the eigen-solver algorithms and good projection results.

3.3.4 Graph embedding

The process of graph embedding follows closely the graph-based non-linear dimensionality reduction method Laplacian Eigenmaps (LE), as described in [47]. The method is based on the Graph Laplacian of the neighbourhood graph created over the end points of the high-dimensional input vectors. Given a set of $n_{\mathcal{S}}$ multi-variate observations $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_{n_{\mathcal{S}}}$ in $\mathbb{R}^l (l \gg 1)$, and a weighted graph G built over the endpoints of these vectors; LE solves the problem of mapping $\mathcal{S}$ to $\mathcal{Y}$ (of lower dimensionality m) so that the connected points in G stay as close as possible.

Consider the mapping of the high-dimensional feature vectors to a line thus $\mathcal{Y} = \{y_1, y_2, \ldots, y_{n_{\mathcal{S}}}\}$ with $y_i \in \mathbb{R}$ (where $i = 1, \ldots n_{\mathcal{S}}$). This mapping can also be represented as a vector $\tilde{\mathbf{y}} = (y_1, y_2, \ldots, y_{n_{\mathcal{S}}})^\top$. A reasonable criterion for choosing a good map is to minimise the following objective function,

$$\sum_{ij} \parallel y_i - y_j \parallel^2 w_{ij} \text{ with } i = 1, \ldots, n_{\mathcal{S}} \text{ and } j = 1, \ldots, n_{\mathcal{S}} \tag{3.6}$$

The objective function (3.6) with our choice of weights w_{ij} incurs a heavy penalty if neighbouring vectors $\mathbf{s}_i, \mathbf{s}_j$ are mapped far apart. Belkin and Niyogi [47] show that the minimisation problem can be expressed as,

$$\frac{1}{2} \sum_{ij} \parallel y_i - y_j \parallel^2 w_{ij} = \tilde{\mathbf{y}} L \tilde{\mathbf{y}} \tag{3.7}$$

where $L = \Delta - W$ is the combinatorial Graph Laplacian matrix. Δ is the diagonal weight matrix whose entries are columns (or rows, since W is symmetric) sums of W (*i.e.* $\delta_{ii} = \sum_j w_{ij}$). The problem in (3.7) reduces to finding

$$\underset{\tilde{\mathbf{y}}^\top \Delta \tilde{\mathbf{y}} = 1}{argmin}\, \tilde{\mathbf{y}}^\top L \tilde{\mathbf{y}}. \tag{3.8}$$

In the more generic case where we embed the graph into $\mathbb{R}^m$ the result is given by the $n_{\mathcal{S}} \times m$ matrix $\tilde{Y}$ where the i-th row provides the embedding coordinates $\mathbf{y}_i = (\tilde{y}_{i,1}, \ldots \tilde{y}_{i,m})^\top$ of

the original vectors $\mathbf{s}_i$. In this case the optimisation problem becomes,

$$\underset{\tilde{Y}^\top \Delta \tilde{Y} = I}{argmin}\, tr(\tilde{Y}^\top L \tilde{Y}), \tag{3.9}$$

where $tr()$ is the trace of the matrix.

The constrain $\tilde{\mathbf{y}} \Delta \tilde{\mathbf{y}} = 1$ for the one dimensional embedding prevents the solution from collapsing onto a point. For the multi-dimensional embedding, $\tilde{Y}^\top \Delta \tilde{Y} = I$ prevents collapse of the solution to a space of dimension less than m. The solution is provided by the eigenvectors corresponding to the lowest, non-zero, eigenvalues of the generalised eigenvalue problem:

$$L\tilde{\mathbf{y}} = \lambda \Delta \tilde{\mathbf{y}}. \tag{3.10}$$

The procedure to perform LE is formally stated below:

1. Compute the combinatorial Graph Laplacian L.
2. Solve the generalised eigenvalue problem of the graph Laplacian (Eq.3.10).
3. Embed into m-dimensional space using the first m eigenvectors in ascending order of eigenvalues, starting from the first non-zero eigenvalue,

$$\begin{aligned} &\text{with } \lambda_0 = 0 < \lambda_1 < \lambda_2 < \ldots < \lambda_m \\ &\mathbf{s}_i \rightarrow \mathbf{y}_i = (\tilde{\mathbf{y}}_1(i), \tilde{\mathbf{y}}_2(i), \ldots, \tilde{\mathbf{y}}_m(i))^\top \end{aligned} \tag{3.11}$$

   where $\tilde{\mathbf{y}}_j(i)$ is the i-th element of the j-th eigenvector and $\mathbf{y}_i \in \mathcal{Y}$ the low-dimensional representation of $\mathbf{s}_i$.

The main computational module of LE is the solution of a sparse generalised eigenvalue problem for which there are a number of established, fast and stable eigensolvers.

Due to the locality preserving characteristics of the Graph Laplacian, LE is able to compactly group normal (common) patterns in the projections and "separate" them from the abnormal (uncommon), without assumptions on linearity or variance significance (Fig. 3.11a,b). Such a difference in the mapping of normal and abnormal instances is beneficial to novelty classification as presented in the experiments (see subsection 4.4.3). In contrast, global linear methods such as MDS and PCA expect the feature space to follow strict assumptions, thus "separation" is not always possible (Fig. 3.11c,d).

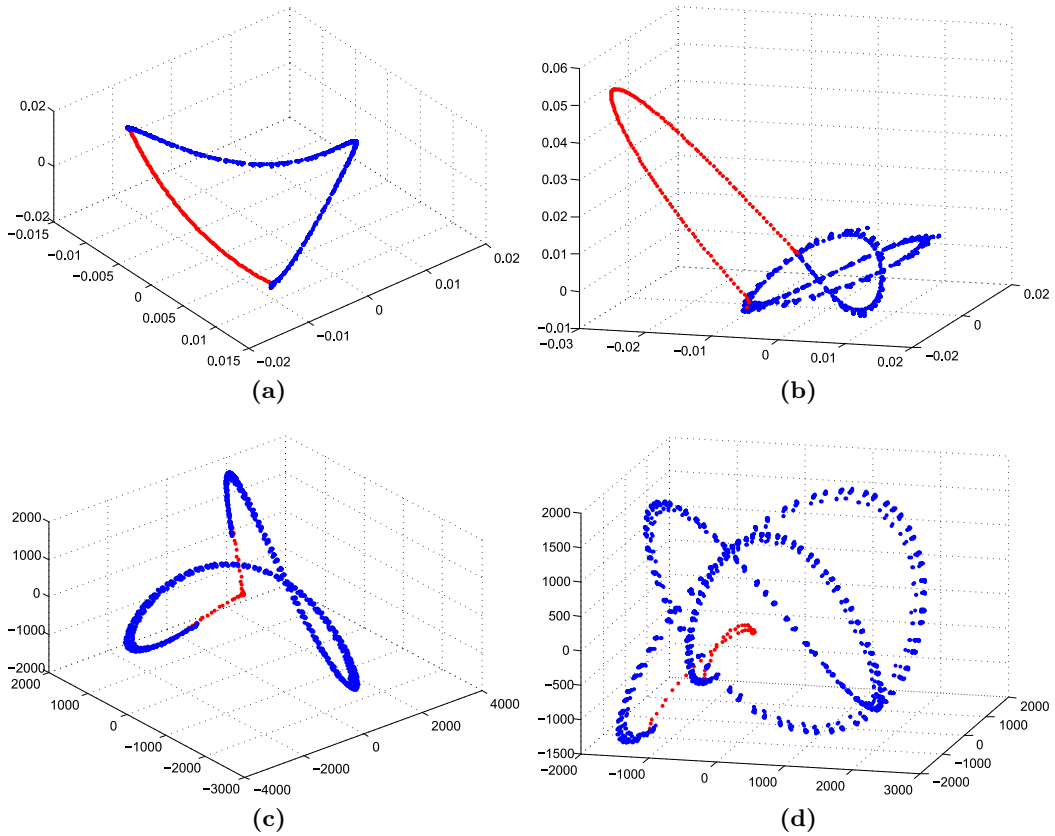


Figure 3.11: LE projection normal and abnormal actions in the S0 sequence using the weighted (τ_o) minimum k-NN graph: (a) dimensions 1-3 and (b) dimensions 4-6. The *abnormal* event is projected away from the main class of normal patterns (especially for b), while the MDS projection is not providing the same quality of separation since the *abnormal* pattern is enclosed inside the *normal*: (c) dimensions 1-3 and (d) dimensions 4-6. (Key: The normal (blue) and abnormal (red) instances are manually colour coded based on the known ground-truth to aid the visual inspection of the results)

3.3.5 Out-of-sample extension

One of the issues that emerges when using Laplacian Eigenmaps is the inability to project new points into the learnt subspace. This limitation restricts the application of LE only to off-line based abnormality detection. To alleviate this limitation the Locality Preserving Projections [72] algorithm is used. This method is also based on the Graph Laplacian and shares similar characteristics with its off-line alternative Laplacian Eigenmaps. In this thesis, we use the spectral regression flavour of Locality Preserving Projection (SR-LPP), a graph-based linear dimensional reduction, which finds the linear approximation of the Laplacian Eigenmaps embedding using a spectral regression technique [73]. The result is a linear mapping that is able to compensate for the non-linear correlation among the features in the input vectors.

Following the SR-LPP method we estimate the projection matrix A which maps the vectors $\mathbf{s}_i \in \mathcal{S}$ (where $i = 1, 2, \ldots, n_{\mathcal{S}}$) to a low-dimensional representation $\mathbf{y}_i \in \mathcal{Y}$ based on the information that exists in the graph W. Consider the projection of the initial vectors to a line $\mathcal{Y} = \{y_1, y_2, \ldots, y_{n_{\mathcal{S}}}\}$ with $y_i \in \mathbb{R}$. This mapping can also be represented as a vector $\tilde{\mathbf{y}} = (y_1, y_2, \ldots, y_{n_{\mathcal{S}}})^\top$. The solution is based on solving the minimisation problem in (3.8),

$$\underset{\tilde{\mathbf{y}}^\top \Delta \tilde{\mathbf{y}}=1}{argmin}\, \tilde{\mathbf{y}}^\top L \tilde{\mathbf{y}} = argmin\, \frac{\tilde{\mathbf{y}}^\top L \tilde{\mathbf{y}}}{\tilde{\mathbf{y}}^\top \Delta \tilde{\mathbf{y}}},$$

since $L = \Delta - W$ the minimisation problem can be converted to the equivalent,

$$argmax\, \frac{\tilde{\mathbf{y}}^\top W \tilde{\mathbf{y}}}{\tilde{\mathbf{y}}^\top \Delta \tilde{\mathbf{y}}}, \tag{3.12}$$

Thus the solution to the optimal mapping is given by the eigenvectors corresponding to the maximum eigenvalues of the generalised eigenvalue problem,

$$W \tilde{\mathbf{y}} = \lambda \Delta \tilde{\mathbf{y}}. \tag{3.13}$$

Note that SR-LPP is based on (3.13) which is equivalent to (3.10), thus LE and SR-LPP are expected to have similar mappings when compared to each other.

To be able to map new instances we assume a linear projective function $f(\mathbf{s}_i) = \mathbf{a}^\top \mathbf{s}_i$ and thus $\tilde{\mathbf{y}} = S^\top \mathbf{a}$ (where the i-th column of S corresponds to the $\mathbf{s}_i$ vector). In this case,

equation (3.12) becomes,

$$argmax\ \frac{\tilde{\mathbf{y}}^\top W \tilde{\mathbf{y}}}{\tilde{\mathbf{y}}^\top \Delta \tilde{\mathbf{y}}} = argmax\ \frac{\mathbf{a}^\top S W S^\top \mathbf{a}}{\mathbf{a}^\top S \Delta S^\top \mathbf{a}}. \tag{3.14}$$

Thus the optimal $\mathbf{a}$'s are produced by solving the maximum eigen-problem:

$$S W S^\top \mathbf{a} = \lambda S \Delta S^\top \mathbf{a}. \tag{3.15}$$

This problem however has a high computational complexity and does not scale well with the number of vectors, yet it is possible to find the linear projective functions without solving the dense eigen-problem (3.15). In their work Cai et al. [73] suggest to use spectral regression where the best fit is given by the regularised least square minimisation problem:

$$\underset{\mathbf{a}}{argmin} \left(\sum_{i=1}^{n_{\mathcal{S}}} (\mathbf{a}^\top \mathbf{s}_i - \tilde{y}_i)^2 + \epsilon \parallel \mathbf{a} \parallel^2 \right) \tag{3.16}$$

where $n_{\mathcal{S}}$ is the number of vectors in $\mathcal{S}$ and $\tilde{y}_i$ is the i-th element of the eigenvector $\tilde{\mathbf{y}}$ calculated by the eigenproblem (3.13). The solution of Eq. (3.16) for the first m major eigenvectors produce the transformation $l \times m$ matrix A, whose i-th column is the $\mathbf{a}_i$ eigenvector of (3.16). Using matrix A new observations can be projected into the low dimensional subspace using equation (2.2). The method is implemented in two steps: *(i)* find the m major eigenvectors of (3.13); and *(ii)* estimate the eigenvectors of (3.15) using the regularised list square equation (3.16).

Equation (3.16), is also called ridge regression and the parameter $\epsilon \geq 0$ controls the amount of shrinkage. Ridge regression overcomes issues with small sample sizes and provides more stable solutions especially when the dimensionality of the feature vectors is high [74]. Furthermore, experimental results have shown that using SR-LPP to perform LPP provides a closer approximation to the Laplacian Eigenmaps compared with alternative implementations of the LPP projection (see subsection 4.6.2).

This section has described an unsupervised method, based on graph-based subspace learning techniques, to discover a low-dimensional subspace that is suitable for abnormality detection. The next section will utilise the subspace to deploy and train a multi-detector framework for video abnormality detection.

3.4 Abnormality detection

This section describes the procedure to deploy a multi-detector framework to address the video abnormality detection problem. For each region R_{ij} (where $i = 1, \ldots, r_h$; and $j = 1, \ldots, r_h$) we associate the set of observations $\mathcal{S}_{ij}$, $\mathcal{X}_{ij}$, $\mathcal{Z}_{ij}$ that are extracted from that region. We then define a "subspace-aware" novelty detector which is a classifier operating on the low-dimensional subspace of the high-dimensional observation vectors. The graph-based dimensionality reduction method produces the low-dimensional representation $\mathcal{Y}_{ij}$ of the feature space utilising both labelled and unlabelled data in $\mathcal{S}_{ij}$. A novelty classifier h_{ij} is then trained in the low dimensional subspace based on the normal labelling (annotation) $\mathcal{X}_{ij}$, provided by the operator. Unlabelled samples $\mathcal{Z}_{ij}$ are projected onto the learnt low-dimensional subspace and the trained classifier will provide the abnormality label.

In off-line detection, the classifier training and testing sets are part of the subspace learning ($\mathcal{X}_{ij} \subseteq \mathcal{S}_{ij}$ and $\mathcal{Z}_{ij} \subseteq \mathcal{S}_{ij}$). The two pipelines for off-line and on-line detectors are presented in Figures 3.12 and 3.13.

3.4.1 Novelty classifier

While the proposed method does not restrict us to a specific novelty classifier, we prefer to select a one-class classifier that: has a small *operational footprint*; and can be trained using only the annotated normal instances from the subspace. The *operational footprint* of the novelty classifier refers to processing and memory requirements of the novelty classifier during abnormality detection. A commonly used classifier is formed by Gaussian mixture models (GMM). GMMs are based on the assumption that the patterns (classes) can be approximated with a mixture of Gaussians. The advantages of this approach are: *(i)* the novelty classifier can be trained in an unsupervised manner; and *(ii)* the training samples can be discarded keeping only a small number of parameters that describe the mixture model.

Assuming that the GMM is composed of n mixtures and the feature vector dimensionality is m (*i.e.* the dimensionality of the subspace), then the number of parameters required for each mixture component are: m^2 for the covariance matrix; nm for the parameters of the mean value; and n weights to measure the influence of each mixture component. Thus the total number η_{GMM} of parameters for models with full covariance is

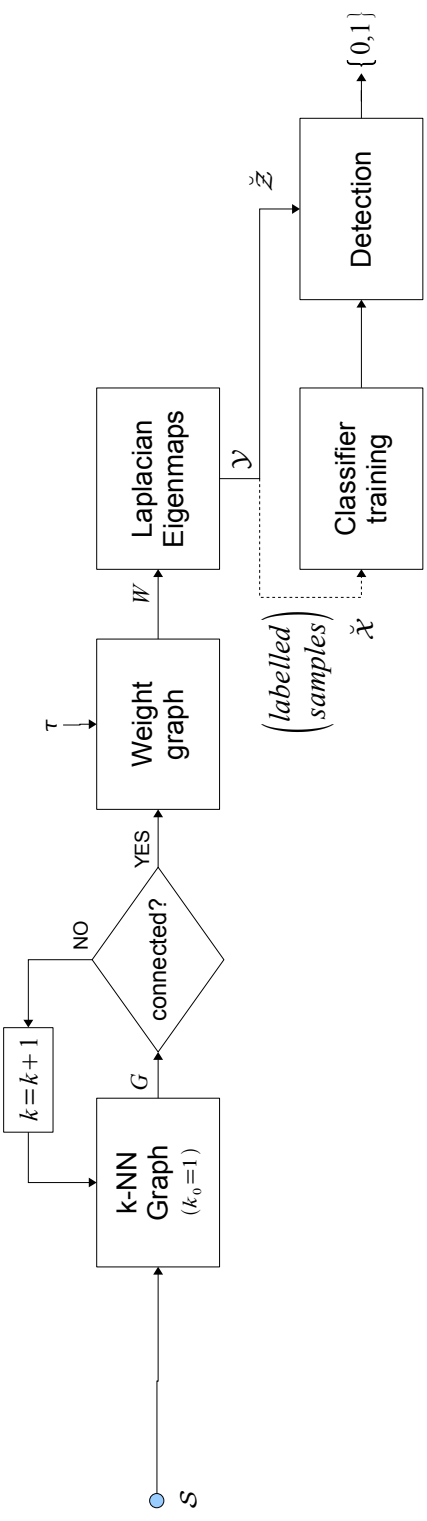


Figure 3.12: Off-line abnormality detection using the subspace learnt from Laplacian Eigenmaps for a region R. (Note: the sets $\check{\mathcal{X}}$ and $\check{\mathcal{Z}}$ refer to the projected samples of the original high-dimensional sets)

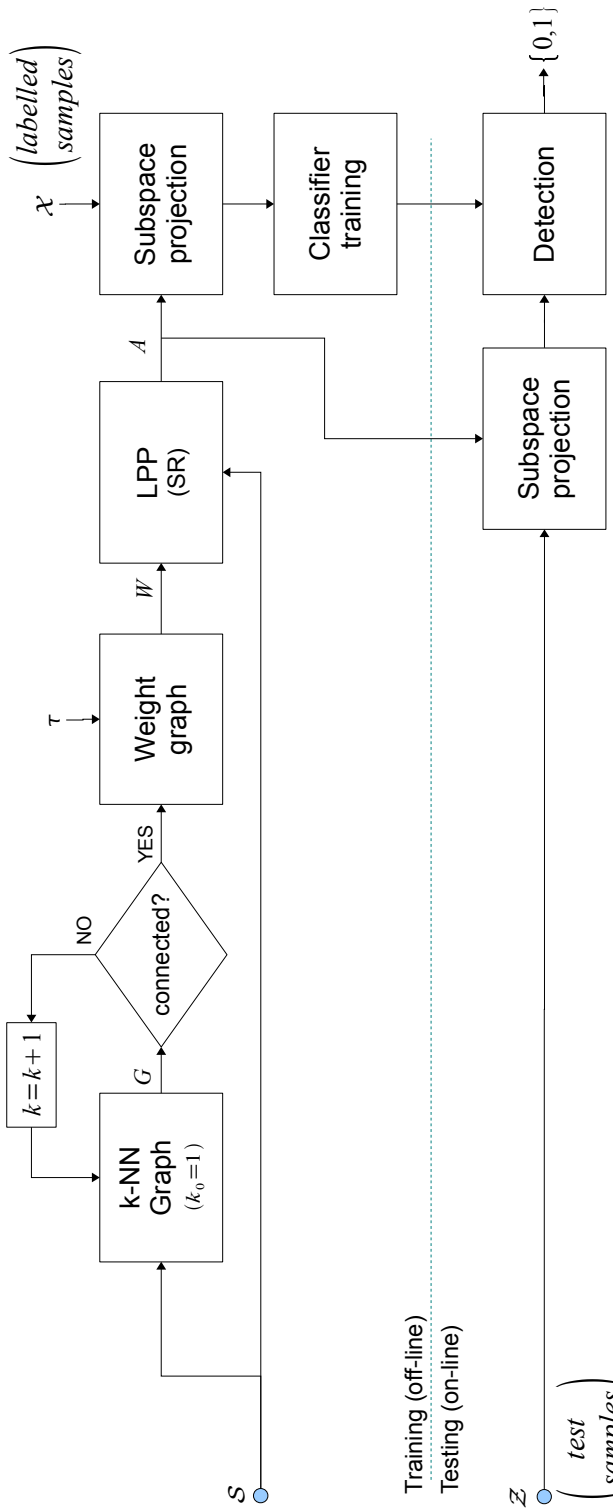


Figure 3.13: On-line abnormality detector for region R using the subspace learnt by SR-LPP.

$\eta_{GMM} = n(m^2 + m + 1)$. To reduce this number, the modelling assumes only diagonal covariance matrices thus the number is now $\tilde{\eta}_{GMM} = n(2m + 1)$.

In contrast, the use of methods such as an one-class SVM would require setting of training parameters and demand an equal amount of positive and negative annotated samples. Unfortunately the parameters are not easily optimised in the abnormality detection problems where prior knowledge of abnormal events is lacking and the events are rare (unbalanced classification problem). Alternative novelty classifiers such as nearest neighbours have been shown to be effective but they require the training samples to be present during detection and are thus more complex and require more memory than GMMs.

For these reasons, the use of GMMs is more appropriate for real-time abnormality detection. Thus for each region R_{ij} a GMM is trained based on the projection $\breve{\mathcal{X}}_{ij}$of the associated training set $\mathcal{X}_{ij}$ of annotated normal instances. The GMM model is fit using the Expectation Maximisation (EM) algorithm and the number of mixture components is selected by maximising the Bayesian Inference Criteria (BIC) over a range for the number of components (*e.g.* $[1, 10]$).

Unlabelled samples $\mathbf{z}_t^{ij} \in \mathcal{Z}_{ij}$ are mapped into the low dimensional space ($\mathbf{z}_t^{ij} \rightarrow \breve{\mathbf{z}}_t^{ij}$) and ranked for abnormality based on $f_{rank}(\breve{\mathbf{z}}_t^{ij}) = -log(pdf_{GMM}(\breve{\mathbf{z}}_t^{ij}))$. Given a threshold θ_{ij} we label observations as *normal* or *abnormal* using,

$$\mathsf{h}_{ij}(\breve{\mathbf{z}}_t^{ij}) = \begin{cases} 0 & if\ f_{rank}(\breve{\mathbf{z}}_t^{ij}) \leq \theta_{ij} \\ 1 & otherwise \end{cases}. \tag{3.17}$$

3.4.2 Multi-detector fusion

After training, each local detector $\mathsf{h}_{i,j}$ provides the local motion abnormality label for the associated motion feature vectors $\mathbf{o}_t^{ij}$ of the single region $R_{i,j}$ at time t in the scene. To reduce false alarms in the detector grid $\mathcal{H} = \{\mathsf{h}_{ij} \mid i = 1, \ldots r_h,\ j = 1, \ldots r_w\}$, we infer the abnormality $\mathsf{H}(t)$ of the complete framework by placing temporal constrains on the labelling of the local detectors. The method was proposed by Adam et al. [13] and is achieved using the following functions,

$$f_\alpha(t) = \begin{cases} 0 & \text{if } \sum_{ij} \mathsf{h}_{ij}(t) < \alpha \\ 1 & \text{otherwise} \end{cases}, \tag{3.18}$$

$$\mathsf{H}(t) = \begin{cases} 0 & \text{if } \sum_{t-\gamma}^{t} f_\alpha(t) < \beta \\ 1 & \text{otherwise} \end{cases}. \tag{3.19}$$

The system has two states: *(i)* the *alarm state* which is set when $f_a(t) = 1$; and *(ii)* the *abnormal state* which is set when $\mathsf{H}(t) = 1$. The *abnormal state* is the final output of the algorithm to the operator. The parameters are defined as follows: *(i)* γ is the number of frames in history that we look back from the current frame; *(ii)* β is the number of alarms in history that are necessary to consider the current frame abnormal; and *(iii)* α is the number of detectors that are required to have abnormal labels before we declare the current frame at alarm state.

3.4.3 Parameter selection

Since abnormal events are not available during training, the parameters α, β, γ and the threshold θ need to be defined manually. Parameter α depends upon the size of the detector relative to the size of the moving objects. For example when small detectors are deployed in scenes with large moving objects then we might need to set $\alpha > 1$. In our case since the size of the detectors is not very small when compared to the moving objects (humans) the value is fixed to 1. Parameters β, γ are selected based on the expected minimum duration for abnormal events in order to temporally filter the results from the detectors. This duration is subject to the combination of the frame-rate and the expected speed of the moving objects (faster moving objects mean that the duration should be smaller to compensate). The relative difference between these two values $(\gamma - \beta)$ give some robustness to occlusions or missing detections and it should not be bigger than $\beta/2$ to avoid merging different events together. Regarding the threshold θ, if we assume that the training sample is free of abnormal instances then a safe value is the maximum rank that is attributed to the training set of vectors. Selecting larger values for the threshold is considered arbitrary and cannot be validated without examples of abnormal events. The number of dimensions

for the subspace is no more than 10 dimensions to keep the computational complexity of the on-line detection low.

3.5 Summary

This chapter has described the methodology for the training and deployment of an abnormality detection framework that is composed of independent novelty classifiers utilising a low-dimensional subspace. The goal is to maximise the information that can be extracted (*transfer of information*) from all the available vectors (both annotated and unlabelled) and to discover a low-dimensional space that reflects the structure and frequency of the motion patterns that exist in the associated region of interest.

This goal is achieved using unsupervised *dimensionality reduction* methods based on the Graph Laplacian of a neighbourhood graph (minimum k-NN). The neighbourhood size k and weighting τ parameter are automatically estimated from the provided data. The weighted graph is used by the graph-based subspace learning approaches of Laplacian Eigenmaps or Locality Preserving Projections (when we require on-line abnormality detection), thus a low-dimensional projection is discovered where normal events are mapped into compact clusters and abnormal instances are mapped far away.

Such a setup allows for a more effective training of a variety of novelty classifiers. However, due to complexity and training limitations that exist in the video abnormality detection problem, it is preferable to use methods that have a small operational footprint and that can be trained in an unsupervised way (automatic selection of model parameters). With this aim, GMMs have been selected to detect abnormal instances.

The framework is based on a grid of non-overlapping regions over which a graph-based dimensionality reduction algorithm discovers a low-dimensional subspace of the high-dimensional features associated with each local region. A novelty classifier, trained on the feature subspace with annotated data from the operator, learns the local normal motion patterns and provides local abnormality detection. Using these subspace-aware local detectors it is possible to infer abnormality for a larger area surveyed by the CCTV sensor by fusing the local abnormality labels that they produce. The process follows the work of Adam et al. [13], who propose a low complexity temporal filtering approach to fuse detector labels and provide abnormality detection for the full scene.

The overall complexity of the on-line detection is low and can be easily performed in embedded systems, while training is dominated by the cost of the graph construction. Nevertheless, subspace learning is required only once and the methodology can exploit recent algorithms on parallel systems to reduce the computational cost. The complete framework requires a small number of training parameters associated with the physical aspects of the scene and thus can be inferred empirically using neither detailed knowledge of the abnormal events nor cross-validation techniques.

The next chapter presents the experimental results that support and justify the motivation for the proposed approach. The method is applied on single and multi-object datasets in experiments of incremental complexity to verify advantages and identify limitations.

Chapter 4

Experimental results

4.1 Introduction

This chapter provides experimental results on the application of the proposed approach for video abnormality detection. The experiments investigate several aspects of the framework. The off-line graph-based subspace learning method is compared with alternative linear and non-linear dimensionality reduction methods to demonstrate the ability to maximise the *transfer of information* from the feature space to a low-dimensional subspace suitable for abnormality detection. Then, the performance advantage of motion vectors against alternative low-level features is verified on both single object and crowded real-world sequences. The resulting off-line local subspace detector is later compared against alternative novelty classifiers to demonstrate the performance advantages. Finally, the full multi-detector framework is compared with linear subspace learning to identify issues and limitations of the application in real-world scenarios.

4.2 Datasets

This section will describe and discuss the motivation behind the use of the specific sequences used for the subsequent experiments. These datasets are grouped into two categories: single; and multi-object, based on the number of moving objects that simultaneously exist in the scene.

(a) **(b)** **(c)**

Figure 4.1: Events in sequence S1: (a) car, (b) man and (c) *null* event.

4.2.1 Single-object dataset

In this work, single object sequences are used as a baseline dataset to test, evaluate and compare the use of the subspace learning methods (LE and SR-LPP) in the proposed abnormal event framework. We select a small fixed region of interest (*i.e.* there is no tracking of the moving objects) from the MPEG-7 dataset (highway) and we create a number of event sequences by concatenating *normal* and *abnormal* events separated by frames of inactivity. The selected region (54×52 pixels in size) is usually occupied by a single object. Four sequences were created in order to test and compare the proposed subspace learning algorithm. These are described in more detail below:

S1 is a small sequence depicting two events with completely different appearance and motion characteristics: *(i)* a car "passing by"; and *(ii)* a man crossing the highway. These two events are different: in appearance; speed; direction of motion; and duration.

S2 is a relatively large sequence (6000 frames) that depicts several instances of moving vehicles and three abnormal events: *(i)* a car reversing in the auxiliary lane; *(ii)* the same car moving forward in the auxiliary lane; and *(iii)* a man crossing the road. These abnormal events (Fig. 4.2) have a number of characteristics that differentiate them from the normal instances. The "man crossing" and "car reversing" events are slow actions (*i.e.* the appearance difference between frames is low). Furthermore, the location where the abnormal car events are taking place is a sparsely occupied area in the sequence (*i.e.* auxiliary lane).

S3 is a smaller sequence (from the same region as S2) that includes in addition to the moving vehicles three abnormal events (Fig. 4.2): *(i)* two instances of the man cros-

Seq.	Number of Frames	ROI Size (pixels)	Description	Frames
S1	451	54x51	Car passing Man crossing	45-75 146-207
S2	6000	54x51	Car reversing Car using auxiliary lane Man crossing	762-881 1917-1961 3303-3353
S3	2900	54x51	Man crossing Car crossing (synthetic) Man crossing	211-261 415-496 2194-2246
S4	1057	54x51	Car using auxiliary lane	173-848

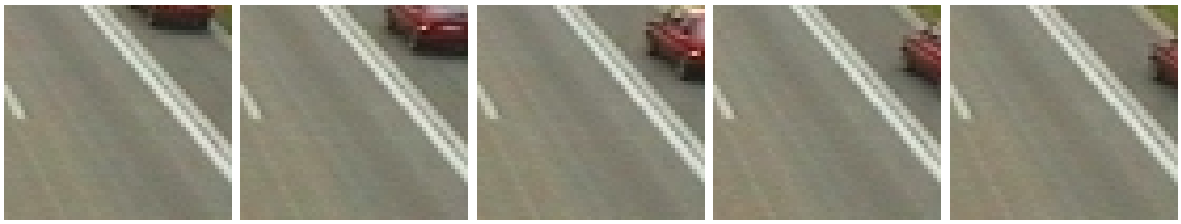
Table 4.1: Highway dataset description.

sing the road; and *(ii)* an abnormal event generated using an static image of a car, slowly moving from right to left. The slowly moving car has been scaled to 70% of the original size to produce less distortion. The resulting scene is similar in speed and distortion to the one of the pedestrian.

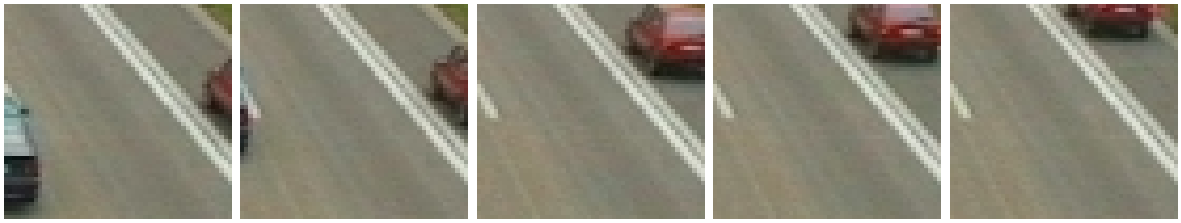
S4 is a small scene that intentionally relaxes the assumption on the number of objects in the scene. The uncommon event in this case is a car using the auxiliary lane, while other vehicles are moving in the highway (Fig. 4.3).

Table 4.1 summarises the contents of the four sequences. In all the sequences normal events (actions) are composed of vehicles that move from the bottom right hand corner to the top left hand corner of the ROI and they constitute the majority of the actions in the scene (Fig. 4.4).

Sequence S1 is used for testing and evaluating the different subspace learning methods in a very simple scenario. S1 consists of only two events, thus any subspace learning method should be able to map the feature vectors in a low-dimensional space in which they are visually separated. However, sequences S2 and S3 are built to help identify suitable methods to be used as a part of an abnormality detection framework. The normal actions in S2 and S3 are numerous and consist of a variety of objects (different appearance). Furthermore, the abnormal events do not always follow the "good behaviour" of linearity (in the case of articulated objects) and high-variance (in the case of slow moving objects) yet are real and valid uncommon events that can take place in a highway scenario. An unsu-



(a) S2



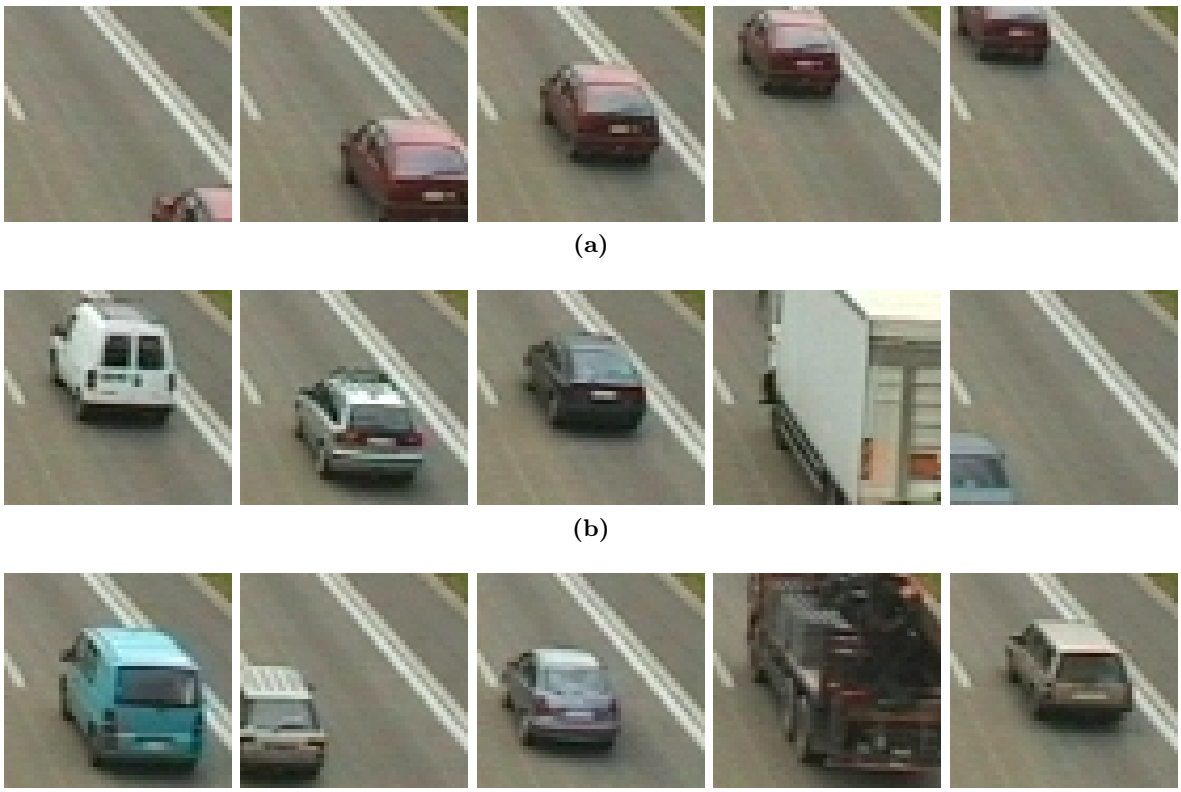
(b) S2

(c) S2 and S3

(d) S2

Figure 4.2: Key frames of abnormal events: (a) car reversing in the auxiliary lane, (b) car using the auxiliary lane and (c) man crossing and (d) synthetic event of the pasted car.

Figure 4.3: Key frames of the car using the auxiliary lane in sequence S4

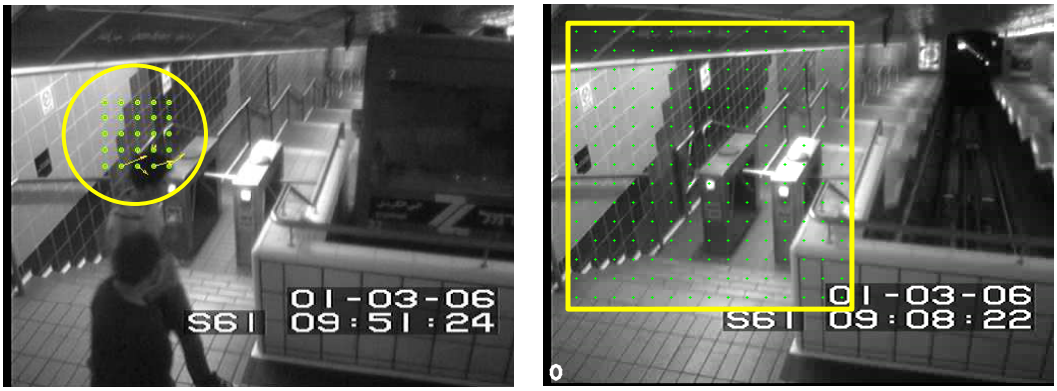


(a)

(b)

(c)

Figure 4.4: Normal instances in the Highway ROI: (a) key frames of normal action patterns from sequence S2 and (b,c) sample frames of vehicles from sequences S2 and S3.



(a) S5A **(b)** S5B

Figure 4.5: Sample frames of sequence S5 depicting a scene form the exit of an underground train station and the selected ROI: (a) S5A for a single detector, (b) S5B for the multi-detector framework.

pervised subspace learning method should be able to handle (computational complexity) the large number of high-dimensional vectors and group the normal instances in a compact cluster. At the same time, it should map the abnormal instances away from the normal cluster. Finally, sequence S4 helps to identify the issues and problems that arise when assumptions on the number of objects and the ratio of normal to abnormal patterns in the sequence are relaxed.

4.2.2 Multi-object dataset

To investigate the performance and compare the proposed subspace-aware local detectors, a challenging sequence (S5) from the dataset used in the work of Adam et al. [13] is selected. It is a scene from an underground train station (64900 frames of size 512x384, 25 fps) where the camera oversees the exit turnstiles of the platform (Fig. 4.5). During the normal motion flow in S5, people that arrived with the train are exiting the platform through the turnstiles and turn left or right on the corridor.

The main characteristics of this sequence are:

1. The moving objects are articulated (humans) and appear in a variety of crowd densities from a single object to multiple objects following different behaviour patterns.

2. The camera placement is such that heavy perspective distortion is present.

3. Partial dynamic occlusions are highly probable.

4. It has been designed specifically for the abnormality detection problem thus the abnormal events are closer to real-world scenarios. The normal events are numerous and the abnormal are rare.

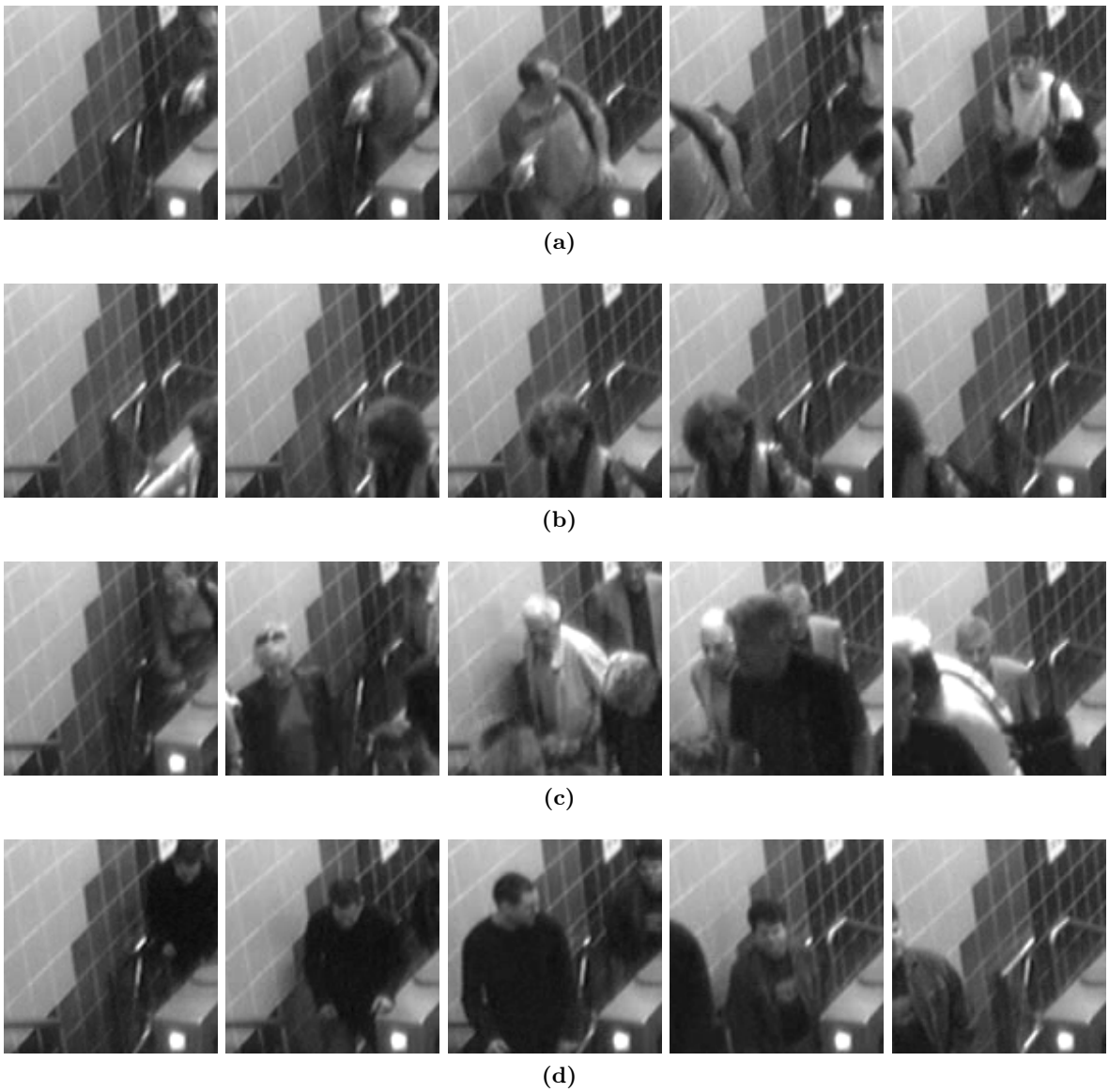
5. Behaviour patterns are not global and have spatial affinity (stairs, corridor) and, as a result, abnormal events can also be local.

Alternative datasets in the literature only include some of these properties. For example, sequences used in work based on object detection and tracking might be multi-object but depict rigid objects with the camera placed at a distance (Figs. 2.2(a,b), 2.4(b) and 2.6). Even when the camera setup is such that the action is taking place in close proximity to the camera sensor, the events are restricted and global (Fig. 2.3), there is no perspective distortion (Fig. 2.5) or the crowd density is relatively low (Figs.2.3 (a,b)).

In order to evaluate the proposed subspace based abnormality detector we define a region of interest as presented in Figure 4.5 (a). This ROI was selected because it has a good ratio of normal to abnormal events with a wide range of duration. Given that the first 7500 frames of the video are normal, the abnormal events consist of people entering the ROI going down the stairs (wrong way) and people jumping. These events involve abnormality that is relative to both speed and angle of motion. Figure 4.6 and 4.7 present sample frames from the normal and abnormal instances that exist in S5A.

The complete multi-detector framework is tested on a larger ROI (S5B) as shown in Figure 4.5(b). Note that S5B is similar to the one selected in the work of [13]. Adam et al. [13] describe as abnormal events only a small number of scripted instances that they have performed and they are located "around" a centre frame without specific duration. In contrast, Kim and Grauman [14], using the same dataset but considering the complete frame, increase the number of abnormal events to 19. This ground-truth, in addition to the abnormal events in [13], includes loitering (*i.e.* a person waiting), people changing their direction (*e.g.* start moving left then moving right) and even motion caused by children. Creating the ground-truth in larger regions is subjective and remains an open problem, it is thus difficult to understand what is normal and abnormal in these cases where similar abnormal actions are ignored [13] or the majority of behaviours is considered abnormal [14].

Based on our definition of the abnormality detection problem, abnormality inferred



(a)

(b)

(c)

(d)

Figure 4.6: Sample key frames of normal instances as they appear in S5A.

(a) wrong way

(b) wrong way

(c) jump

(d) wrong way (small distortion)

Figure 4.7: Sample key frames of abnormal instances as they appear in S5A. (Key: people performing abnormal actions are shown with a bounding box for illustration purposes)

based on the abstract annotation of normal behaviour provided by the operator and the assumption that such actions constitute the majority of behaviours in the scene. To this extent, we have summarised the main paths of motion that exist in the first 7500 frames (annotated clip) and considered abnormal those actions that have not been observed before. The ground-truth for the S5B area where the local detectors are deployed in sequence S5 was produced in two steps:

First, the full image frame ground-truth was created using the Viper tool [75] where we manually defined the bounding box of the perpetrator (the person who performs the abnormal action). The events start when the action is abnormal and stop when the associated object stops moving or reverts to normal behaviour. It should be noted that if the person moves in an abnormal way at a later instance then a new event is declared. Because of these rules a number of events are separated by a small gap (e.g the person stops moving for a number of frames) and thus two events are recorded. The abnormal events are separated into; *(i)* moving down the stairs into the platform; and *(ii)* moving from right to left in the corridor. There are no miscellaneous events (*e.g.* children moving).

Second, the events that take place in the S5B area of the scene were found automatically. The algorithm performs a proximity check of the bounding box in the full image frame ground-truth and the S5B area boundary. An event exists in S5B if a bounding box from the full ground-truth is closer than two pixels to the boundary of S5B. When two abnormal events take place in S5B at the same time, they are merged into one abnormal event. Note that, the bounding boxes in the full frame ground-truth are only used to find the start and stop time instances of the event in the S5B area and the final ground-truth does not contain any bounding box information. Figures 4.8 and 4.9 provide representative samples of the *normal* and *abnormal* behaviours that are produced by these rules. While these rules have their limitations, *e.g.* spatially independent events might be merged; they provide a set of rules that can be easily implemented to provide performance evaluation. The resulting ground-truth for S5B consists of 14 events.

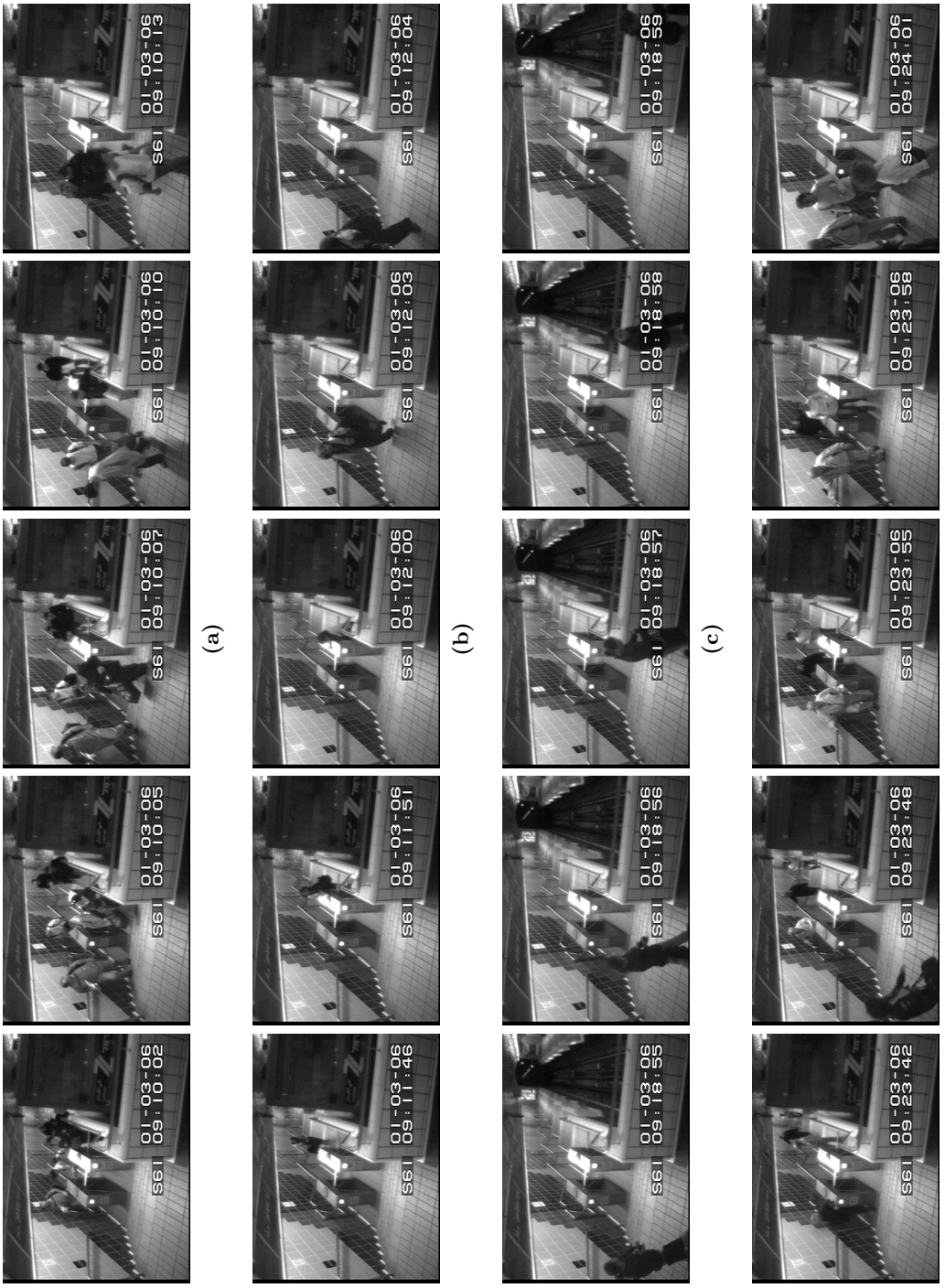
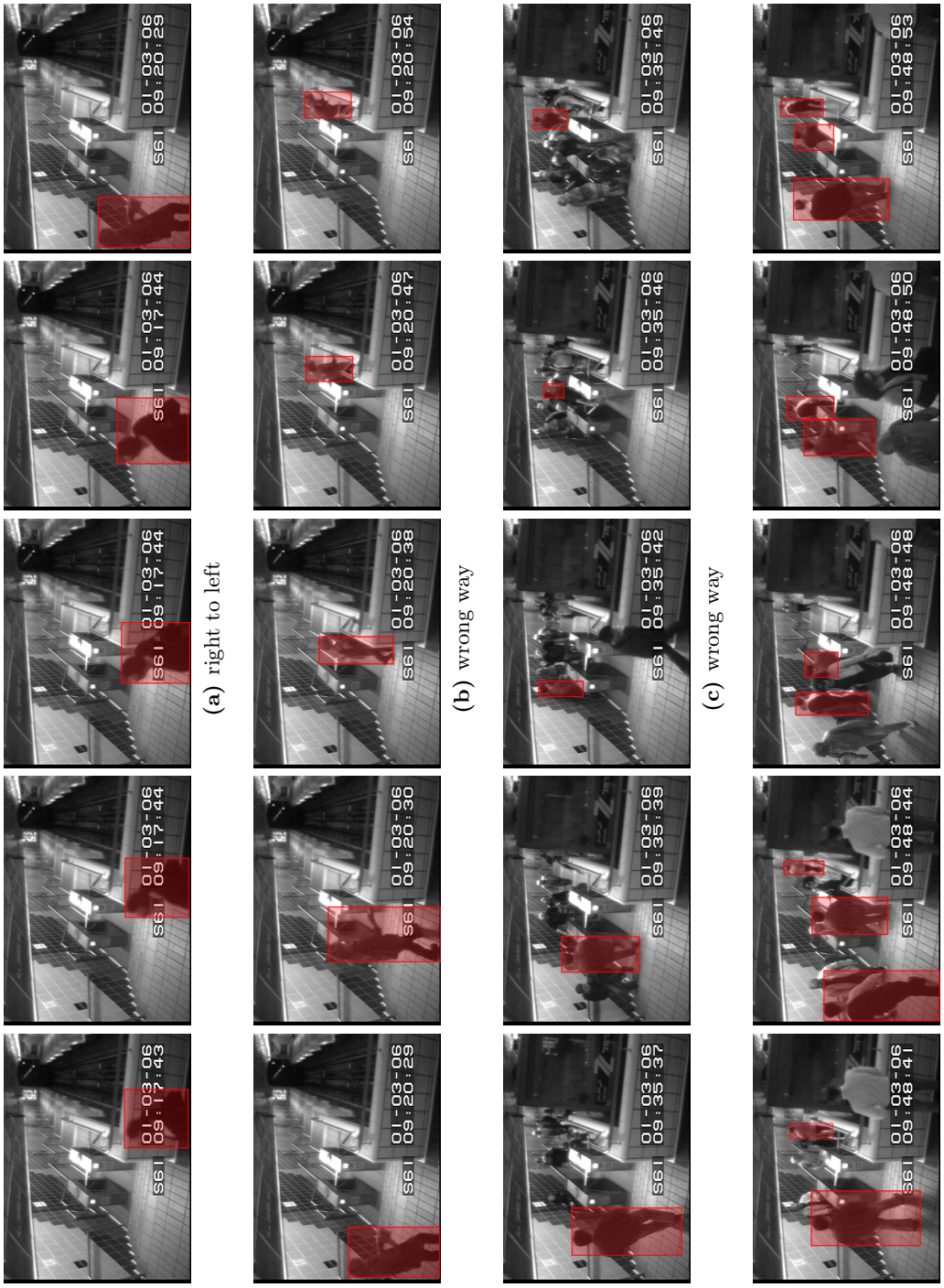


Figure 4.8: Sample key-frames of normal instance in S5 given that the first 7500 frames are normal.



(a) right to left **(b)** wrong way **(c)** wrong way **(d)** wrong way (multiple instances)

Figure 4.9: Sample key-frames of abnormal instances in S5 given that the first 7500 frames are abnormal. (Key: people performing abnormal actions are shown with a bounding box for illustration purposes)

Method	Graph	Weight
LE-A	minimum k-NN	τ_∞
LE-B	minimum k-NN	τ_μ
LE-C	temporal ε-graph	τ_∞
LE-D	temporal ε-graph	τ_μ

Table 4.2: Parameter variations used for LE in 3D projections. (Key: for the τ value definition please refer to 3.3.3)

4.3 Subspace learning

In this section, we discuss and compare the off-line subspace learning approach (see subsection 3.3.4) with alternative state-of-the art dimensionality reduction algorithms on mapping single-object video sequences in a low-dimensional space. Appearance-based features extracted from the video sequences are used to construct a low-dimensional mapping representing the acquired feature vectors in the ROI. This makes it possible to map and visualise the actions in the scene, as represented by the feature vectors, in a low-dimensional space. The off-line subspace learning approach is evaluated using the suggested graphs (minimum k-NN and ε-temporal graph) and compared with PCA, MDS, Isomap and MVU. The aim is to identify the advantages of graph-based dimensionality reduction on representing event patterns and to demonstrate that the proposed subspace learning method based on LE is better suited for projecting abnormal instances that exist in the videos.

4.3.1 Preliminaries

The proposed approach is based on the off-line method described in 3.3.4 using LE. The different graph and value τ (defined in subsection 3.3.3) combinations used are denoted with LE-(A, B, C and D) and are given in Table 4.2. For Isomap and MVU, we used the same neighbourhood size that was calculated for the LE-A variation (other parameters are set to default values). MVU was used in combination with the C library for SemiDefinite Programing (CSDP) [76]. However, projections of the S2 and S3 sequences were acquired using the alternative incremental approach of MVU, using 2000 initial vectors, due to limitations imposed by the available memory on the test system.

In order to compare the different subspace learning methods we visualise the results in a three dimensional subspace. Given a video sequence we extract the pixels in the region

of interest and associate them with a single detector region R. The appearance-based observations are constructed by concatenating the pixel colour values that are enclosed in R at time t to form a high-dimensional observation vector $\mathbf{o}_t \in \mathbb{R}^l$. The number of dimensions is related to the size of the region in pixels and the number of colour channels that are describing each pixel. Thus a 100×100 colour (RGB) region will result to a feature with $l = 100^2 * 3 = 30000$ dimensions. All the projection methods are applied using the high-dimensional appearance based vectors $\mathcal{S}$ (where $\mathbf{s}_t = \mathbf{o}_t$) to produce the three dimensional mapping $\mathcal{Y}$. The events are colour coded manually in the plots where normal events are coloured with the same colour, except for S1 where both actions (car and man) have different colours.

The projections are evaluated on abnormal event *separation*. This term refers to the difference in the low-dimensional mappings between normal and abnormal instances. Please note that the term *separation* as used in this context is not equivalent to detection since no classifier is utilised to achieve any labelling. Visually we define the abnormal events *separation* as the effect that the dimensionality reduction methods have in mapping the abnormal events away from the normal while considering at the same time how compact the mapping of the normal instances is. Performance is also relative to the amount of overlap that exists between the normal and abnormal classes in the low-dimensional projection.

Numerically, *separation* is evaluated based on how well the abnormal instances fit within the distribution of the normal class (in the projected space). Thus, given the low-dimensional projection $\mathcal{Y}$ of the feature vectors $\mathcal{S}$ in the single region R and the projection $\breve{\mathcal{X}}$ of the set of labelled normal vectors $\mathcal{X}$, we compute the score $f_{sap}(\mathbf{y}_t, \breve{\mathcal{X}})$ of an instance $\mathbf{y}_t \in \mathcal{Y}$ using the Mahalanobis distance:

$$f_{sep}(\mathbf{y}_t, \breve{\mathcal{X}}) = (\mathbf{y}_t - \boldsymbol{\mu})^T C_{\breve{\mathcal{X}}}^{-1} (\mathbf{y}_t - \boldsymbol{\mu}), \tag{4.1}$$

where C is the covariance matrix of the projected normal vectors $\breve{\mathcal{X}}$ and $\boldsymbol{\mu}$ is their mean. To evaluate the separation of a set of feature vectors that belong to an abnormal event $\mathbf{y}_t^{(i)} \in \mathcal{Y}^{(i)} \subset \mathcal{Y}$ (where $i \in \{1, 2, \ldots\}$ identifies the different abnormal events), we can use the *average separation index* (ASI) calculated over all the elements of $\mathcal{Y}^{(i)}$,

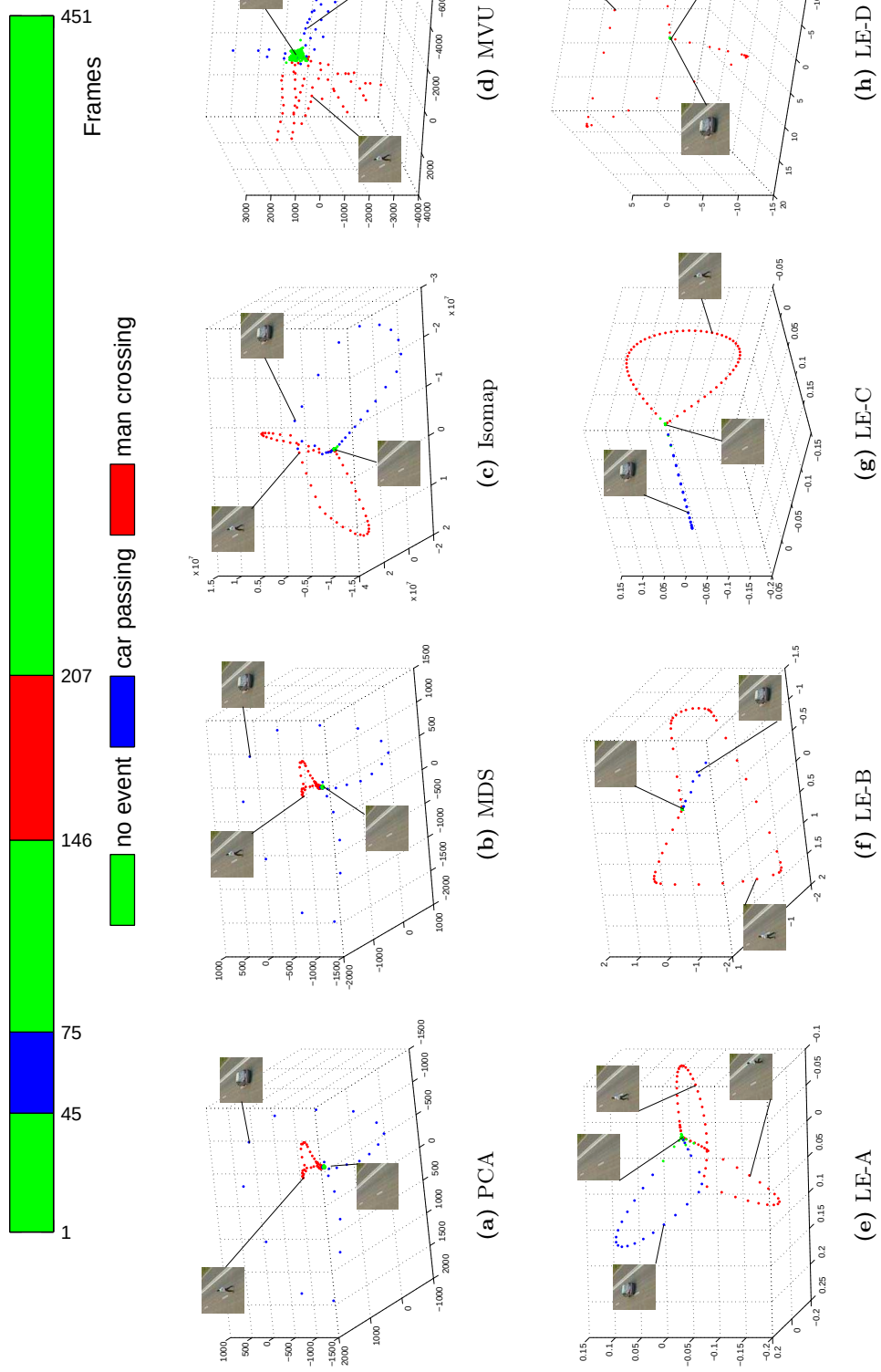
$$f_{ASI}(\mathcal{Y}^{(i)}) = \frac{1}{n_{\mathcal{Y}^{(i)}}} \sum \{ f_{sep}(\mathbf{y}_t^{(i)}, \breve{\mathcal{X}}_{all}) \mid \forall \mathbf{y}_t \in \mathcal{Y}^{(i)} \}, \tag{4.2}$$

where $n_{\mathcal{Y}^{(i)}}$ is the number of vectors in $\mathcal{Y}^{(i)}$ and $\breve{\mathcal{X}}_{all}$ is the set which contains *all* of the normal instances as defined in the ground-truth. ASI provides a measure of the relative distance of the abnormal instances from the class of normal, based on the assumption of a single Gaussian distribution for the normal pattern class. Small values (negative in the logarithmic scale) denote a poor separation. In this case, the uncommon event vectors are inside the point cloud of the *normal* vectors. Values close to 10 account for event projections that have a fair degree of overlapping with the *normal* set. Finally, values greater than 10 correspond to projections where the highlighted events are further away from the normal vectors.

4.3.2 Event representation

We apply the dimensionality reduction methods (*i.e.* PCA, MDS, Isomap, MVU and LE) on the appearance based observation vectors from sequence S1, which contain only two events, (Fig. 4.10). The PCA projection separates the two events. The "man crossing" event generates a small loop inside the bigger loop of the "car passing" event. MDS also provides a similar embedding. Isomap results in a projection where different events are mapped in different loops of points (trajectories). MVU projects the two events in opposite directions but the path that the objects follow in this space is not as clear as in the case of the rest of the algorithms. Changes in angle of the trace of the man correspond to changes in the shape (size) of the man while he runs across the highway.

The LE variation and Isomap have the same appearance (Fig. 4.10). The events are placed in loops that start and end in the *null event* where no action takes place in the scene. The LE-B variation mapping has a topology similar to the PCA projection, but reveals additional interesting features. The pedestrian performs a periodic movement while crossing the road from right to left and periodically changes his shape and size, thus the sequential observation vectors in the region have maxima and minima in distance between them. The periodic movement of the man generates angles and density differences in the LE projection similar to MVU. If we use the temporal ε-graph (*i.e.* setup LE-C) then the two events are better separated. Since the "man crossing" is a slow action and the "car



(a) PCA **(b)** MDS **(c)** Isomap **(d)** MVU

(e) LE-A **(f)** LE-B **(g)** LE-C **(h)** LE-D

Figure 4.10: Dimensionality reduction algorithms applied on video sequence S1 using colour frames. (Key: vectors are colour coded manually in the plots based on the ground truth to aid the visualisation)

passing" is a fast action, there are no connections between them in the temporal ε-graph thus, LE will make sure that these vectors are as far away as possible in the projections. Additionally, when we use LE-D, the vectors representing the fast moving car collapse to a small area close to the "no event" vectors, while the vectors associated with the pedestrian generate a loop.

PCA (Fig. 4.10(a)) provides a descriptive summary of the actions in the region of the video. Isomap forces each action to project in a distinct loop. LE gives a fusion of these properties in various degrees depending on the graph and weighting schemes used, from the Isomap projection (Fig 4.10(e)) to the PCA or the MVU projections (Fig. 4.10f) and even further to extreme projections (Fig. 4.10(g,h)) when using the temporal ε-graph.

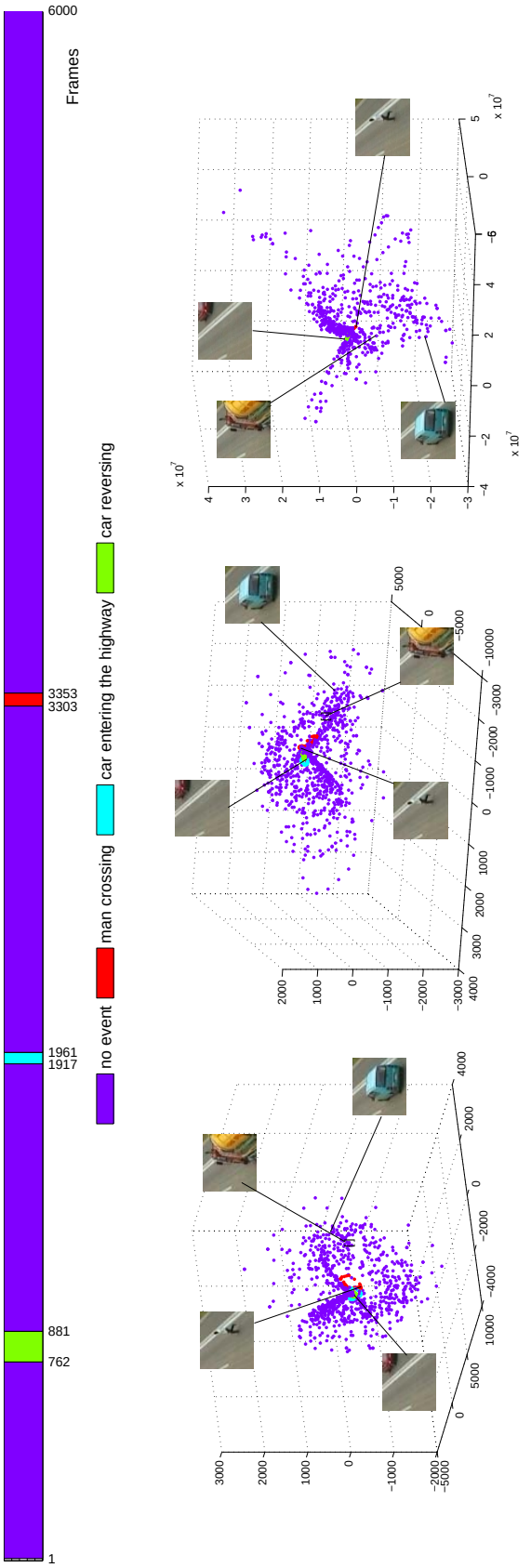
As seen from the results, LE is able to separate the two events in the sequence (cause by a car and a man) in the low-dimensional representation. Although the other dimensionality reduction algorithms perform well on this simple scenario, LE combines their distinct characteristics under a single framework.

4.3.3 Abnormal events

To further explore the advantages and limitations of LE as a means to acquire low-dimensional mappings that can be used for abnormal event detection, we apply the LE variations (LE-A and LE-B) to S2 and S3 (Figs. 4.11 and 4.12). LE-B and LE-D were not included because the application of the weighting scheme τ_μ resulted in an unstable (singular) solution of the eigenvalue problems, thus results could not be acquired (see subsection 4.3.5).

Figure 4.11 compares the mapping algorithms applied on an extended video, S2. The *linear dimensionality reduction* algorithms, PCA and MDS, fail to consider the non-linear correlation of the feature vectors, as they use only the variance or distance information in the high dimensional space. The internal structure is also not discovered by global graph-based dimensionality reduction algorithms (*i.e.* Isomap and MVU in Figure 4.10).

In contrast, when LE-A (Fig. 4.11(e)) is used with sequence S2, the loops that occur describe the car (green and light blue dots) but not the man crossing the road. The normal vectors are distributed on a triangular surface: the lower corner holds the vectors that show a car (or truck) in the centre of the ROI; the two top corners correspond to



(a) PCA (b) MDS (c) Isomap

(d) MVU (e) LE-A (f) LE-C

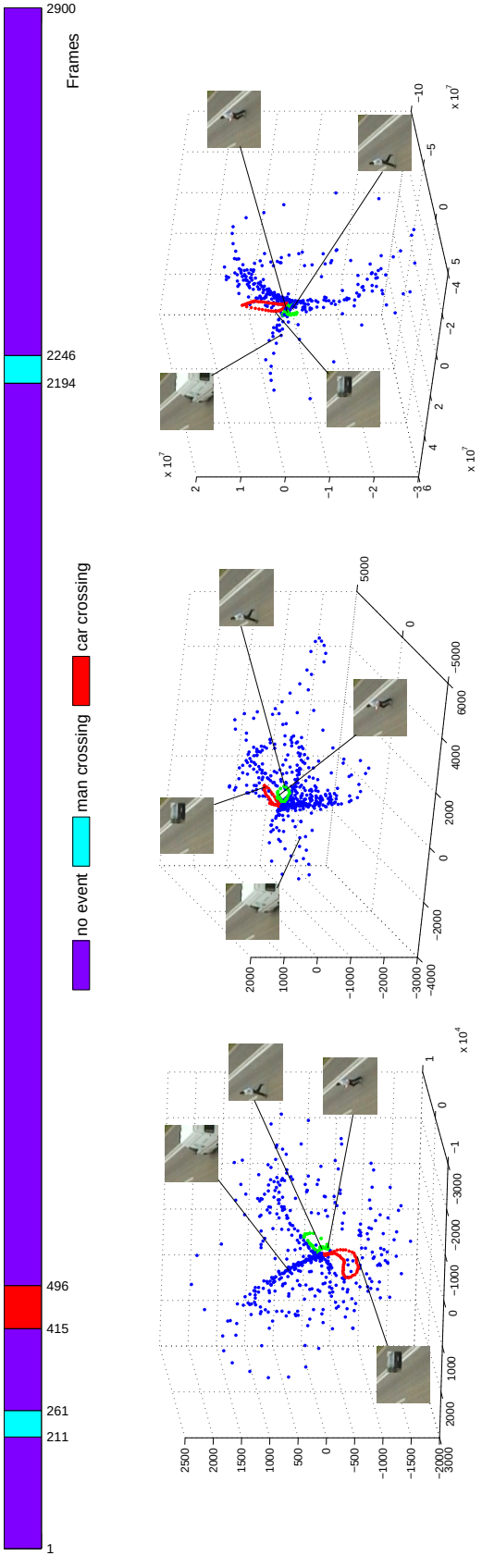
Figure 4.11: Comparison of projections of video sequence S2, using colour frames, to highlight the uncommon events. (Key: vectors are colour coded manually in the plots based on the ground truth to aid the visualisation)

vectors under different illumination conditions and some small movement of the camera; the area in between is occupied by the vectors where the cars move in and out of view. The trace associated with each vehicle follows a similar path; it starts at the top corner, moves down to the low corner and returns to the original position when it is out of the view.

Using the temporal ε-graph (*i.e.* LE-C) in sequence S2 changes the mapping in a drastic way (Fig. 4.11(f)). The temporal ε-graph creation rules only penalise the connectivity of vectors which change slowly. Both the "man crossing" and "car in reverse" events are slow thus in the graph the associated vectors are *lonely* nodes. However, when the car is moving forward the movement is fast enough that the average distortion of the associated vectors is large thus these nodes are *popular* in the graph. The same effect applies to the vectors of the normal patterns where the cars move fast in the scene. LE embeds the temporal ε-graph so that *popular* nodes are mapped compactly together and the *lonely* nodes are mapped far away from the area of the normal patterns.

The S3 sequence is another example that demonstrates the effect of the temporal ε-graph for slow changing actions. In this case the sequence combines an artificial (*i.e.* "car crossing") and two instances of a real-life unusual event (*i.e.* "man crossing"). The alternative global subspace projections (PCA, MDS, Isomap) cannot provide any visual clues about the highlighted events, while LE-A projection (Fig. 4.12(e)) shows the highlighted events in separate loops. Similarly to the S2 sequence (Fig. 4.12 (e,f)), the usual events are visualised in a triangular shaped surface and the unusual events form loops out of that surface. MVU also provides good visual separation for the abnormal events making the "car crossing" event is easier to spot.

The results for sequence S3 are improved by applying LE-C (Fig. 4.12(f)). The instances of the abnormal events are mapped so that the projected points follow a straight line. Although the abnormal actions (*i.e.* "man crossing" and "car crossing") events are both slow, they are not embedded close together in the low-dimensional subspace. The difference between the associated feature vectors is large enough that the vectors are not connected in the graph, thus in the LE projection each abnormal action is placed far away from the other.



(a) PCA (b) MDS (c) Isomap

(d) MVU (e) LE-A (f) LE-C

Figure 4.12: Comparison of projections of video sequence S3, using colour to highlight the uncommon events. (Key: vectors are colour coded manually in the plots based on the ground truth to aid the visualisation)

Algorithms	S2			S3	
	man crossing	car reversing	car using the auxiliary lane	man crossing	car crossing
PCA	-0.20	-0.93	-0.40	0.25	0.45
MDS	-0.20	-0.93	-0.40	0.25	0.45
Isomap	-0.98	-0.52	-0.46	0.16	0.57
MVU	0.57	2.50	2.44	0.51	1.75
LE-A	1.15	4.48	4.14	0.43	0.79
LE-C	6.38	7.32	-0.39	2.62	8.05

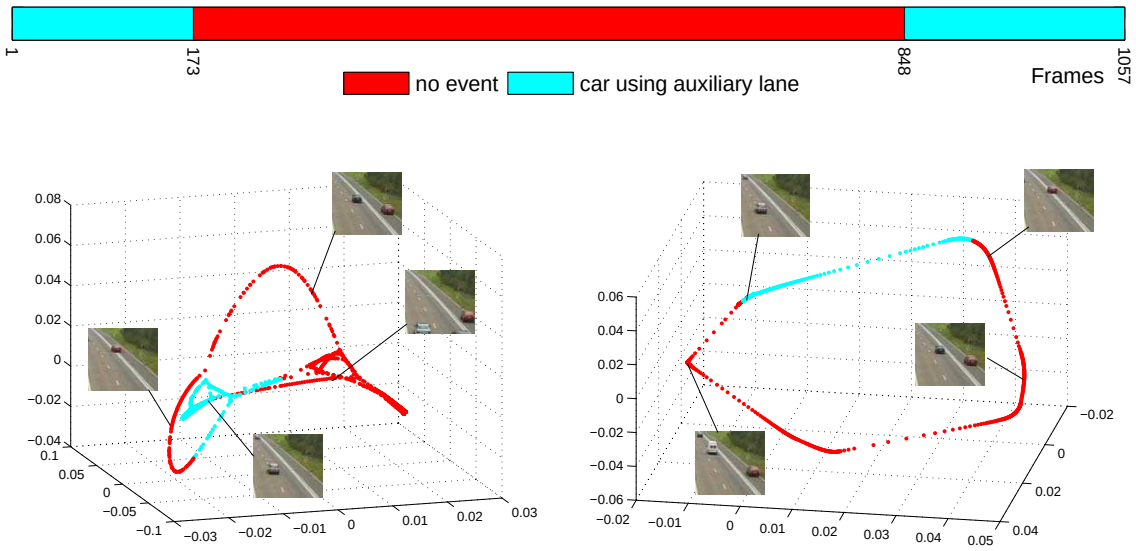
Table 4.3: The *average separation index* (Eq. 4.2) for the abnormal events in sequences S2 and S3. (Key: the values are presented in the logarithmic scale)

4.3.4 Event separation

The existence of visible structures in the projections, which can provide us with clues about the characteristics of the motion/action in the scene, is a desired property for event representation. However, for abnormality detection the *separation* in the low-dimensional subspace of the uncommon events from the normal instances is also important. This can be evaluated by calculating the ASI value (Eq. 4.2) of each abnormal event against all the normal instances in the projection.

Table 4.3 shows the ASI values for the abnormal events of the projections (using PCA, MDS, Isomap, MVU, LE-A and LE-C) on sequences S2 and S3. The results show that LE, especially with the LE-C set of parameters is able to separate the abnormal events very well. The only exception is that the "car using the auxiliary lane" event in S2 has a very low score. The object in this event is moving fast thus the main assumption of the temporal ε-graph does not hold. However, using the minimum-k-NN graph (*i.e.* LE-A), it is possible to provide a better separation for the "car using the auxiliary lane" event but on average the performance is lower than the LE-C setup. Alternative linear (*i.e.* PCA and MDS) and non-linear (*i.e* Isomap) methods fail to separate the uncommon events. The method that provides comparable results with LE-A is MVU. However, the computational complexity of MVU is very high and does not scale well with the number of feature vectors (*i.e.* long sequences).

We can argue, that based on the visual and quantitative comparison, LE is more suitable to provide a low-dimensional space that can be used for training of classifiers and



(a) LE-C **(b)** LE-D

Figure 4.13: Application of LE-A and LE-D on a multiple object scene (S4). (Key: vectors are colour coded manually in the plots based on the ground truth to aid the visualisation)

detection of abnormal events. The conclusions from the comparison in 4.3 can be extended to the subspace learning algorithm SR-LPP, which provides an approximation to the LE projection. This is due to the fact that the SR-LPP method is based on Eq. (3.13), which is equivalent with the eigenproblem solved by LE (*i.e.* Eq. (3.10)).

4.3.5 Issues in subspace learning

In the previous subsections (*i.e.* 4.3.2, 4.3.3 and 4.3.4), results were presented using sequences with one moving object in the ROI and thus the number of possible motion patterns was limited. However, when multiple objects move inside the ROI, the number of possible patterns increases exponentially. In such cases the assumption that the normal patterns classes are the majority and that there are numerous examples of normal actions in the samples used for subspace learning, is difficult to satisfy. To demonstrate this issue, a multi-object sequence with a small number of vectors is used (*i.e.* S4).

The application of the Laplacian Eigenmaps is presented in Figure 4.13. Every new object in the video creates a new path in the projected space (Fig.4.13(a)). These paths cross each other in various places corresponding to vectors in the sequence with similar content. When LE-D setup is used, the complete sequence is projected as one loop (Fig. 4.13(b)).

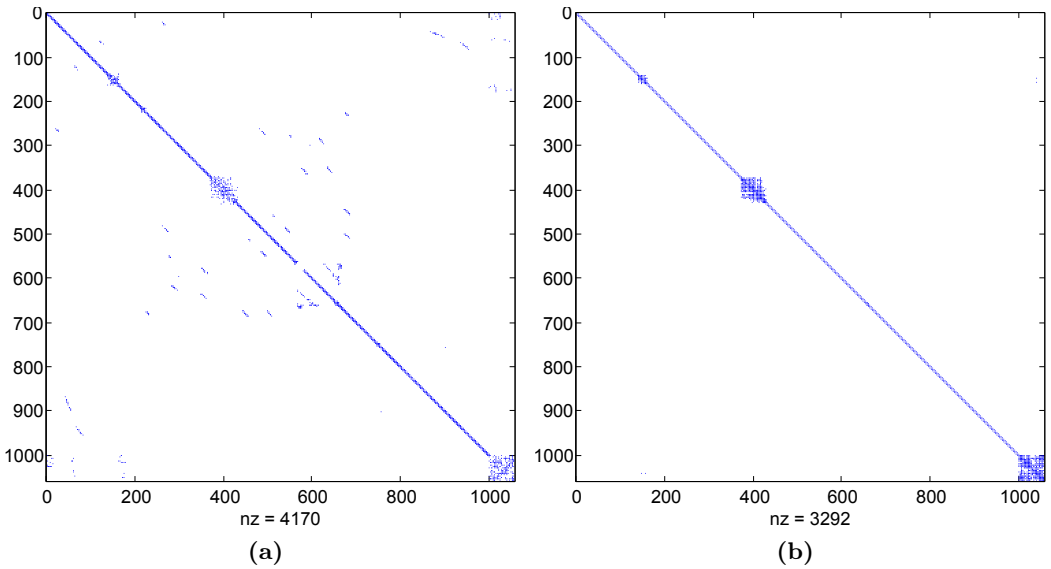


Figure 4.14: Nearest neighbours graph when multiple patterns exist and the number of vectors is not enough: (a) minimum-k-NN graph and (b) temporal ε-graph of S4 do not have any significant structure away from the main diagonal.

However, there is no visible grouping of patterns in normal and abnormal classes as seen in Figures 4.11 and 4.12.

The problem is associated with the information that is retained by the neighbourhood graph produced from the appearance-based vectors constructed by concatenating the pixel values in the ROI of the sequence S4 (Fig. 4.14). When the action patterns are not properly represented (*i.e.* there are not enough examples of the different patterns in the sequence), the graph is fully connected using only a small number of nearest neighbours and these connections mainly correspond to the time-line direction of the video stream. To avoid cases in which the graph and thus the discovered subspace does not provide useful information about the abnormality detection problem, we need to: *(i)* increase the subspace learning samples; or *(ii)* reduce the size of the ROI to lower the complexity and number of the action patterns.

The first requirement is easier to achieve since the proposed subspace learning process is unsupervised (no need to annotate instances) and scales well with the number of vectors. The second requirement places restrictions on the size of the ROI which is less intuitive if there is no information about the size of the moving objects. Nevertheless, the graph structure allows us to detect this representation problem prior to the process of training and validating the abnormality detection system. When there are not enough examples

of the normal patterns the neighbourhood graph structure provides visual clues (structure similar to Figure 4.14) and thus we can act accordingly to avoid the representation issue (*e.g.* provide more samples to the subspace learning module).

Another issue is associated with the graph weighting schema and the numerical stability of the solution. Experiments in sequences S2 and S3 have shown that the LE solution is sensitive to the selected value of τ. In these cases the value $\tau = \tau_\mu$ resulted in badly scaled matrices and the eigen-problem became singular. Thus the eigensolver was not able to provide a numerically reliable solution. The problem has been addressed in later experiments (section 4.4) where using $\tau = \tau_o$ has shown that the numerical stability of the solution in sequence S3, using a variety of features, has improved. The next section will investigate the effect of using different low-level features in order to detect the abnormal instances in a region of interest.

4.4 Feature comparison

In the previous section we have demonstrated that the proposed off-line graph-based dimensionality reduction methodology is able to provide a low-dimensional subspace where abnormal events are mapped far away from the class of normal patterns. The next step is to study commonly used features that can be extracted from the video sequence and to evaluate their performance for abnormality detection. To this extent we evaluate the abnormality detection performance of appearance, change detection and motion vector based feature vectors, for subspace-aware detectors. Even though each feature has a number of intrinsic limitations, the goal is to identify features that are suitable for reliable and low-complexity independent abnormality detectors in video surveillance scenarios.

4.4.1 Preliminaries

The different types of extracted features are studied using sequence S3 that has a variety of moving vehicles (*e.g.* cars, vans and trucks numbering) 20 occurrences in total thus there is wide variety in speeds and appearance of the moving objects. The sequence is first preprocessed by a low-pass blurring filter based on a 5-pixel square sliding window, to remove noise. We assume a single region R that is equal to the ROI. We then extract the following features for comparison:

	Pixels		Change detection mask		Motion vectors	
Colour space	RGB	gray	RGB	gray	gray	gray
Low pass	5×5	5×5	5×5	5×5	5×5	5×5
Threshold	–	–	10	8	–	–
Block size	–	–	–	–	5×5	5×5
Maximum displacement	–	–	–	–	5	10
Dimensionality	8424	2808	2808	2808	200	200
Setup	A1	A2	B1	B2	C1	C2

Table 4.4: Experimental setup parameters for feature comparison.

1. The colour and grayscale *appearance based* observations (A1 and A2) are produced by concatenating the pixel values in the region R.

2. The *change detection mask* is a binary matrix $C_t = [c_{i,j}(t)]_{i_h \times i_w}$ where inactive "0" pixels are considered background pixels and the active "1" pixels are associated with foreground objects. The size of the resulting binary image is equal to the original image and is calculated by thresholding the absolute pixel-wise difference $\widetilde{I}_t$ of I_t and I_{t-1},

$$c_{i,j}(t) = \begin{cases} 1 & \text{if } (\tilde{I}_t^R(i,j) > th_c) \wedge (\tilde{I}_t^G(i,j) > th_c) \wedge (\tilde{I}_t^B(i,j) > th_c) \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

$$\text{with } i = 1, \ldots i_h \text{ and } j = 1, \ldots i_w$$

For grayscale images I_t^Y, I_{t-1}^Y (produced by Eq. 3.1) and their absolute pixel-wise difference $\tilde{I}_t^Y$, the equation is simplified to,

$$c_{i,j}(t) = \begin{cases} 1 & \text{if } (\tilde{I}_t^Y(i,j) > th_c) \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

$$\text{with } i = 1, \ldots i_h \text{ and } j = 1, \ldots i_w$$

The threshold th_c is selected by visually inspecting the output for a part of the video sequence and selecting th_c that maximises the detection of foreground pixels that belong to moving objects while minimising noise (*i.e.* pixels that do not belong

to moving objects). The observation vectors are constructed by concatenating the pixel values in C_t into a multi-dimensional column vector. Table 4.4 presents the parameters sets for the colour (B1) and grayscale (B2) setups for the change detection masks.

3. The motion vectors are calculated using two setups C1 and C2, based on block matching (as described in 3.2). The setups differ on the size of the search area.

A summary of the setups and their parameters is provided in Table 4.4. We construct the proposed minimum k-NN graph using τ_o (defined in 3.3.3), for each setup using all the available vectors. The off-line subspace learning method (based on LE) then provides the low-dimensional projection. The desired characteristics of an ideal projection for abnormality detection should map the normal event vectors in a compact area. However, the abnormal events should be mapped far away from the normal event class. This *separation* (see subsection 4.3.1) of the normal and abnormal instances in the projections, is expected to improve the performance of novelty classifiers. To aid the visual comparison of the projections, the events are colour coded manually in the plots (based on the known ground-truth) where normal events are coloured with the same colour.

To evaluate the use of each feature to produce a mapping suitable for abnormality detection, we use the ASI score (*i.e.* Eq. (4.2)). We then divide each result by the number of subspace dimensions, to generate the *average separation index per dimension* (ASID) for each highlighted event,

$$f_{ASID}(\mathcal{Y}^{(i)}, m) = \frac{f_{ASI}(\mathcal{Y}^{(i)})}{m}, \tag{4.5}$$

where the sets $\mathcal{S}$, $\mathcal{Y}$, $\mathcal{X}_{all}$ and $\breve{\mathcal{X}}_{all}$ (*i.e.* the projection of $\mathcal{X}_{all}$) are associated with each specific setup (Table 4.4), $\mathcal{Y}^{(i)}$ (with $i \in \{1, 2\}$) is the subset containing one of the abnormal events (*i.e.* "man crossing" and "car crossing" for sequence S3) and m is the dimensionality of the low-dimensional representation. The value of m normalises the f_{ASI} value so that we can compare the values from the mappings for which the subspace dimensionality differs. Large values are an indication that the highlighted event is mapped far away from the normal action patterns.

The f_{ASID} value provides a measure of the *separation* (see of the abnormal instances against all the *normal* instances as defined in the ground-truth. However, in off-line detection scenarios the operator will only label a sample as normal $\mathcal{X} \subseteq \mathcal{S}$. Using Eq. (4.1) we can construct a low-complexity novelty classifier and evaluate the performance of a subspace-aware local detector,

$$\mathsf{G}(\breve{\mathbf{z}}_t) = \begin{cases} 0 & \text{if } f_{sep}(\breve{z}_t, \breve{\mathcal{X}}) \leq \theta \\ 1 & otherwise \end{cases}, \tag{4.6}$$

where $\breve{\mathbf{z}}_t$ are the unlabelled vectors and $\breve{\mathcal{X}}$ the labelled samples in the low-dimensional subspace. The training and detection is taking place on the low-dimensional representation provided by the off-line unsupervised subspace learning algorithm (Laplacian Eigenmaps).

Results are evaluated based on the Receiver Operating Characteristic (ROC) and Precision and Recall (PR) curves [77] produced by varying the threshold θ. Precision (PR) describes how certain we are that the detected abnormal instance is truly abnormal. Recall (RC) measures the probability that all the abnormal instances are detected. These measures are given by,

$$\begin{aligned} PR &= \frac{\text{True positive}}{\text{True positive} + \text{False positive}} \\ RC &= \frac{\text{True positive}}{\text{True positive} + \text{False negative}} \end{aligned} \tag{4.7}$$

ROC and PR curves provide complementary information especially when highly unbalanced datasets are classified. Furthermore, the quality of the classifier based on the ROC curve can also be associated with the scalar value of the Area Under Curve (AUC) of the ROC graph. AUC is defined as the area that exist under the ROC curve in the ROC graph and its range is $[0, 1]$, where 1 corresponds to the perfect classifier.

4.4.2 Visual comparison

In the three-dimensional projections (Fig. 4.15) each event is projected in a loop that starts and ends at the "no activity" feature vector. The main loop is produced by the normal events of the passing cars (blue colour). This is, however, better defined (more compact)

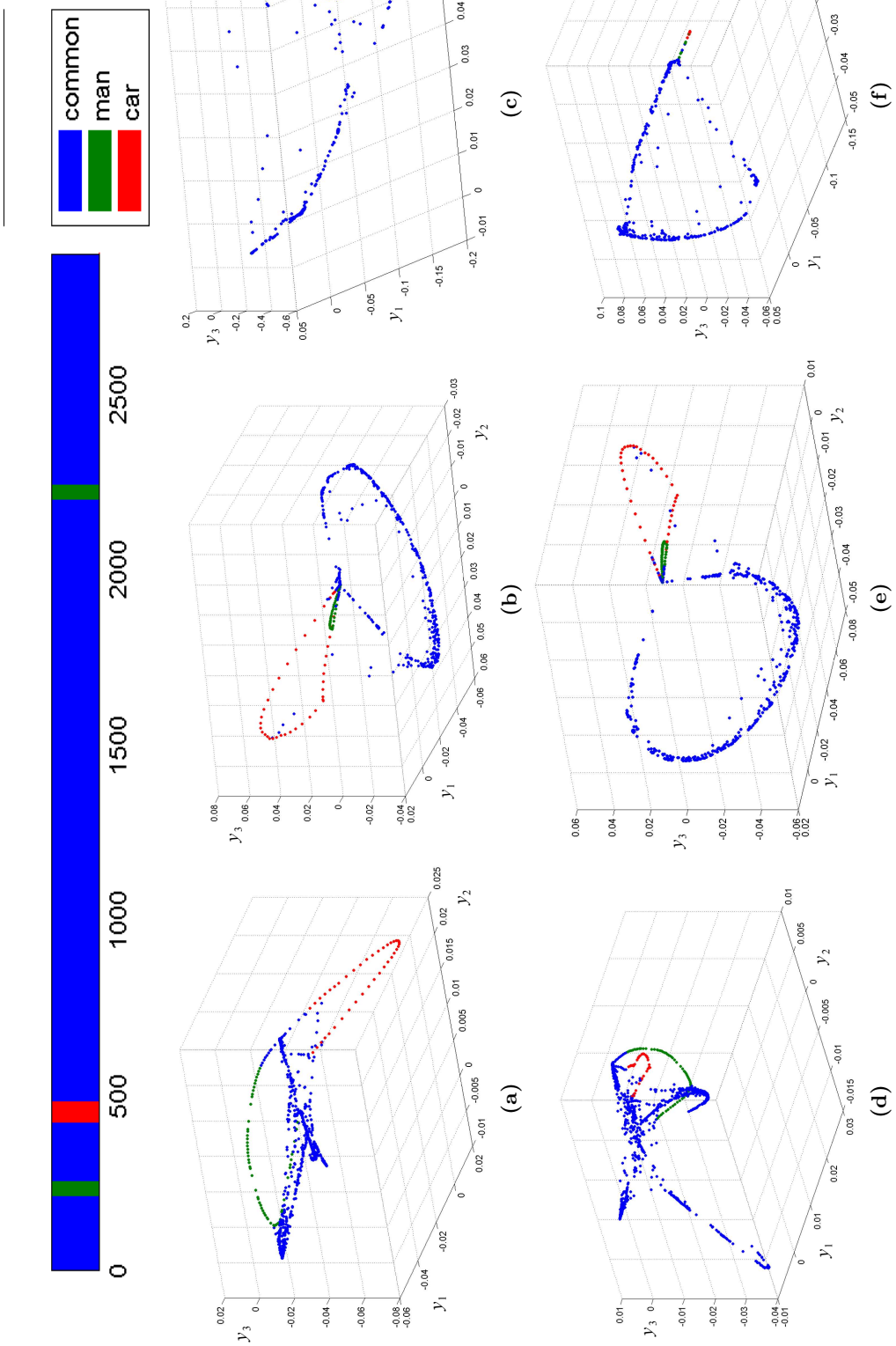
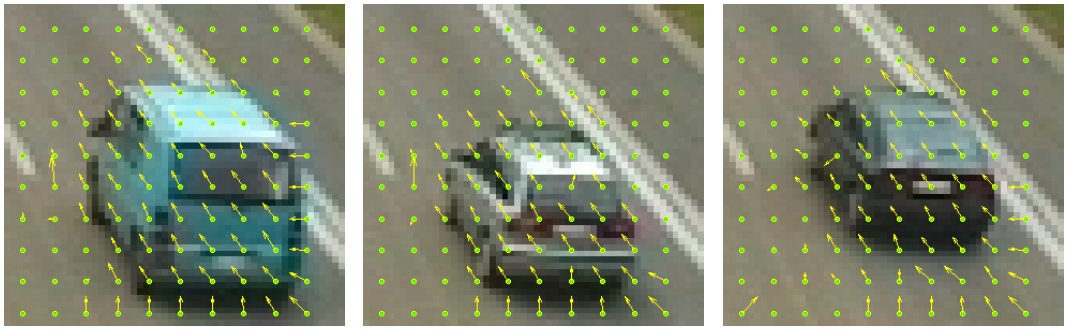


Figure 4.15: First three dimensions of the mapping produced by Laplacian Eigenmaps: (i) Pixels: (a) colour (*A1*) and (d) gray (*A2*) , (ii) Change detection mask: (b) colour (*B1*) and (e) gray (*B2*) and (iii) Motion vectors: (c) *C1* setup (f) *C2* setup. (Key: vectors in the plots are colour coded manually based on the ground truth to aid the visualisation)



(a) (b) (c)

Figure 4.16: Motion vectors represent objects that follow the same path in a similar manner, regardless of of their appearance.

by the *B1*, *B2* and *C2* (but not *C1*) setups when compared with the pixel-based setups (*A1*, *A2*). This can be attributed to the fact that the feature vectors produced by the raw pixels are more sensitive to noise and the appearance difference between cars. The change detection and motion feature vectors (*B1*, *B2* and *C2*) display a better separation between the normal and abnormal events. The *C2* projection however shows a few abnormal vectors very close to the "no activity" vectors. This is attributed to very few perceivable motion vectors in the ROI since the object is not fully visible (during entry and exit) and the motion vectors calculated at the edges of the ROI are unreliable. Finally, the projection provided based on the *C1* setup fails to show the expected structure and abnormal events are not mapped away from the class of normal patterns. The low performance is due to the maximum search radius parameter value used. The search window is small thus the fast moving objects (*e.g.* cars) are not represented properly in the observation vectors and cannot be distinguished from the abnormal events.

Each type of feature has intrinsic limitations. The *colour* or *grayscale* pixel-based features are very sensitive to appearance changes. We can see this in the projections (Fig. 4.15), where common events have a wide spread over the three-dimensional space. Even though these objects follow the same path, they do not share a similar appearance (Fig. 4.16). This sensitivity leads to performance degradation in abnormality detection. The *change detection mask* based features are less sensitive to appearance changes (common motion patterns are mapped to a compact ring in Figure 4.15). However, it may not detect objects that move slow, stop or have areas of uniform appearance. In addition to this, duplicate imprints may be generated by the motion of the object. Due to these

(a) (b) (c)

Figure 4.17: Limitations of motion vectors and change detection from sequence S1: (a,b) Uniform colour, (d) Duplicate detection of a running man (change detection only).

problems (Fig. 4.17 (b,c)), crowded scenes will not be represented well by change detection masks. Finally, *motion vector* based features suffer from similar limitations in the case of non-moving objects and objects with uniform appearance (Fig. 4.17 (a)), but there are no issues such as with duplicate imprints and motion vectors tend to be more reliably in crowed scenes.

4.4.3 Novelty detection

The ASID (Eq. 4.5) value is calculated to provide a measure of abnormal event separation for each mapping. Figures 4.18(a,b) show the resulting scores using an increasing number of dimensions (3, 6 and 9) from each feature space. To provide more detail on the motion vectors, results are also calculated using feature vectors constructed from the magnitude or the angle of the computed motion vectors (*i.e.* setups *C1-mag*, *C2-mag*, *C1-ang* and *C2-ang*).

Since the ASID is based on the Mahalanobis distance (Eq. 4.1), we are restricted by the Gaussian assumption regarding the distribution of the normal vectors in the projected space. Figure 4.15 shows the projection of motion features in three dimensions where the main motion pattern is projected on a loop. As presented in Figure 4.18(b) the first three dimensions can provide a small *separation* since the ASID value is between $0 - 10$. Nevertheless the loop of normal patterns, while not Gaussian, is projected onto a plane almost perpendicular to the plane formed by the abnormal vector loops. Thus a Gaussian distribution is a good approximation. Furthermore, in higher dimensions the trend to pack

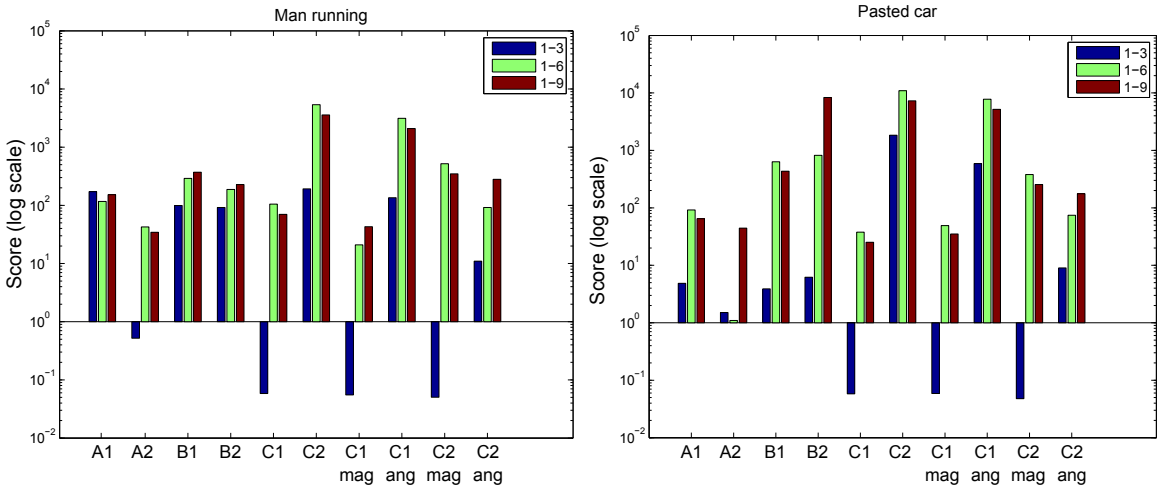
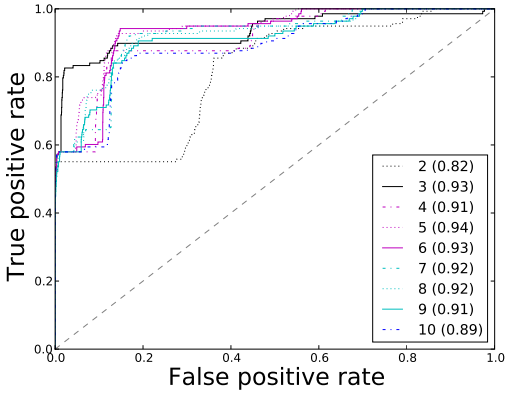


Figure 4.18: Separation scores of abnormal events in sequence S3 based on projection to 3, 6 and 9 dimensions: (a) man running event and (b) pasted car event. (Key: pixels using colour (*A1*) and grayscale (*A2*); change detection mask using colour (*B*1) and gray (*B2*); motion vectors using *C1* setup and *C2* setup; using the magnitude of motion vectors C1-mag and C2-mag; and using the angle of motion vectors C1-ang and C2-ang)

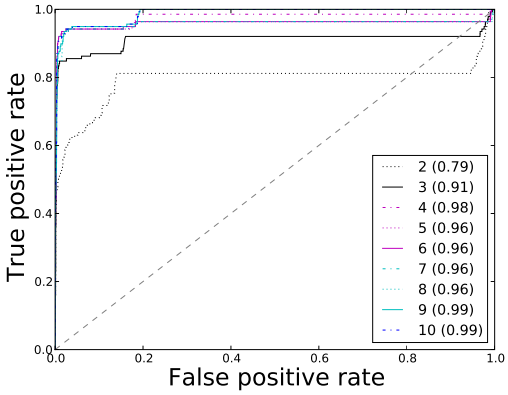
the normal vectors into a small area is magnified providing ASID values close to 10^4.

The ASID values confirm our visual inspection conclusions where change detection and motion vectors gave better visual *separation*. Selecting the proper value for the search window parameters (*i.e.* s_h, s_w) greatly improves the performance of the magnitude (*i.e.* *C2-mag* setup), which along with good estimation of the angle (*i.e.* *C2-ang* setup) accounts for the higher overall performance of the *C2* setup.

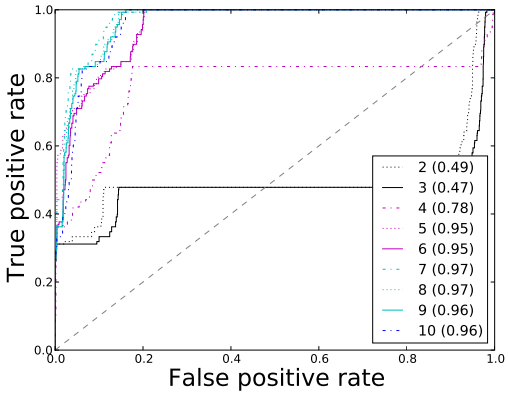
To investigate the use of the learnt subspaces for abnormality detection, we utilise detector $\mathsf{G}(\check{\mathbf{z}}_t)$ in Eq. (4.1). The training set $\mathcal{X}$ contains a single normal instance of a car passing by (*i.e.* frames 1211-1238). To produce the ROC curves we use $\mathsf{G}(\check{\mathbf{z}}_t)$ and vary the value of the threshold θ. The results are presented in Figure 4.19 where the *B2* and *C2* setups provide better classification performance than those using setup *A2*. *B2* is marginally better since it has a higher AUC value, but the cost of higher feature space dimensionality (ratio 14:1 compared to the motion vectors) favours the motion vectors in terms of computational complexity and memory requirements.



(a)



(b)



(c)

Figure 4.19: ROC curves of the abnormality detection and classification varying the number of dimensions, trained using one normal action instance of a car (1% of sequence length). Setups: (a) *A1*, (b) *B2* and (c) *C2*.

4.4.4 Crowded scenes

The performance difference in single object-scenarios between motion vectors and change detection masks is very small. However, in crowded scenarios where multiple objects move in the ROI, the change detection masks will not be able to properly describe the patterns in the scene. To verify this assumption we compare motion vectors and change detection mask on a local abnormality detector of a region from a real-world scene (*i.e.* S5A).

We assume a single region R equal to the S5A region (Fig. **4.5**a). The sequence is converted to grayscale and preprocessed using a two-dimensional low-pass blurring spatial filter with a square window of 7 pixels. We calculate motion vectors ($m_h = 5$, $m_w = 5$) using a 16×16 block and a search area of 16×16 pixels. The change detection mask for the same region (112×112 pixels), is produced using a threshold of $th_c = 6$. To further reduce complexity we remove the vectors with no activity in the ROI, leaving only one frame before and after each video clip. The process reduces the size of the feature vectors set to 5090 ($\mathcal{S}_C$) and to 3764 ($\mathcal{S}_M$) vectors for change detection mask and motion, respectively.

Given that the first 7500 frames of the sequence are normal, the training for the motion vectors $\mathcal{X}_M$ consists of the first 584 vectors and for the change detection mask $\mathcal{X}_C$ of the first 784 vectors. Testing is performed on the unlabelled section of the sequence ($\mathcal{Z}_C = \mathcal{S}_C \setminus \mathcal{X}_C$ and $\mathcal{Z}_M = \mathcal{S}_M \setminus \mathcal{X}_M$). The novelty classifier $\mathsf{G}(\check{\mathbf{z}}_t)$ is trained and tested on the low-dimensional subspace provided by the off-line graph-based dimensionality reduction method. The graph is based on the minimum-k-NN rules and is weighted using τ_o(see section 3.3.3).

Figure (4.20) compares the classification performance when using motion vectors against change detection masks. The classification is evaluated based on the ROC and PR curves. The number of subspace dimensions for each feature was selected based on the highest AUC value. In these graphs (Fig. 4.20), the motion vectors show a lower false alarm and high true positive rates, while the change detection mask is no better than a random guess of the abnormality label. The PR curves are more informative and show that motion vectors are offering a better classification performance (almost 4 times better precision) while the change detection mask cannot offer more than 0.5 recall. Since the methodology used is exactly the same for both features, we can conclude that the performance difference depends only on the feature vectors. Small sub-image regions (*e.g.* S5A) have a higher

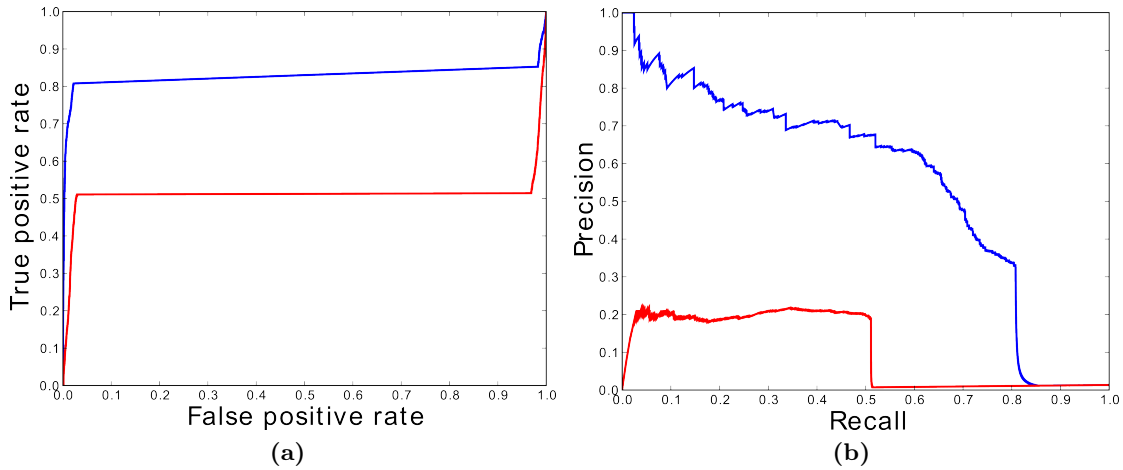


Figure 4.20: Classification performance comparison for S5A: (a) ROC curves, (b) Precision and Recall. (Key: (blue) motion vectors and (red) change detection mask).

likelihood to become crowded. Furthermore, in real-world scenes, such as depicted in S5, shadows, illumination changes and slow apparent motion (due to perspective distortion) are very common. Change detection mask seams to be more sensitive to these factors and thus has a low performance.

In this section we have compared low-level features in-order to identify a suitable candidate for use in subspace based abnormality detection. Of the three features tested (*i.e.* appearance, change detection mask and motion vectors) motion vectors were found to provide better overall performance, considering computational complexity and detection accuracy, in both single-object and crowded datasets. The next subsection will evaluate the classification performance of single off-line detectors utilising a low-dimensional space against commonly used novelty detectors utilising the high-dimensional observations.

4.5 Classifier comparison

While the results presented thus far for the subspace-aware novelty classification show that it can achieve good classification performance, it is important to establish how alternative methods perform on the same problem. This section will perform a comparison of novelty detection methods on a region of interest of the real-world sequence (S5A) against a subspace-aware low-complexity novelty classifier, as defined in Eq. 4.6. The alternative methods (SVM and k-NN classifier) will be applied on the original high-dimensional ob-

servations based on motion vectors. The detection performance for each classifier will be evaluated based on precision and recall values for the S5A region.

4.5.1 Preliminaries

Novelty detection can be achieved by using classifiers such as a one-class SVM [38] or k nearest neighbours classifier (k-NN classifier). The first method in our comparison (SVM) maps the training class of positive (normal) instances into a higher dimensional space using a predefined kernel function. In this space the algorithm learns the optimal boundary lines that enclose the single normal class. Based on these boundaries the test vectors are labelled as normal or abnormal. The SVM classifier is implement by the SVMLIB [78]. Results are presented both using a Radial-based-Function (RBF) and a linear kernel. The second method (k-NN classifier) is an adaptation of the multi-class nearest neighbour classifier to one-class novelty detection problems. The k nearest neighbours of the test vector is discovered among the training samples. The average Euclidean distance of the test vector from it's neighbours is then thresholded to label for abnormality. Finally, the subspace-aware novelty classifier is based on $\mathsf{G}(\breve{\mathbf{z}}_t)$ where the subspace is produced by LE (see subsection 3.3.4), on the motion vectors extracted from the S5A region (Fig. (4.5)a). The graph is based on the minimum-k-NN rules and is weighted using the value τ_o(see section 3.3.3). Motion vectors $\mathcal{S}_M$ are calculated for the S5A region as discussed in subsection 4.4.4. The training vectors $\mathcal{X}_M$ consist of the first 584 vectors, as defined in subsection 4.4.4. For the one-class SVM and k-NN we use the high-dimensional motion vector observations $\mathcal{X}_M$, while the subspace-aware novelty classifier is applied on the low-dimensional subspace $\breve{\mathcal{X}}_M$. Finally, the labelling results are filtered with a 5-frame window median filter to remove non-persistent labels.

The k-NN classifier requires only two parameters to be manually defined (number of neighbours and threshold), while the more complex one-class SVM requires: the selection of the kernel function (plus the parameters associated with it); the learning parameters for the training; and the threshold constant. The classification threshold for the one-class SVM is provided by the SVMLIB implementation with default parameters applied for training. The threshold value for the k-NN classifier is estimated as follows. We apply the k-NN classifier on each annotated training vector without thresholding. The maximum

value given from the algorithm (*i.e.* k-NN classifier) is then used as the threshold for abnormality labelling. In a similar manner, we find the threshold for the classifier $\mathsf{G}(\check{\mathbf{z}}_t)$.

We calculate precision PR and recall RC values (see Eq. 4.7) of the labelling based on *per frame* and *per event* abnormality detection respectively, to quantify the detection performance. When considering single frames we evaluate the detectors ability to detect the abnormal instances at time t, in this case: *(i)* true positive instances TP_f are those that are correctly labelled abnormal; *(ii)* False positive instances FP_f are incorrectly labelled abnormal; and *(iii)* false negative instances FN_f are incorrectly labelled normal. However, abnormality detectors in real-world scenarios are commonly required to detect events and not frames. Thus we need to see how the detectors perform in detecting the abnormal events (*i.e.* time segments of the video) in the sequence. We define a simple set of rules for evaluating the detection of these events: *(i)* true positive detections TP_e exist when at least one observation labelled abnormal overlaps in time with an event from the ground truth; *(ii)* false positive detections FP_e are video segments that are incorrectly labelled abnormal (*i.e.* they do not overlap with the ground truth); and *(iii)* false negative detections FN_e are assigned where abnormal video segments from the ground-truth are not abnormal in the detector's output. These event detection guidelines will give high values to detectors which have detected extended video segments as abnormal. For example, a detector could return all the sequence as abnormal and have high PR_e and RC_e values. However, frame-based precision PR_f and recall RC_f values will show that such a detector is performing poorly. Thus the combination provides a complementary view of the classification performance.

4.5.2 Abnormality labelling

The results (Fig. 4.21) using the one class SVM are inferior to the proposed approach based on the subspace-aware classifier. This can be attributed to high dimensionality [79] and non-linear correlation that exists between input vectors. In addition, one-class SVMs have been found to be very sensitive to the parameter selection in the work of Manevitz and Yousef [38].

The nearest neighbour classifier shares the characteristics of the nearest neighbour graph and is able to provide a boundary for the abnormal events without assumptions on the class distribution. It is also better at handling high-dimensional feature spaces.

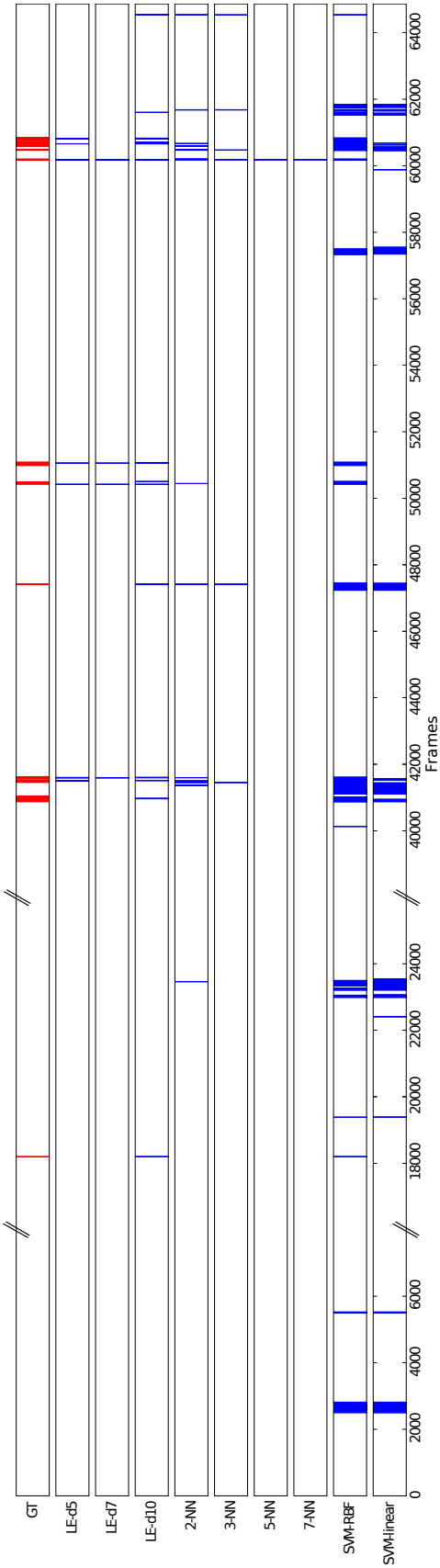


Figure 4.21: Frame labelling comparison of one-class novelty classification in sequence S5A. The ground-truth (GT) is in red and the detections provided from the novelty detectors are given in blue. (Key: *(i)* proposed approach using Laplacian Eigenmaps and projecting 5, 7 and 10 dimensions (LE-d5, LE-d7 and LE-d10), *(ii)* novelty nearest neighbour classifier (k-NN) using 2, 3, 5 and 7 neighbours, *(iii)* one-class SVM using the RBF and linear kernels and *(iv)* GT - Ground truth)

Method	Frames					Events				
	TP_f	FP_f	FN_f	PR_f	RC_f	TP_e	FP_e	FN_e	PR_e	RC_e
LE-d5	46	3	823	0.939	0.053	7	0	4	1	0.636
LE-d7	25	3	844	0.893	0.029	4	0	7	1	0.364
LE-d10	121	17	748	0.877	0.139	10	3	1	0.769	0.909
2-NN	65	45	804	0.591	0.075	7	5	4	0.583	0.636
3-NN	17	18	852	0.486	0.020	3	3	8	0.500	0.273
5-NN	4	0	865	1	0.005	1	0	10	1	0.091
7-NN	1	0	868	1	0.001	1	0	10	1	0.091
SVM-RBF	757	1430	112	0.346	0.871	11	54	0	0.169	1.0
SVM-linear	122	1381	747	0.081	0.140	6	54	5	0.100	0.545

Table 4.5: Precision and Recall values for the abnormality detection methods on the selected ROI (S5A). (Key: *(i)* proposed approach using 5, 7 and 10 dimensions (LE-d5, LE-d7 and LE-d10), *(ii)* novelty nearest neighbor classifier (k-NN) using 2, 3, 5 and 7 neighbors and *(iii)* one-class SVM using the RBF and linear kernels)

However, the labelling is very sensitive to the selection of the neighbourhood size. Using a small number of neighbours (Fig. 4.21), the classifier provides a labelling that misclassified a few of the abnormal events. Yet, increasing the neighbourhood size tends to reduce the number of abnormal events that are detected. In contrast, the proposed off-line dimensionality reduction using the minimum k-NN graph discovers a mapping which compresses the normal (common events) and maps abnormal instances away from the normal class. Such a mapping allows the subspace-aware novelty detector to label correctly the abnormal events.

Table 4.5 presents the *per frame* and *per event* evaluation based on precision and recall values. Overall the proposed subspace-aware novelty detection, based on LE and the minimum k-NN graph, performs better compared to alternative novelty detectors. Precision is high ($\bar{PR}_f \sim 0.903, \bar{PR}_e \sim 0.923$) and decreases slowly as the subspace dimensionality increases. When we increase the subspace dimensionality, we introduce more noise in the projections which results in a bigger number of false positives. Therefore the event precision decreases more quickly. Recall is low since not all the frames are detected. However, the detector correctly detects most of the abnormal events using 10 dimensions ($RC_e = 0.909$). The k-NN novelty classifier is not able to detect all the abnormal events and the precision and recall values decrease when increasing k. The SVM classifier labels a lot of instances as abnormal and achieves high RC_e value using the RBF kernel. Precision shows that this labelling is not reliable and there are numerous false positives.

This section has compared the proposed off-line subspace learning method described in

3.3.4, combined with a low-complexity novelty detector, to alternative non-subspace-aware novelty detection algorithms. The results suggest that the subspace-aware abnormality detector is able to detect all the abnormal events that take place in the associated region with higher precision than the alternative methods. The number of false alarms is also kept low, something that can be associated with the filtering capabilities of the Graph Laplacian and our selected value for the graph and weighting parameters (*i.e.* minimum-k-NN graph and the τ_o value). The next section will investigate on the use of the SR-LPP method to create a subspace-aware abnormality detector suitable for on-line use.

4.6 On-line detection

The abnormality detector needs to function on-line for real-world scenarios. Therefore the LE based approach is not suitable as it does not allow out of sample extension. We can instead use the on-line approach that utilises the LPP algorithm. LPP can be viewed as a linear approximation of the LE. The output of LPP is a transformation matrix which allows mapping new samples onto the discovered subspace. Replacing LE with LPP is logical and requires a minimal amount of implementation changes in the current motion abnormality detector framework. However, the performance is expected to be lower for the following reasons:

- On-line methodologies are usually hindered by smaller training sets compared to the complete dataset that batch mode methods have at their disposal.
- The mapping is an *approximation* of the equivalent non-linear projection provided by LE.

While the first reason is based on the fact that graph-based subspace learning benefits and is particularly successful when data are in abundance (*i.e.* number of samples goes to infinity) the second has not been directly investigated in the literature.

However, there exist a number of implementations of LPP with subtle differences in the way they discover the transformation matrix. In this experiment we will investigate the application of two LPP implementations: *(i)* the original LPP [72]; and *(ii)* the more recent implementation SR-LPP [73]. We apply the methods in batch mode (off-line) and

compare with LE, in order evaluate the eligibility of using LPP as a replacement for the LE.

4.6.1 Preliminaries

We use the sequence S3 (2900 frames, 15 fps) which is created by concatenating events from a small ROI (54×52 pixels) in a highway. The sequence has a variety of moving vehicles (*i.e.* cars, vans and trucks) numbering 20 occurrences in total. The off-line subspace learning framework using LE is compared with two versions of the LPP method:

LPP based on an implementation using the original implementation

SR-LPP based on recent work using spectral regression to implement the LPP method

Code for the LPP implementations was provided by the respective authors [72, 73]. The graph used in all methods is constructed from the motion vectors calculated from the sequence using the *C2* setup defined in Section 4.4. Each grid of vectors is reshaped into a long feature vector (50 dimensions) to form a set of high-dimensional vectors $\mathcal{S}$. These features are used to find the minimum k-NN graph. The graph edges are rescaled to the range of $[0, 1]$ and then weighted using the τ_o parameter defined by Otsu's method on the edge histogram of the graph (see subsection 3.3.3). The component values in the high-dimensional vectors where also rescaled to lie in the range $[0, 1]$. This is performed by dividing the components in the vectors by the maximum value found in $\mathcal{S}$. These changes in scale are necessary in order to utilise the higher representation accuracy that floating point numbers have between 0 and 1 when compared to the representation error that is present in larger values.

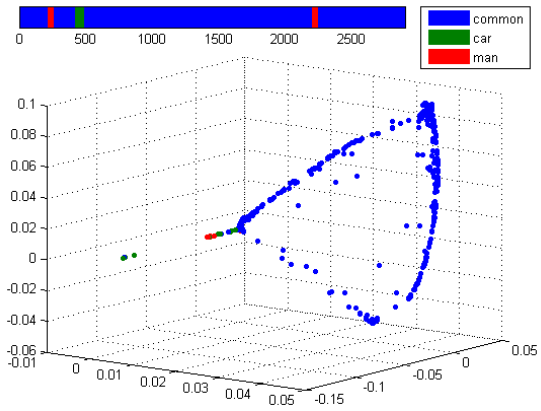
The dimensionality reduction methods are evaluated visually by inspecting the three dimensional projections of the complete sequence (*i.e.* S3). The vectors in the plots are colour coded manually based on the ground truth to aid the visualisation. Furthermore, the use of LE, LPP and SR-LPP for subspace learning in a subspace-aware abnormality detector is evaluated based on the low-complexity novelty detector $\mathsf{G}(\check{\mathbf{z}}_t)$ defined Eq. 4.6. The training set $\mathcal{X}$ contains a single normal instance of a car passing by (*i.e.* frames 1211-1238).

4.6.2 Comparison of LE, LPP and SR-LPP

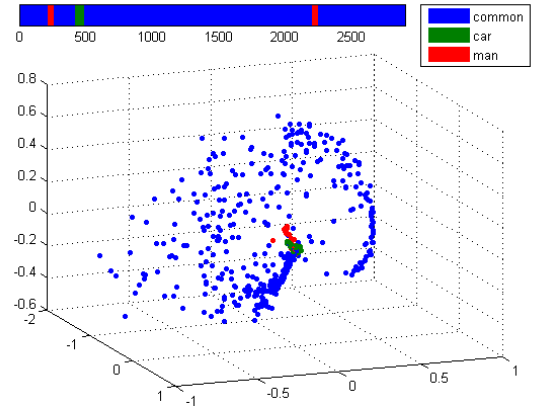
The projections provided by the subspace learning methods (Fig. 4.22) show that the only method that provided similar results to LE is the SR-LPP version of the algorithm. While in theory both implementations of LPP should provide similar results, the implementations differ in complexity and numerical stability. The respective authors [72, 73], to overcome these issues apply a number of computational "tricks" and transformations so that the running complexity of the algorithm is low and the algorithm is numerically stable. However, in order to achieve good numerical stability under difficult problems (*i.e.* small number of samples, close to singular matrices) the original LPP implementation is using PCA to preprocesses the original feature space and remove dimensions with low variance. However, as seen in section 4.3.2, information for abnormal events is not always relevant to the maximum vector variance, thus PCA on the original feature space is equivalent to filtering the data and removing non-linear dynamics from the feature vectors. Such filtering is enough to distort the results when using the original LPP algorithm (Fig. 4.22(b)). In contrast, the SR-LPP method is closer to the LE mapping (Fig. 4.22(c)) as it uses a different implementation based on spectral regression to find the solution, which has better numerical stability and does not filter the information like PCA.

The ROC curves (Fig. 4.23) show that the SR-LPP performs very similarly to the LE method. For low number of subspace dimensions SR-LPP is outperforming LE and is more consistent on it's performance (*i.e.* ROC curves). Nevertheless, the overall performance based on the AUC values of LE is higher than that of the out-of-sample methods. The performance of the original LPP method is significantly lower, which confirms our conclusions from the visual inspection of the three-dimensional projections.

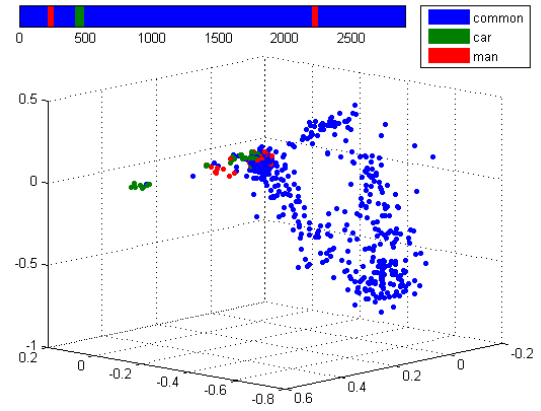
This section has investigated the use of different implementations of the LPP algorithms in order to extend the off-line subspace learning for on-line scenarios. It is apparent that the best candidate for the out-of-sample extension of the system is the SR-LPP implementation. This is based on both the visual inspection of the results and evaluation metrics that have been used to evaluate the system's performance. The next section will investigate the use of multiple abnormality detectors based on the graph-based dimensionality reduction method (SR-LPP) and evaluate the performance of on-line detection on a real-world scenario.



(a)

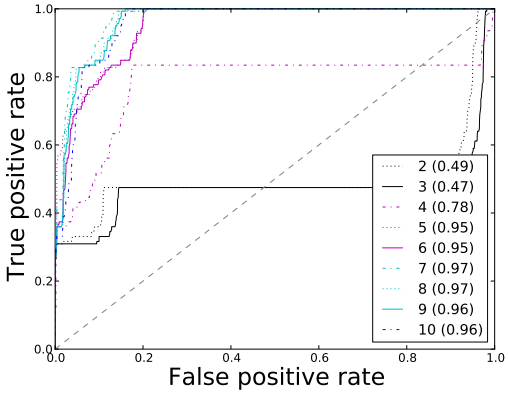


(b)

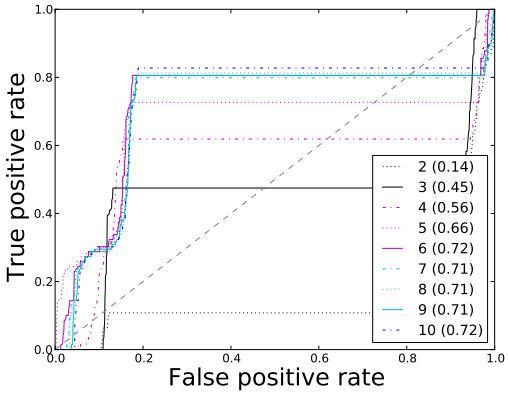


(c)

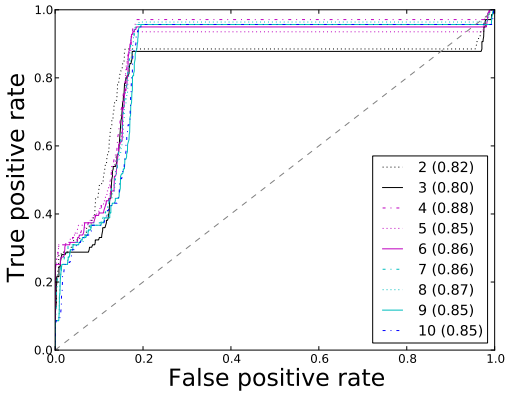
Figure 4.22: Three dimensional mappings of sequence S3 provided by: (a) LE, (b) LPP, (c) SR-LPP. (Key: vectors in the plots are colour coded manually based on the ground truth to aid the visualisation)



(a)



(b)



(c)

Figure 4.23: ROC curves of the novelty detection in sequence S3 utilising different *dimensionality reduction* methods and the varying dimensionality of the subspace: (a) LE, (b) LPP and (d) SR-LPP.

(a) (b)

Figure 4.24: Motion and detector grid defined in the greater region S5B of the real world sequence S5

4.7 Scene abnormality detection

To provide abnormality detection for a larger area of the scene we deploy a number of local detectors in a non-overlapping grid and fuse their results in order to infer the overall abnormality. Each local abnormality detector will utilise a "personal" subspace discovered by the proposed subspace learning approach (using SR-LPP) and the results will be fused based on the temporal filtering that functions (3.18) and (3.19) offer. We compare the framework based on subspace-aware detectors against a framework based on novelty detectors applied *directly* on original high-dimensional feature vectors. Additionally, we investigate and evaluate the results against the use of an alternative linear subspace learning (*i.e.* PCA) and examine the correlation between the detection performance and the size of the subspace learning sample.

4.7.1 Preliminaries

We evaluate the framework on the real-world sequence S5 regarding the first 7500 frames of the sequence are normal. In the labelled part of the video, people exit the platform through the turnstiles, move up the stairs and walk left (or right) along the corridor at the top of the stairs. In addition to the people exiting the platform, there are a few instances where someone is moving across the corridor from left to right. Based on these observations we define as abnormal actions those that do not follow these patterns. Thus actions, such as moving across the corridor from right to left or entering the platform, are considered

abnormal.

Motion vectors are calculated on the main action area S5B (Fig.4.24a) with $b_w = 19$,$b_h = 19$ and maximum displacement of 20 pixels (*i.e.* $s_h = s_w = 20$). Each image frame is preprocessed using a two-dimensional blurring filter (window = 7×7). The detector grid consists of 3×3 non-overlapping regions where each region contains a set of 5×5 motion vectors (Fig. 4.24b). To remove noise from the measurements we apply two temporal filters: *(i)* a median filter (window = 7); and *(ii)* a moving average filter (window = 3). Each filter is applied over time independently for each dimensional component of the feature vectors. The motion vectors enclosed in each region R_{ij} (with $i \in \{1,2,3\}$ and $j \in \{1,2,3\}$) provide the associated observation feature vectors $\mathbf{o}_t^{i,j}$. To avoid problems with the matrix eigensolvers (especially for PCA) we remove duplicate vectors from the samples used for the subspace learning and the classifier training (denoted with $\mathcal{S}_{ij}^*, \mathcal{X}_{ij}^*$ respectively).

We apply the framework on the complete sequence to evaluate the behaviour of the detector grid. While in previous examples, the use of a novelty classifier based on the Mahalanobis distance was enough, in this case we use the GMM novelty classifier as described in subsection 3.4.1. After subspace learning, each detector is trained using the default parameters for the EM framework on the low-dimensional projections of the annotated *normal* samples.

The frame abnormality is inferred by the local detectors with a setup ($\alpha = 1, \beta = 7, \gamma = 10$) which provides temporal filtering raising an alarm if the local detectors are signalling the last 7-10 frames as abnormal ($\sim 1/3$ seconds duration). The selected threshold θ_{ij} for each detector is the maximum rank given by each classifier on the training set. We calculate the precision and recall for abnormal events against the ground-truth and compare the results from direct application on the original (high dimensional) feature space and in subspaces learned by performing PCA and SR-LPP. The subspace dimensionality is fixed at 2 dimensions and the experiment is repeated using, at each time, an increasing size for the subspace learning set. Table 4.6 provides the relative size of the subspace learning and classifier training sets for each setup.

Setup	1	2	3	4	5	6	7
$\mathcal{X}$	7.5k						
$\mathcal{S}$	7.5k	10k	20k	30k	40k	50k	60k
$\mathcal{S}^*$ (avg.)	251	255	390	516	539	848	970

Table 4.6: Number of samples used for each setup (Key: $\mathcal{X}$, $\mathcal{S}$ refer to the sample size of the subspace learning and training sets given to the complete abnormality detection framework, $\mathcal{S}^*$ reported is the average subspace learning sample size of the feature vectors given to the 9 detector regions as calculated after the removal of duplicate instances, *i.e.* $\mathcal{S}^*_{i,j}$ with $i \in \{1,2,3\}$ and $j \in \{1,2,3\}$).

4.7.2 Comparison

We calculate the precision and recall values for abnormal event detection using each setup (Table 4.6). The confusion matrix is constructed considering a detected abnormal video segment as a true positive if at least one labelled frame is within two frames distance from the ground-truth. Multiple detections within the expected abnormal video segments are ignored and false positives are the video segments that cannot be associated with a real abnormal event. Abnormal video segments from the ground-truth that do not have any detection are considered false negatives.

Figure 4.26 provides the precision and recall scores relative to the subspace size. Results indicate that the direct application of novelty detection in the high-dimensional subspace (denoted as "direct" framework) has a small number of false positives but it fails to detect all the abnormal events. The application of PCA causes the performance of the detectors to improve and is able to detect the majority of the events, but has more false alarms. When we increase the sample size for subspace learning results with PCA do not improve. This is expected since larger sample sets can include a number of abnormal events and the PCA subspace learning process is sensitive to these "outliers". However, the subspace provided by SR-LPP does not suffer from such shortcomings. Even though the performance initially low, it steadily increases with the increase in the number of subspace learning samples, even when the majority of abnormal events are included in the learning. The equal performance ratio for subspace-aware detectors using the PCA and SR-LPP is 3.86, thus it requires almost 4 times more subspace training vectors to achieve the same performance using the SR-LPP algorithm.

However, as seen in Figure 4.25, the best performing setups for PCA (subspace set 1) and SR-LPP (subspace set 7) have an increased number of false detections. These false

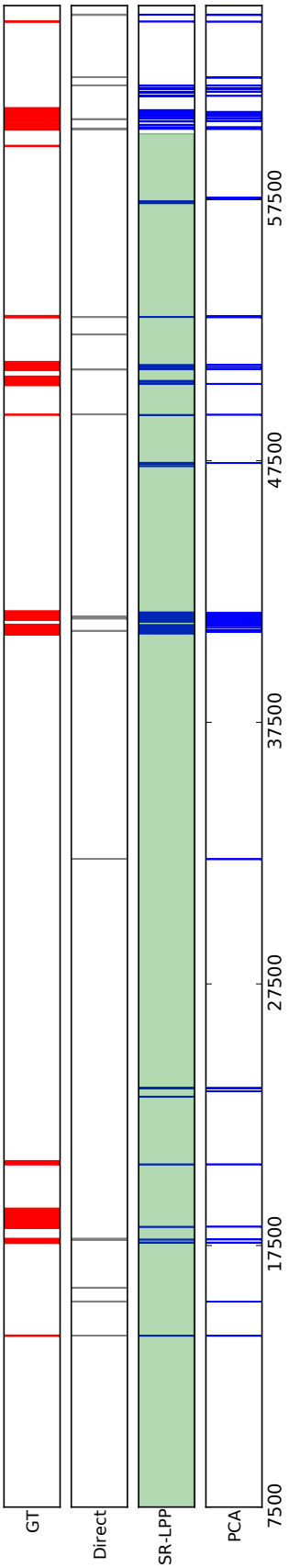


Figure 4.25: Labelling comparison of the ground truth (GT) with the multi-detector frameworks using: *(i)* the high-dimensional feature vectors; and *(ii)* subspace-aware frameworks based on SR-LPP and PCA. (Key: coloured segments (blue or gray) are abnormal and the extent of the video sample used for subspace learning in SR-LPP is denoted with green).

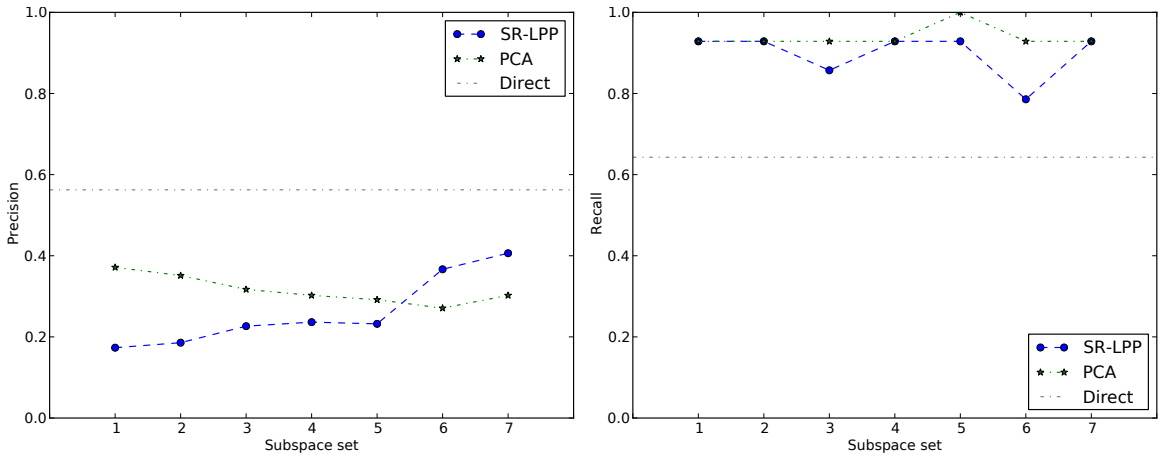


Figure 4.26: Precision (a) and recall (b) values for abnormality detector frameworks using: direct application on the high-dimensional feature vectors, PCA and SR-LPP. (x-axis: subspace learning set)

alarms are the main reason why the highest *F factor* (Harmonic mean of precision and recall) for the subspace learning ($F_{PCA} = 0.53$, $F_{LPP} = 0.57$) is marginally lower than the "direct" method ($F_{Direct} = 0.60$).

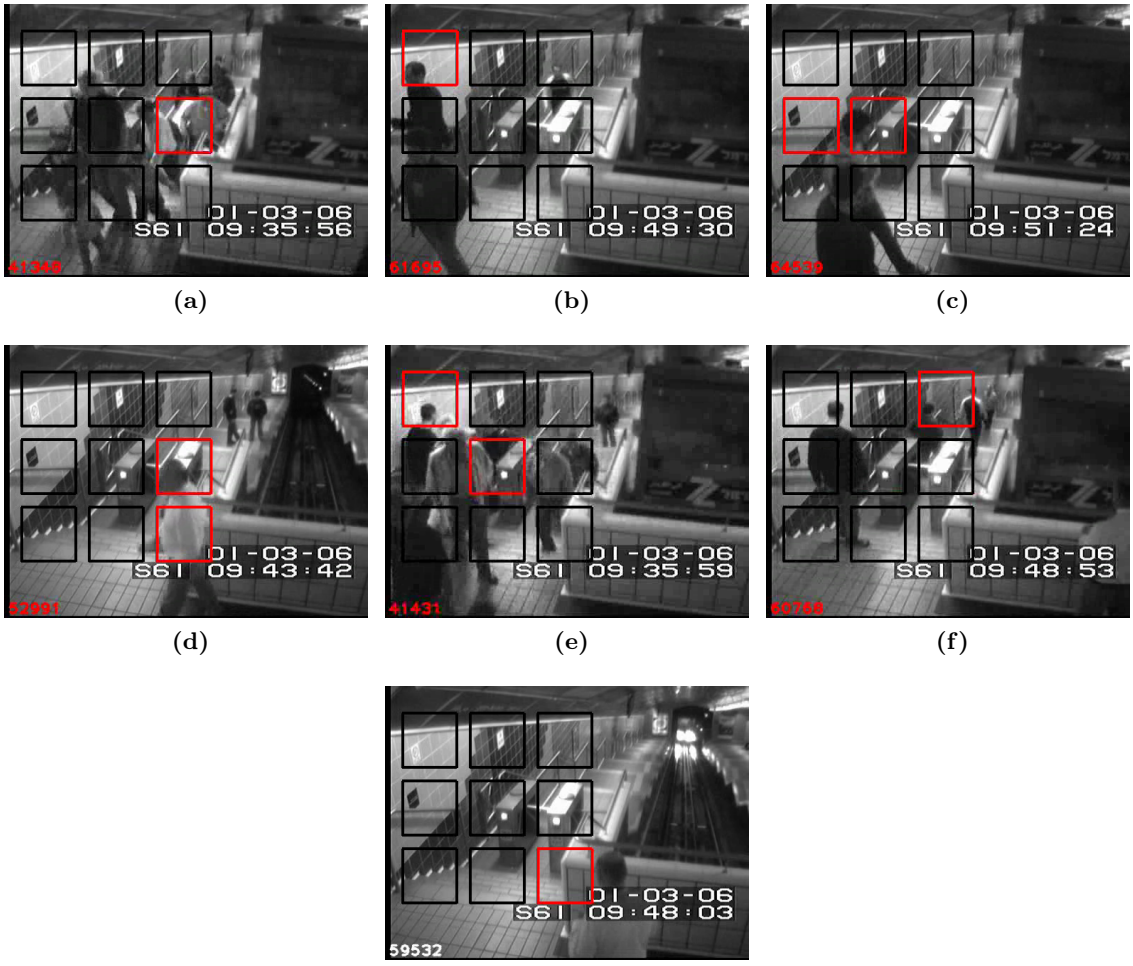
Figures 4.27 and 4.28 show sample frames from the best performing PCA and SR-LPP setups respectively. The detections that are provided by the fusion module relative to the labelling results of the local detectors can be described as *false*, *correct*, *problematic* and *missed*:

False: defined when the local detectors are falsely activated and cause the fusion module to declare an abnormal event. (*e.g.* Figures 4.27(a,b,c) and 4.28(a,b,c)).

Correct: true positive detections where the abnormal actions are clearly detected by the correct local detectors (*e.g.* Figures 4.27(d) and 4.28(d)).

Problematic: appears when there is discrepancy between the correct labelling from the fusion module and the local detectors. It is thus possible to have correct detection of events where some detectors are active, while the abnormal action takes place in another region. (*e.g.* Figures 4.27(e) and 4.28(e)). Alternatively, problematic correlation is when some of the abnormal actions in the scene are detected (*e.g.* Figures 4.27(f) and 4.28(f)).

Missed: refers to abnormal actions that were not detected by any local detector (*e.g.*4.28(g))

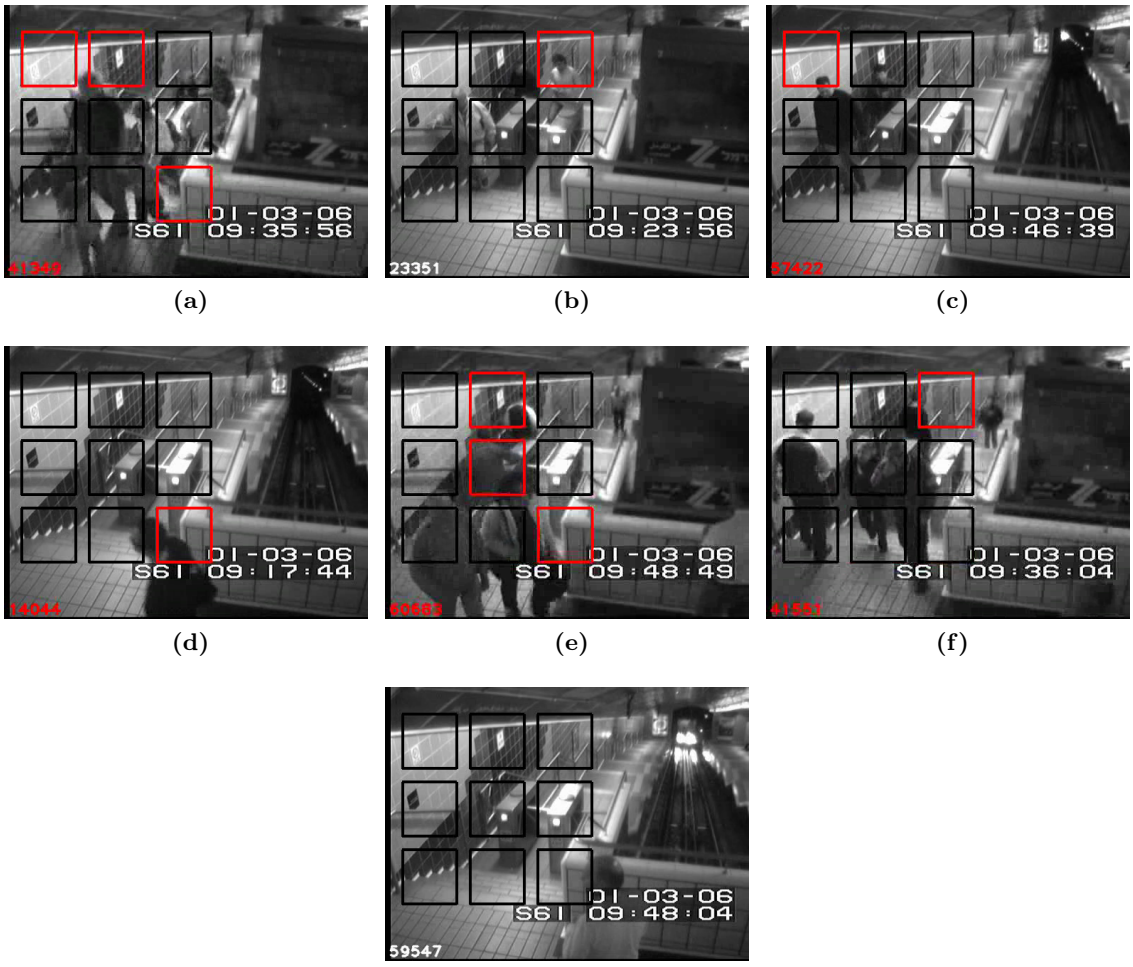


(a) (b) (c)

(d) (e) (f)

(g)

Figure 4.27: Sample images of detections and their correlation with the local detector layout using the best PCA setup: (a,b,c) false positive, (d) true positive, (e) detector abnormal state is false, (f) detector is missed a locally abnormal instance and (g) framework missed the abnormal event. (Key: abnormal state for the detectors is denoted by a red rectangle while abnormal state for the abnormality framework is denoted by the red frame number on the bottom left of the image frame)



(a) **(b)** **(c)**

(d) **(e)** **(f)**

(g)

Figure 4.28: Sample images of detections and their correlation with the local detector layout using the best SR-LPP setup: (a,b,c) false positive, (d) true positive, (e) detector abnormal state is false, (f) detector is missing abnormal instance, and (g) framework missed the abnormal event. (Key: abnormal state for the detectors is denoted by a red rectangle while abnormal state for the abnormality framework is denoted by the red frame number on the bottom left of the image frame)

or the fusion module has suppressed the alarm state of a detector due to temporal filtering (*e.g.*4.27(g)).

The errors suggest that a limiting factor in the performance of the subspace-aware detectors can be associated with the small number of available samples. In these cases, the provided vectors are not enough to describe the patterns in the local region and thus the subspace is not properly defined. Furthermore, the novelty classifier does not have enough samples to learn the normal patterns and thus it not well trained and has higher sensitivity.

4.8 Summary

This chapter described experiments that were performed in order to evaluate and test the use of the proposed graph-based dimensionality reduction method to improve the abnormality detection in low-level features extracted from video sequences. The experiments were demonstrated on single-object sequence representing a ROI of a highway and on regions extracted from a challenging multi-object real-world sequence of an underground train station.

The visual comparison of projections, on single-object dataset, demonstrate the suitability of the subspace learning based on the Laplacian Eigenmaps algorithm compared to commonly used linear (*e.g.* PCA and MDS) and rival non-linear (*e.g.* Isomap and MVU) dimensionality reduction methods. Results show, that Laplacian Eigenmaps is able to find a low-dimensional representation of the scene and uncommon (abnormal) actions, without assumptions on variance (*e.g.* PCA), providing better performance and lower complexity over alternative dimensionality reduction methods. In addition, the proposed graph can provide clues for the violation of assumptions regarding the normal patterns in the subspace training sample and a number of techniques to overcome this limitation are suggested.

The off-line subspace learning method is combined with a low-complexity novelty classifier based on the Mahalanobis distance to form an off-line local abnormality detector. Low-level features (appearance, change detection and motion) are then compared to identify features suitable for single and crowded regions. Results suggest that the use of motion vectors greatly improves the "separation" of the normal and abnormal events in the low-dimensional projection while keeping the computational complexity of the method low.

In contrast, appearance is not able to provide the same performance on the single-object sequences and change detection mask is not suitable for crowded scenes.

The off-line detector is later compared against non-subspace aware (i.e. methods using the original feature space) techniques using popular novelty detection algorithms (SVM and k-NN classifiers) on the local region extracted from a real-world sequence. The comparison shows that alternative methods applied in the high dimensional feature space are sensitive to the parameter selection and fail to overcome the *curse of dimensionality.* However, the novelty classifier trained on the low-dimensional subspace produced by the proposed off-line subspace learning approach is able to detect the abnormal events with higher accuracy and fewer false alarms. To extent the local detector for on-line scenarios the LE graph embedding method is replaced with LPP. Two different algorithmic approaches LPP (original) and SR-LPP are compared and SR-LPP is shown to provide a better approximation of the subspace discovered by the off-line algorithm.

The complete framework using a grid of local abnormality detectors is applied on a real-world scene from an underground station. The experiment compares three variations of the on-line multiple local motion abnormality detector schema based on direct (i.e. high-dimensional), classical (*i.e.* PCA) and graph-based (*i.e.* SR-LPP) subspace learning. The local detectors are based on a GMM classifier trained on the associated local region and their labelling is fused to produce the abnormality state for the scene. Results suggest that PCA is better suited for cases when the subspace learning is limited to the labelled samples. When more labelled and unlabelled data are available, the SR-LPP approach gradually improves the generalisation properties of the detectors and is robust to outliers. In order to outperform the PCA approach, the SR-LPP approach requires an excess of 3.9 times more samples. The analysis also revealed that both subspace methods are able to detect the majority of abnormal events compared to the direct use of the high-dimensional feature space with which only half of the events are detected. However, the local novelty detectors have more false alarms. This is attributed to the number of samples, as when their number is low, subspace-aware novelty classifiers suffer from over-fitting and cannot generalise well on new instances.

Chapter 5

Conclusions

5.1 Summary of achievements

This thesis addressed the problem of anomaly detection in crowded video sequences using low-level features with limited labelling information on normal events and without prior knowledge of abnormal events. The goal is to utilise both labelled and unlabelled samples, taking advantage of the knowledge that abnormal events are rare and normal events are in abundance, and thus to find a low-dimensional subspace suitable for abnormality detection. This is achieved without assumptions of linearity of the feature space and the characteristics that separate the normal and abnormal pattern classes (*e.g.* that the direction of maximum variance is important). At the same time, the method aims to keep the operational and deployment complexity low so that the framework is suitable for real-world scenarios where parameter setting through cross-validation is either not available (no abnormal events are provided) or costly. The main assumptions are that large number of unlabelled samples is available and that the majority of the motion patterns are normal. Further assumptions are that the video sequence does not contain scene cuts and that the labelled training instances do not contain abnormal events.

The proposed method utilises a spatial grid of local motion detectors based on a novel subspace representation of the feature space produced by graph-based linear dimensionality reduction methods. Namely, the unsupervised minimum-k-nearest neighbour graph is constructed and weighted to represent and approximate the local neighbourhood structure of the feature vectors in the high-dimensional space. Based on the Graph Laplacian a

low-dimensional representation is produced that emphasises the local grouping of patterns and the frequency of their appearance. The output is a subspace that closely follows the internal structure of the data that are associated with each local detector. The mapped instances of a manually labelled *normal* sample are used to train a subspace-aware novelty detector based on Gaussian mixtures and to provide the local abnormality labelling. The final decision is based on fusing the local detections by imposing temporal constraints on the labelling provided by the local detectors.

Experiments using single-object sequences demonstrate that the unsupervised subspace learning framework using Laplacian Eigenmaps outperforms popular alternative subspace learning approaches, such as PCA, MDS, Isomap and MVU. It has also been demonstrated in multi-object sequences that a low-complexity local detector based on the produced low-dimensional representation outperforms the performance of popular non-subspace-aware novelty detection algorithms (*i.e.* SVM and nearest neighbour novelty classifier). The off-line method is extended to on-line abnormality detection using SR-LPP which provides a close approximation of the embedded non-linear mapping based on Laplacian Eigenmaps.

The proposed on-line multi-detector framework based on the on-line graph-based subspace learning approach is able to detect the majority of abnormal events. Yet, in order to provide better performance than PCA, it is necessary to have more samples (3.9 times more). Nevertheless, the framework suffers from an increased number of false alarms which suggests that the subspace-aware detectors are more sensitive. The issue affects both frameworks that use PCA and SR-LPP and can be attributed to the small number of samples that are available for subspace learning and the novelty classifier. As a result, the produced subspace does not reflect all the normal patterns in the local region and the novelty classifier is not properly trained. Nevertheless, because the proposed graph-based subspace learning is not hindered by the existence of abnormal events in the subspace learning samples, it is possible, with limited additional cost to the operator (no need for labelling and manual removal of abnormal instances), to use more training samples and improve the generalisation properties of the local detectors.

5.2 Future work

Based on the analysis and discussion in the previous chapters a number of issues and questions arise which provide guidelines for improvement and extension of the presented research.

The subspace learning process is essential to the transformation and filtering of information. The proposed neighbourhood graph provides a good approximation of the similarity between vectors and their local neighbourhood, yet it does not specifically enforce temporal consistency. That is, the construction rules are unaware of any sequence between vectors. The temporal information that exists in the graph is a side-effect of the assumption that the vectors are in abundance and that the evolution of the motion vectors through time is smooth (*i.e.* video sequence without scene cuts). However, since we already have the knowledge of the temporal evolution of the motion vectors it is possible to augment the graph with connections of temporal nature. Unfortunately, information in the graph is represented by two properties: *(i)* connectivity; and *(ii)* weighting. As a result, the two graphs (neighbourhood and temporal) describe two different characteristics of the feature space with the same properties. The challenge we are faced with when augmenting the graph with prior knowledge is to balance the influence (connectivity and weighting) of the additional *known* information (*e.g.* temporal) relative to the *unknown* neighbourhood structure. Furthermore, in order for the subspace learning process to remain parameterless, the size of the temporal neighbourhood (similar to the k) needs to be defined in an automatic way. The definition of such criteria is an open challenge.

In addition to the temporal augmentation of the graph, further improvements can be achieved by finding the optimal subspace dimension for each local detector. Recent advancements on local intrinsic dimensionality estimation [80] suggest that it is possible provide information about the optimal subspace dimensionality. However, the methods are complex and their sensitivity to the initial parameters proves to be a very challenging problem.

Apart from the subspace learning process, experiments demonstrate that the multi-detector framework suffers under specific conditions of training. Since each detector is spatially anchored, it is possible that its placement is not optimal to describe the local motion patterns. Graph-based subspace learning requires the training samples (label-

led/unlabelled) to be extensive. Thus the method is expected to fail when there is minimal or no information (*i.e.* non-zero motion vectors) about the local region. Regions that are good to deploy the abnormality detector on could be verified using cross validation. However, such techniques are not applicable when knowledge of abnormal events is unavailable and thus the challenge is to be able to detect when a region is not suitable given the provided training samples. A possible direction could be to analyse the *visual* structure of the graph and identify the cases where there is not enough information to describe the variability of the patterns. Furthermore, it is possible that the classifier itself might suffer from lack of representative training (labelled) samples, yet this issue cannot be easily identified without cross-validation. It is argued that in these cases a semi-supervised training approach for the novelty classifier can potentially improve the generalisation properties of the detectors.

Finally, improvements can be achieved by revisiting the fusion method in order to incorporate spatial-temporal restrictions and to track the abnormality through the scene before the event is declared abnormal, to reduce false alarms. In addition ti this, the fusion algorithm should also consider the confidence of the detector. These changes should aim at low complexity and reduce number of free parameters needed to achieve good performance.

Bibliography

[1] K. Scott-Brown and P. Cronin, "Detect the unexpected: a science for surveillance," *Policing: An International Journal of Police Strategies & Management*, vol. 31, no. 3, pp. 395–414, 2008.

[2] N. Johnson and D. Hogg, "Learning the distribution of object trajectories for event recognition," *Image and Vision Computing*, vol. 14, no. 8, pp. 609–615, August 1996.

[3] A. Mecocci, M. Pannozzo, and A. Fumarola, "Automatic detection of anomalous behavioural events for advanced real-time video surveillance," in *Proc. of the IEEE International Symposium on Computational Intelligence for Measurement Systems and Applications*, Lugano, Switzerland, 29–31 July 2003, pp. 187–192.

[4] C. Piciarelli and G. Foresti, "On-line trajectory clustering for anomalous events detection," *Pattern Recognition Letters*, vol. 27, no. 15, pp. 1835–1842, November 2006.

[5] T. Izo and W. E. L. Grimson, "Unsupervised Modeling of Object Tracks for Fast Anomaly Detection," in *Proc. of the IEEE International Conference on Image Processing*, vol. 4, San Antonio, TX, USA, 12–15 October 2007, pp. 529–532.

[6] F. Porikli and T. Haga, "Event Detection by Eigenvector Decomposition Using Object and Frame Features," in *Proc of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, Washington, DC, USA, 27 June – 2 July 2004, pp. 114–114.

[7] M. T. Chan, A. Hoogs, J. Schmiederer, and M. Petersen, "Detecting rare events in video using semantic primitives with HMM," in *Proc. of the IEEE International Conference on Pattern Recognition*, vol. 4, Cambridge, UK, 23–26 August 2004, pp. 150–154.

[8] F. Jiang, Y. Wu, and A. K. Katsaggelos, "Abnormal Event Detection from Surveillance Video by Dynamic Hierarchical Clustering," in *Proc. IEEE of the International Conference on Image Processing*, vol. 5, San Antonio, TX, USA, 16–19 September 2007, pp. V – 145–V – 148.

[9] 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*, vol. 28, no. 9, pp. 1450–1464, September 2006.

[10] J. Salas, H. Jimenez-Hernandez, J.-J. Gonzalez-Barbosa, J. B. Hurtado-Ramos, and S. Canchola, "A Double Layer Background Model to Detect Unusual Events," in *Advanced Concepts for Intelligent Vision Systems*, vol. 4678, 2007, pp. 406–416.

[11] A. Basharat, A. Gritai, and M. Shah, "Learning object motion patterns for anomaly detection and improved object detection," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, USA, 24–26 June 2008, pp. 1–8.

[12] R. R. Sillito and R. B. Fisher, "Semi-supervised Learning for Anomalous Trajectory Detection," in *British Machine Vision Conference*, Leeds. UK, 1–4 September 2008.

[13] 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*, vol. 30, no. 3, pp. 555–560, 2008.

[14] J. Kim and K. Grauman, "Observe locally, infer globally: A space-time MRF for detecting abnormal activities with incremental updates," in *Computer Vision and Pattern Recognition*, 2009, pp. 2921 –2928.

[15] E. Andrade, O. Blunsden, and R. Fisher, "Performance Analysis of Event Detection Models in Crowded Scenes," in *Proc. of the IEE International Conference on Visual Information Engineering*, Guildford, UK, 7–9 July 2006, pp. 427–432.

[16] M. Verleysen and D. François, "The curse of dimensionality in data mining and time series prediction," in *Proc. of the Computational Intelligence and Bioinspired Systems - International Work-Conference on Artificial Neural Networks*, Barcelona, Spain, 8–10 June, Springer-Verlag 2005, pp. 758–770.

[17] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.

[18] T. Xiang and S. Gong, "Incremental and adaptive abnormal behaviour detection," *Computer Vision and Image Understanding*, vol. 111, no. 1, pp. 59 – 73, 2008.

[19] T. Xiang and S. Gong, "Video Behavior Profiling for Anomaly Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 893–908, 2008.

[20] T. Xiang and S. Gong, "Activity based surveillance video content modelling," *Pattern Recognition*, vol. 41, no. 7, pp. 2309 – 2326, 2008.

[21] D. Zhang, D. Gatica-Perez, S. Bengio, and I. McCowan, "Semi-Supervised Adapted HMMs for Unusual Event Detection," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, San Diego, CA, USA, 20–26 June 2005, pp. 611–618.

[22] P. Cui, L.-F. Sun, Z.-Q. Liu, and S.-Q. Yang, "A Sequential Monte Carlo Approach to Anomaly Detection in Tracking Visual Events," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, 18–23 June 2007, pp. 1–8.

[23] K. Sudo, T. Osawa, H. Tanaka, H. Koike, and K. Arakawa, "Online anomal movement detection based on unsupervised incremental learning," in *Proc of the IEEE International Conference Pattern Recognition*, Tampa, FL, USA, 8–11 December 2008, pp. 1 –4.

[24] J. Li, S. Gong, and T. Xiang, "Global Behaviour Inference using Probabilistic Latent Semantic Analysis," in *British Machine Vision Conference*, Leeds, UK, 1–4 September 2008.

[25] J. Li, S. Gong, and T. Xiang, "Scene Segmentation for Behaviour Correlation," in *Proc. of the European Conference on Computer Vision*, Marseille, France, 12–18 October 2008, pp. 383–395.

[26] D. Russell and S. Gong, "Exploiting Periodicity in Recurrent Scenes," in *British Machine Vision Conference*, Leeds, UK, 1–4 September 2008.

[27] D. Russell and S. Gong, "Multi-layered Decomposition of Recurrent Scenes," in *Proc. of the European Conference on Computer Vision*, Marseille, France, 12-18 October 2008, pp. 574–587.

[28] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, Washington, DC, USA, 27 June – 2 July 2004, pp. 819–826.

[29] R. Hamid, S. Maddi, A. Bobick, and I. Essa, "Unsupervised analysis of activity sequences using event-motifs," in *Proc. of the ACM international workshop on Video Surveillance & Sensor Networks*, Santa Barbara, CA, USA, 27 October 2006, pp. 71–78.

[30] I. Pruteanu-Malinici and L. Carin, "Infinite Hidden Markov Models for Unusual-Event Detection in Video," *IEEE Transactions on Image Processing*, vol. 17, no. 5, pp. 811–822, 2008.

[31] L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *Computer Vision and Pattern Recognition*, 2009, pp. 1446 –1453.

[32] E. B. Ermis, V. Saligrama, P.-M. Jodoin, and J. Konrad, "Motion segmentation and abnormal behavior detection via behavior clustering," in *Proc of the IEEE International Conferene on Image Processing*, San Diego, CA, USA, 12–15 October 2008, pp. 769–772.

[33] O. Boiman and M. Irani, "Detecting Irregularities in Images and in Video," *International Journal of Computer Vision*, no. 1, pp. 17–31, August 2007.

[34] L. R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, February 1989.

[35] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the Nystrom method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 214 –225, February 2004.

[36] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, no. 4, pp. 1–45, 2006.

[37] J. Shlens, "A Tutorial on Principal Component Analysis," April 2009, version 3.01 [last visited November 2010]. [Online]. Available: http://www.snl.salk.edu/ shlens/pca.pdf

[38] L. M. Manevitz and M. Yousef, "One-class svms for document classification," *Journal of Machine Learning Research*, vol. 2, pp. 139–154, 2002.

[39] E. B. Ermis, V. Saligrama, P.-M. Jodoin, and J. Konrad, "Abnormal behavior detection and behavior matching for networked cameras," in *Proc. of the IEEE International Conference on Distributed Smart Cameras*, Palo Alto, CA, USA, 7–11 September 2008, pp. 1–10.

[40] T. Kohonen, "Emergence of invariant-feature detectors in the adaptive-subspace self-organizing map," *Biological Cybernetics*, vol. 75, no. 4, pp. 281–291, October 1996.

[41] R. Kindermann and J. L. Snell, *Markov Random Fields and Their Applications*, ser. Contemporary Mathematics, AMS, 1980.

[42] M. E. Tipping and C. M. Bishop, "Mixtures of Probabilistic Principal Component Analyzers," *Neural Computation*, vol. 11, no. 2, pp. 443–482, 1999.

[43] S. Kullback and R. Leibler, "On Information and Sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[44] H. S. Seung and D. D. Lee, "COGNITION: The Manifold Ways of Perception," *Science*, vol. 290, no. 5500, pp. 2268–2269, 2000.

[45] J. Lin, E. Keogh, and W. Truppel, "Clustering of streaming time series is meaningless," in *Proc. of the ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, San Diego, CA, USA, 13 June 2003, pp. 56–65.

[46] J. Chen, "Making subsequence time series clustering meaningful," in *Proc. of the IEEE International Conference on Data Mining*, Houston, TX, USA, 27–30 November 2005 2005.

[47] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, Jun. 2003.

[48] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A Global Geometric Frame-work for Nonlinear Dimensionality Reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, December 2000.

[49] S. T. Roweis and L. K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, December 2000.

[50] P. Comon, "Independent Component Analysis, a new concept ?" *Signal Processing*, vol. 36, no. 3, pp. 287–314, April 1994.

[51] T. F. Cox and M. A. Cox, *Multidimensional scaling*, 2nd ed., ser. Monographs on Statistics and Applied Probability, Chapman & Hall/CRC, 2001, vol. 88.

[52] L. K. Saul, K. Q. Weinberger, F. Sha, J. Ham, and D. D. Lee, *Semi-supervised Learning*, 2006, ch. Spectral methods for dimensionality reduction, MIT Press, pp. 277–289.

[53] B. A. Draper, K. Baek, M. S. Bartlett, and J. R. Beveridge, "Recognizing faces with PCA and ICA," *Computer Vision and Image Understanding*, vol. 91, no. 1-2, pp. 115–137, 2003.

[54] L. Vandenberghe and S. Boyd, "Semidefinite Programming," *SIAM Review*, vol. 38, pp. 49–95, 1996.

[55] K. Q. Weinberger and L. K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *International Journal of Computer Vision*, vol. 70, no. 1, pp. 77–90, 2006.

[56] M. Balasubramanian and E. L. Schwartz, "The Isomap Algorithm and Topological Stability," *Science*, vol. 295, no. 5552, p. 7, January 2002.

[57] L. van der Maaten and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, Nov 2008.

[58] O. Samko, A. Marshall, and P. Rosin, "Selection of the optimal parameter value for the Isomap algorithm," *Pattern Recognition Letters*, vol. 27, no. 9, pp. 968–979, 2006.

[59] N. Mekuz and J. K. Tsotsos, "Parameterless Isomap with Adaptive Neighborhood Selection," in *Symposium of the German Association for Pattern Recognition*, ser. Lecture Notes in Computer Science, vol. 4174, Berlin, Germany, Springer Berlin / Heidelberg, 12–14 September 2006, pp. 364–373.

[60] L. Yang, "Building Connected Neighborhood Graphs for Locally Linear Embedding," in *Proc. of the IEEE International Conference on Pattern Recognition*, vol. 4, Hong Kong, China, 20–24 August 2006, pp. 194–197.

[61] L. Yang, "K-edge connected neighborhood graph for geodesic distance estimation and nonlinear data projection," in *Proc. of the IEEE International Conference on Pattern Recognition*, vol. 1, Cambridge, UK, 23–26 August 2004, pp. 196–199.

[62] L. Yang, "Building k edge-disjoint spanning trees of minimum total length for isometric data embedding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1680–1683, 2005.

[63] Li Yang, "Building k-edge-connected neighborhood graph for distance-based data projection," *Pattern Recognition Letters*, vol. 26, no. 13, pp. 2015–2021, October 2005.

[64] D. Meng, Y. Leung, Z. Xu, T. Fung, and Q. Zhang, "Improving geodesic distance estimation based on locally linear assumption," *Pattern Recognition Letters*, vol. 29, no. 7, pp. 862–870, May 2008.

[65] J. Jaromczyk and G. Toussaint, "Relative neighborhood graphs and their relatives," in *Proceedings of the IEEE*, vol. 80, no. 9, September 1992, pp. 1502–1517.

[66] K. R. Gabriel and R. R. Sokal, "A New Statistical Approach to Geographic Variation Analysis," *Systematic Zoology*, vol. 18, no. 3, pp. 259–278, September 1969.

[67] G. T. Toussaint, "The relative neighbourhood graph of a finite planar set," *Pattern Recognition*, vol. 12, no. 4, pp. 261–268, 1980.

[68] S.-W. Cheng and Y.-F. Xu, "On [beta]-skeleton as a subgraph of the minimum weight triangulation," *Theoretical Computer Science*, vol. 262, no. 1-2, pp. 459–471, July 2001.

[69] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000, [last visited November 2010]. [Online]. Available: http://opencv.willowgarage.com/wiki/

[70] J. Chen, H. R. Fang, and Y. Saad., "Fast approximate k-nn graph construction for high dimensional data via recursive lanczos bisection," *Journal of Machine Learning Research*, vol. 10, pp. 1989–2012, 2009.

[71] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, January 1979.

[72] X. He and P. Niyogi, "Locality Preserving Projections," in *Proc. of the Annual Conference on Neural Information Processing Systems*, ser. Advances in Neural Information Processing Systems, vol. 16, Whistler, BC, Canada, MIT Press, 8–13 December 2004, pp. 153–160.

[73] D. Cai, X. He, and J. Han, "Spectral Regression for Efficient Regularized Subspace Learning," in *Proc. of the IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, 14–21 October 2007, pp. 1–8.

[74] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, August 2003.

[75] V. Mariano, J. Min, J.-H. Park, R. Kasturi, D. Mihalcik, D. Doermann, and T. Drayer, "Performance Evaluation of Object Detection Algorithms," in *Proc. of the IEEE International Conference on Pattern Recognition*, Quebec, Canada, 11–15 August 2002, pp. 965–969.

[76] B. Borchers, "CSDP : A C library for semidefinite programming," Department of Mathematics, New Mexico Tech, Socorro, NM, USA, Tech. Rep., 1997, [last visited November 2010]. [Online]. Available: https://projects.coin-or.org/Csdp/

[77] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861 – 874, 2006.

[78] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm. [last visited November 2010].

[79] S. Klement, A. Madany Mamlouk, and T. Martinetz, "Reliability of Cross-Validation for SVMs in High-Dimensional, Low Sample Size Scenarios," in *Proc. of the International Conference on Artificial Neural Networks*, ser. Lecture Notes in Computer Science, vol. 5163, Prague, Czech Republic, Springer, 3–6 September 2008, pp. 41–50.

[80] K. Carter, R. Raich, and A. Hero, "On Local Intrinsic Dimension Estimation and Its Applications," *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 650 –663, February 2010.