Iconic Indexing for Video Search

Graves, Andrew Phillip

ISSN 1470-5559

Iconic Indexing for Video Search

Andrew Phillip Graves

Queen Mary
University of London
Department of Computer Science

RR-06-09 June 2006

Iconic Indexing for Video Search

Andrew Phillip Graves

Submitted for the degree of Doctor of Philosophy

Queen Mary, University of London

June 2006

Abstract

The ability to search video is an important and challenging problem. This is especially so in the surveillance domain, in which many thousands of cameras record real-world action. A video search system needs to generate and store an expressive but also compact *index* that can be used for a variety of tasks, such as retrospective investigation and real-time intervention. The index must also be *uncommitted*, because, as it is generated, the retrospective search tasks to be performed upon it are generally unknown.

In this thesis, inexpensive action-based features to used to form the index. In the first part of the thesis, frame action is extracted in the form of a cellular grid of active cells. Furthermore, the segmentation of activities is performed using an adapted spatio-temporal connected-components algorithm. These provide a novel action-based representation without the need for performing object detection and tracking. The indices are used for the *temporal segmentation* task using a sliding window method.

Whereas geography based representations provide information on the occurrence and locality of action, they do not capture local appearance structure and directionality. In the second part of the thesis, action coefficients are computed using a localised wavelet transform. Centroids, found by a clustering process, form an Iconic visual vocabulary then used to perform frame indexing. Temporal segmentation is achieved by cumulative analysis of the representation over time. Furthermore, a video *summarisation* is computed using the most discriminant active pixels in the scene.

It is beneficial in the search process to integrate manually assigned semantics into a graph for belief based browsing. This provides a semi-automatic semantic search. To this end, a traditional competing models approach, trained with wavelet coefficients, is compared against a novel rank voting algorithm for semantic browsing. Furthermore, a Bayesian fusion network is used to perform a combination of evidence.

Declaration

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

Some parts of the work have previously been published as:

- A.P. Graves and S. Gong. Spotting scene change for indexing surveillance video. In *British Machine Vision Conference*, pages 469–478, Norwich, England, September 2003.
- A.P. Graves and S. Gong. Wavelet-based holistic sequence descriptor for generating video summaries. In *British Machine Vision Conference*, pages 167–176, Kingston, England, September 2004.
- A.P. Graves and S. Gong. Surveillance video indexing with iconic patterns of activity. In *IEE International Conference on Visual Information Engineering*, pages 409–416, Glasgow, Scotland, April 2005.

Andrew Graves

London, June 2006.

Acknowledgements

I would to thank my supervisor Professor Shaogang Gong for providing guidance and support throughout this endeavour. Thank you for introducing me to the fascinating subject of computer vision. Thanks also to Professor Peter McOwan, Professor Mounia Lalmas, Tao Xiang, Marie-Luce Bourguet and Thomas Rölleke for various hints and discussions along the way.

I am grateful to the Engineering and Physical Sciences Research Council (EPSRC) for sponsorship of this work.

I would like to acknowledge the tremendous support given to me within the Department by Joan, Gill and Carla, and also by the subject librarian, Kathy. I am also thankful for the friendship and camaraderie of my fellow Vision colleagues and postgraduate students over the years, in particular Lukas, Hayley, Alessio, Dave, Alex, Jeff, Andy, Keith, Adam, Melanie, Lourdes and Fabrizio. Thanks to everyone in the Department for providing a good environment in which to study.

Finally, to my family and friends for great support. I would especially like to thank Bryn for listening to my theories, and for reminding me that life is there to be grabbed and enjoyed.

Contents

1 Introduction **8**

1.1 Video search 10

1.2 The approach 12

1.2.1 Pre-attentive video feature extraction 13

1.2.2 Generating a video sequence representation 13

1.2.3 Automatic partitioning and conceptual visualisation 13

1.2.4 Semi-automatic labelling and investigation 13

1.3 Contributions 14

1.4 Structure of the thesis 15

2 Video indexing and search: A review **18**

2.1 Data search methodology 18

2.1.1 Browsing, visualisation and summarisation 19

2.1.2 Query by example 24

2.2 The semantic gap 26

2.3 Indexing methods for visual search 27

2.3.1 Text based approach 28

2.3.2 Image feature based approach 29

2.3.3 Video feature based approach 31

2.3.4 Semantic based approach 33

2.4 Video structure discovery 35

2.4.1 Shot transition detection 36

2.4.2 Scene change detection 38

2.5 Surveillance 41

2.6 Discussion 42

3 Pre-attentive video processing **45**

3.1 A sequence and its segmentation 46

3.2 Frame based video indexing 47

3.2.1 A measure of reliable temporal change 47

3.2.2 Grid based frame descriptor 53

3.2.3 Frame spatio-temporal context 56

3.3 Activity based video indexing 59

3.3.1 Significant activity segmentation over space and time 59

3.3.2 Spatio-temporal activity profiling 61

3.4 Similarity metrics for visual search 63

3.4.1 Spatial similarity using transformation cost 66

3.4.2 Exploiting temporal context 68

3.5 Explanation based partitioning 71

3.5.1 Localised temporal coherence 71

3.5.2 Finding significant coherence minima 72

3.6 Experiments 74

3.7 Discussion 85

4 Iconic video indexing **87**

4.1 The need for discriminant visual context 88

4.2 A wavelet-based sequence descriptor 88

4.2.1 Wavelet analysis of temporal change 88

4.2.2 An iconic visual vocabulary 91

4.2.3 A cumulative analysis of iconic appearance 94

4.2.4 A video scene trajectory 94

4.3 Model order selection using entropy 95

4.4 Partitioning by trajectory approximation 97

4.5 Visualisation using discriminant action 99

4.6 Experiments 100
4.7 Discussion 114

5 Semi-semantic analysis 117

5.1 Semantic belief based browsing 118
5.2 Competing models approach 119
5.2.1 Expectation maximisation training 119
5.2.2 Haar-based models for sequence investigation 121
5.3 The rank voting method 125
5.3.1 Frame-based ranking 125
5.3.2 Rank positions as votes 126
5.4 Modality fusion 129
5.4.1 Combination of evidence 129
5.4.2 An algorithm for constructive inference 132
5.5 Experiments 135
5.6 Discussion 136

6 Conclusions and Future Work 138

6.1 Motivation 138
6.2 Conclusions 140
6.2.1 Pre-attentive processing 140
6.2.2 Iconic indexing 140
6.2.3 Semi-semantic analysis 141
6.3 Future Work 142

A Glossary 144

B Normalisation of a series 147

Bibliography 148

Chapter 1

Introduction

In recent years, the potential of multimedia applications, combined with significant advances in computer vision and information retrieval research, has led to widespread interest in visual information retrieval. Efforts are focused on the ability to efficiently, and effectively, represent and search visual data. These tasks are commonly known as *indexing and retrieval*. Although image analysis and retrieval are at a mature state of development, video remains a significant challenge. The interest in video is motivated by rapidly growing video databases, i.e. the emergence of those generated by mobile phones, home video or surveillance security cameras, and facilitated by large increases in computing power and storage capacity.

Specifically, the development of useful visual search systems has been significantly hampered by the *semantic gap*. Whereas computer representations are numerical in nature, search requirements are rooted in semantic meaning. The translation from a numerical representation into a semantic description, i.e. the bridging of the semantic gap, has attracted widespread research but remains unsolved. The core problem is that successful approaches for recognition, for example in face recognition, are computationally unstable, require significant clean training data, do not scale, and consequently are over constrained and not useful for more generic visual search. Furthermore, we find that video captured in the critical domains, such as surveillance, often do not possess the

necessary visual cues (e.g. colour) and often lack detail (low resolution).

An alternative, more pragmatic, philosophy is to facilitate a semi-automatic visual search. Rather than bridging the semantic gap in one leap, this approach attempts to minimise the effect of the gap by a combination of: a system providing a more intuitive numerical description; and a user is trained to interpret the description and navigate the visual data. Consequently, there is a growing need for techniques and tools that facilitate semi-automatic video search that assumes some level of user skill. A semi-automatic *video investigation system* can be used typically for a number of tasks:

Retrospective investigation. Many hours of skilled human computer operator time is currently used in traversing the large collections of video that are acquired during a typical crime analysis. Time and money could be saved if a computer system could analyse the databanks automatically, and intelligently present pertinent information to detectives.

Scene profiling. One of the most important roles for current surveillance systems is the generation of *usage statistics*. Such figures are generated to either identify potential hazards such as full platforms on the London Underground, or to aid in the design of new public spaces for maximum utility.

Real-time intervention. Another potential application for automated surveillance is an *alert generation system*, whereby a system monitors many surveillance streams, identifies threatening behaviour, and prompts security personnel to intervene. Such a system requires a strong recognition ability for low-quality data and must result in few false-positive situations. Closed circuit television is considered an effective tool in crime prevention (Welsh and Farrington, 2002).

Abnormality detection. Whereas crime intervention is concerned with identifying known crime behaviours, an alternative approach is to detect abnormal actions in video. The video collection itself is used as the template for normal behaviour. Abnormal actions may be highlighted to a human operator for more investigation.

Video archiving. An important, but as yet unchartered, potential application exists in

the management of home video. In recent years, as video capture devices have proliferated in the guise of small hand-held video recorders and also in mobile-phones, the volume of home video has increased enormously. It needs to be managed effectively using tools that assume low end user ability.

For the duration of this thesis, we are more interested in the retrospective investigation task for common Closed Circuit Tele-Vision (CCTV) based surveillance video. This area may be considered a hybrid of information retrieval, surveillance and computer vision, and consequently is relatively unexplored. Also, in collaboration with the DTI/EPSRC ICONS and EPSRC/MOD INSIGHT projects[1], sufficient video data was made available for modelling.

1.1 Video search

Many modern computer systems are often concerned with the capture and processing of large amounts of data, in particular that captured from video capture devices. The crucial tasks of *representation* (how the data is formatted, organised and compacted), *storage* (how the data is physically archived and fetched), and *search* (how the data is sifted for more important parts), must be solved. We consider that the tasks of representation and search are intrinsically linked as to perform effective search an appropriate representation model is needed. We focus upon:

Indexing. The process of transforming a video collection into a representation that is optimal for searching.

Retrieval. The process of performing search. It is achieved by comparing a query representation to the index and forming a ranking.

Browsing. The process of navigating a sequence to find content.

Traditionally, *information retrieval* has been largely focused on the ability of systems to perform text-based search. Retrieval models and mechanisms are well documented and are known to be successful, for example Salton's vector space approach

[1] See `http://www.dcs.qmul.ac.uk/research/vision/`

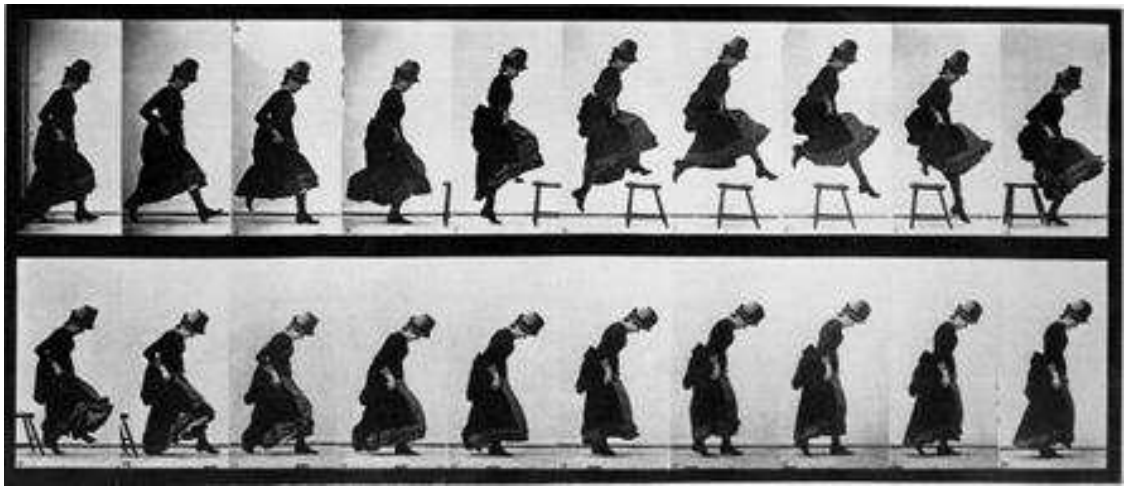


Figure 1.1: An Eadweard Muybridge collotype plate from 1887. A bank of cameras were used to record the sequential movements of a subject. The result provides an impression of motion when viewed in temporal order. Used with permission of the Victoria and Albert museum, London.

for term/query comparison (Salton, 1989). In recent years, text-based retrieval systems have been scaled to the World Wide Web (WWW) and have proved to be effective and massively popular. One could even argue that such systems have entered the public consciousness, for example the verb *to Google* is widely used to refer to the act of performing search using the Google engine[2]. The ability of textual search is greatly facilitated by the numerical representation of text in documents, such as American Standard Code for Information Interchange (ASCII), leading to efficient numerical document representations and *similarity metrics*. However, the extension of text-based systems to images and video has proved to be more troublesome. Such an approach typically exploits image filenames, anchor text (the text in the web-page link to the image), and existing associated textual descriptions to form a text-based representation[3]. Unfortunately, this is mostly ineffective because the fundamental content of images and video is interpreted by *Human cognitive visual perception* and cannot be adequately expressed in words.

Therefore, to perform image and video search an understanding and translation of the visual content is required. An image consists of a two-dimensional (2D) rectangle grid of pixels and a video consists of a number of similar images that when shown in a

[2]See `http://en.wikipedia.org/wiki/Google\_(verb)`
[3]For an example see `http://images.google.com/`

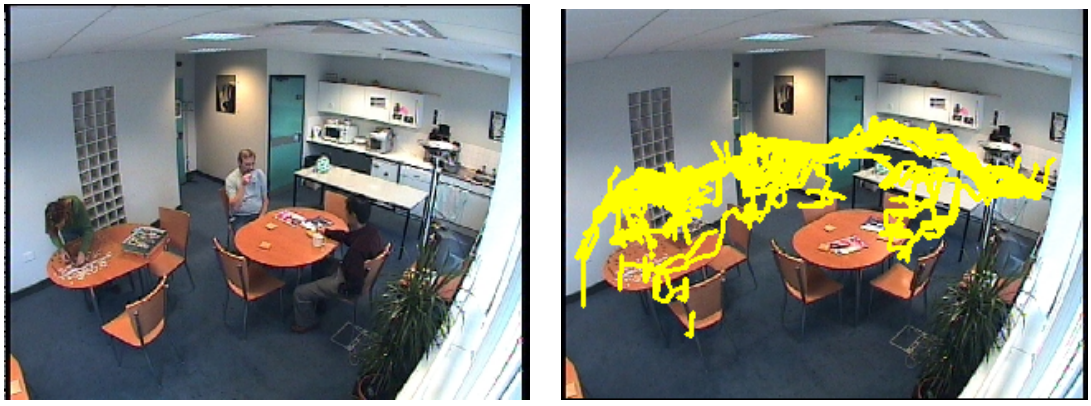


Figure 1.2: An indoor tearoom scene and extracted trajectories. The trajectories give little indication of the content that is occurring and do not provide a discriminant context.

strict *temporal order* give the impression of a moving scene. With the added temporal dimension the image sequence is considered three-dimensional (3D). See Figure 1.1 for an early sequence that illustrates this construction.

To be able to automatically interpret *dynamic scenes*, to discover and represent the content, has become one of the main goals of a computer vision system. Such an approach generally requires the detection of an object in the scene and monitoring its movement and behaviour over several frames. However, we consider that the common *object-detection-tracking* (ODT) paradigm is unsuitable for video search as it provides no useful representation that can be compared (Xiang and Gong, 2006). For example, a tracking system will find many similar *trajectories* in a scene but they do not provide a useful indication of what is happening. See Figure 1.2.

1.2 The approach

The aim of this research is to address the problem of automatically extracting information about the scene from video data, and using the information to perform video search. This involves the generation of a compact, efficient and expressive sequence representation, the automatic partitioning of the sequence into segments, image visualisation to highlight the dynamic scene content, and the provision of tools for semi-automatic labelling and investigation. More specifically, the following problems are addressed:

1.2.1 Pre-attentive video feature extraction

It is difficult to compute an inexpensive reflection of the action content in a video sequence that is impervious to noise. Noise is caused by many factors, for example coding, changing global illumination, or the visual ambiguity of the content. Prominent examples of visual ambiguity are found in motion analysis, for example (Longuet-Higgins, 1984), and in art, for example in Leonardo's Mona Lisa (Gombrich, 1995). A representation is desired that is both computationally inexpensive, robust to noise, and captures the action content effectively

1.2.2 Generating a video sequence representation

A video representation that is both compact and robust, yet also expressive and generic, is a significant challenge. It must be able to capture the pertinent information about what is occurring in the scene. It is also desirable for the approach to be expressed in a mathematically understood format so that known metrics and models can be used if required. For example, if a histogram is used then metrics such as χ^2 become available (Lew, 2001).

1.2.3 Automatic partitioning and conceptual visualisation

Automatic partitioning, also known as *temporal segmentation*, of a sequence into segments. Once discovered, segments provide a much more autonomic and generic search experience and are better suited as the fundamental content block during video search.

Conceptual visualisation of the video content. Using segmentation, a window onto the video content is needed to provide the user with a understanding of the underlying content without being distracted by image details captured by the entire sequence. We consider partitioning and visualisation to be critical in performing sequence navigation and video search.

1.2.4 Semi-automatic labelling and investigation

To perform a semi-automatic video investigation, the ability to attach labels and interactively examine the content using semantics is required. If the user annotates a number

of frames, or segment chunks, with a known semantic label, then the similarity of the chunk can be used to infer the semantics of the remainder of the sequence. Therefore, an easy mechanism is desired for integrating user assessments into a navigation tool.

1.3 Contributions

The novel contributions of this thesis are:

1. A robust temporal change model is computed for extracting more salient action content in a scene. A compact descriptor is then formed that captures the location of scene action using a grid-based approach. The sequence representation is analysed to discover significant activities, considered as spatio-temporal connected active cells, by a process of *activity segmentation*. Each activity is profiled, using spatial and temporal characteristics, and a cost-based similarity metric is used to perform search. A sliding window based approach is proposed for performing temporal segmentation.

2. To capture localised appearance of an action, rather than its location, a novel wavelet based sequence descriptor using the Haar basis function (Graps, 1995) is proposed. The transform is computed upon a local region of temporal change, meaning that the coefficients capture the directionality characteristics of the cell action. The common coefficients, considered to be the powerful visual elements in the scene, are found by a clustering process and used to form an *Iconic visual vocabulary*. The vocabulary is extremely compact and is used for scene description and search.

3. To reflect the changing content in the scene, a cumulative analysis of the occurrence of the vocabulary elements is computed for a video sequence. It is found that the cumulative histograms have different characteristics for different long-term scenes. To focus upon important change Principal Component Analysis is performed on the cumulative histogram to represent a content change subspace. The most important three dimensions are used to form and visualise a *Video scene*

trajectory of a video sequence. In effect, the trajectory remains continuous if the scene content remains continuous, permitting scene breaks to be discovered at the vertex positions of significant trajectory change.

4. A sequence visualisation approach is proposed where *Segment summary frames* are computed for each discovered segment, using a visualisation of the the most interesting action occurring. Interesting activity is found using an analysis of its occurrence distribution throughout the sequence.

5. A novel technique is proposed for performing semantic analysis of video using a small number of manual assessments. A *Rank voting* approach uses the rank positions of labelled items, produced by a content-based retrieval system, to vote for their respective semantic labels. The result facilitates semi-automatic, semantic-sensitive, video browsing and analysis. Furthermore, the combining of semantic estimates is proposed using a Bayesian fusion framework and a constructive inference algorithm.

1.4 Structure of the thesis

The remaining chapters of this thesis are arranged as follows:

Chapter 2 provides a review of related research in video indexing, video structure discovery, visualisation and the surveillance domain. The limitations and important issues to be address are discussed.

Chapter 3 describes an approach for performing a robust temporal-change based feature extraction using a grid-based descriptor. Segmentation and profiling of significant activities is also addressed. The problem of performing pre-attentive partitioning is addressed using a cost-based activity similarity metric.

Chapter 4 addresses the problem of forming a compact, uncommitted representation for capturing scene action content. The task of analysing the long-term continuity in a sequence is addressed using a cumulative analysis of the representation. Automatic partitioning and segment conceptual visualisation are performed.

Chapter 5 address the problem of performing semi-automatic labelling for facilitating an intuitive semantic browsing system.

Chapter 6 summarises and concludes the work presented in this thesis. The potential future research directions are discussed.

Appendix contains a glossary of terms and also common mathematical procedure.

Figure 1.3 illustrates how Chapters 3-5 relate to each other. Chapter 3 presents a pre-attentive approach. The video data is transformed and the scene action estimated. A cost-based similarity metric is proposed and video search and temporal segmentation demonstrated. Chapter 4 presents an iconic approach. The term *icon* is used to refer to a small patch of localised activity. Activity content is estimated using a wavelet descriptor. A trajectory is formed in a content change subspace and used to perform temporal segmentation. A method is proposed for conceptual summarisation of the discovered segments. Chapter 5 presents a semantic approach. The manual assessments are used to estimate the semantic content of a video. This is presented to the user to enable interactive sequence investigation.

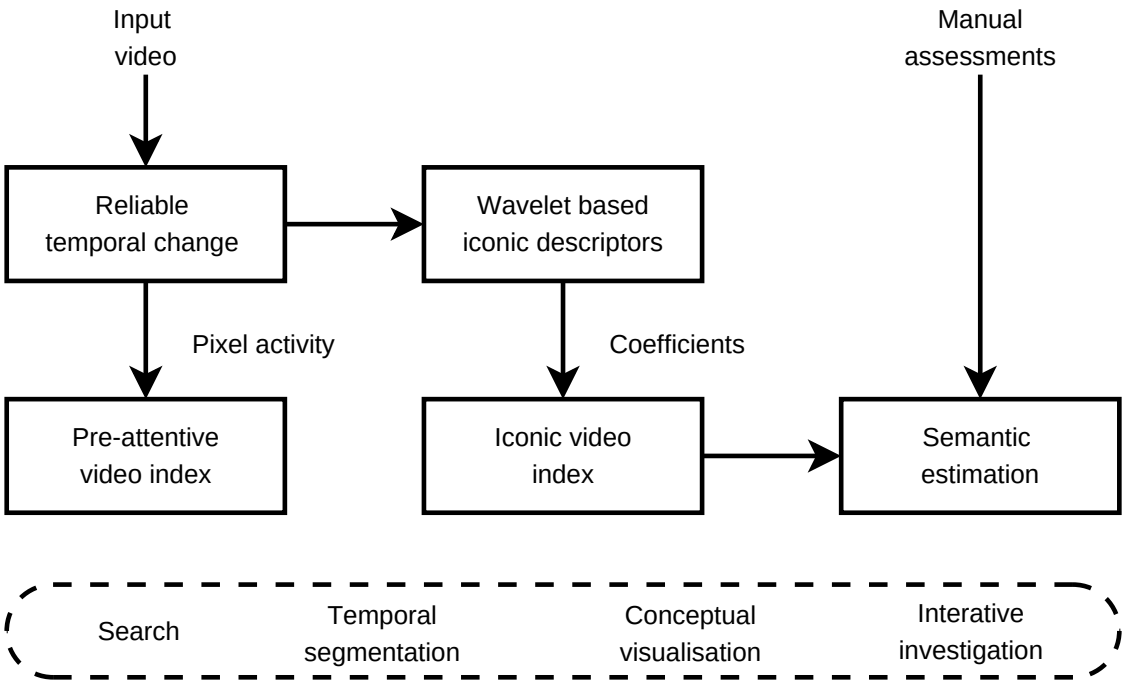


Figure 1.3: A diagram to illustrate how Chapters 3-5 relate. In Chapter 3, an input video is processed and a reliable temporal change formed. The result is a pre-attentive video index that can be used for search and temporal segmentation tasks. In Chapter 4, the reliable temporal change is used to compute an iconic description using the coefficients produced by a Haar wavelet transform. The result is an iconic index that can be used for search and temporal segmentation tasks. We also propose an approach for conceptual visualisation. In Chapter 5, manual assessments are added to the iconic index. These are used in a semi-semantic estimation method that permits interactive investigation.

Chapter 2

Video indexing and search: A review

In this Chapter known methods for performing video indexing and search are reviewed. In Section 2.1 two important paradigms are introduced: browsing and query-based retrieval. In Section 2.3 methods for performing textual, feature-based and semantic indexing are described. In Section 2.4 the problem of video structure discovery is discussed, also known as the problem of temporal segmentation, that is needed for partitioning a sequence into retrievable components. In Section 2.5 surveillance indexing and the common difficulties are described. Finally, in Section 2.6 the limitations of existing work are summarised and used to motivate the work presented in this thesis.

2.1 Data search methodology

As new digital information is generated and stored it needs to be indexed for effective and easy access and search. The essential purpose of any retrieval system is to satisfy a user's *information need* using a finite set of documents. Methods for text information retrieval have progressed rapidly in the last twenty years (Rijsbergen, 1979; Baeza-Yates and Ribeiro-Neto, 1999). Huge web-based textual search engines now exist and enjoy widespread use in society, for example, Google, Yahoo or Altavista. Beyond text, as the quantity of digitised visual information has increased exponentially, a growing need for

multimedia retrieval systems has been witnessed (Maybury, 1997; Bimbo, 1999; Lew, 2001). For example, the the British Broadcasting Corporation (BBC) stores "over one and a half million items of video and film, or about 600,000 hours of footage"[1] in its archives, that are used to construct programmes for future transmission. As programme construction using existing clips is less expensive and quicker than new clip generation, performing effective visual search is critical. Similarly, in a very short space of time, the number of Closed Circuit Tele-Vision (CCTV) cameras that record daily activity has surged, leading to slow manual search during crime investigation. A similar situation exists with home video, for both that captured from a home video recorder and mobile devices. Clearly, the need for effective visual search is paramount.

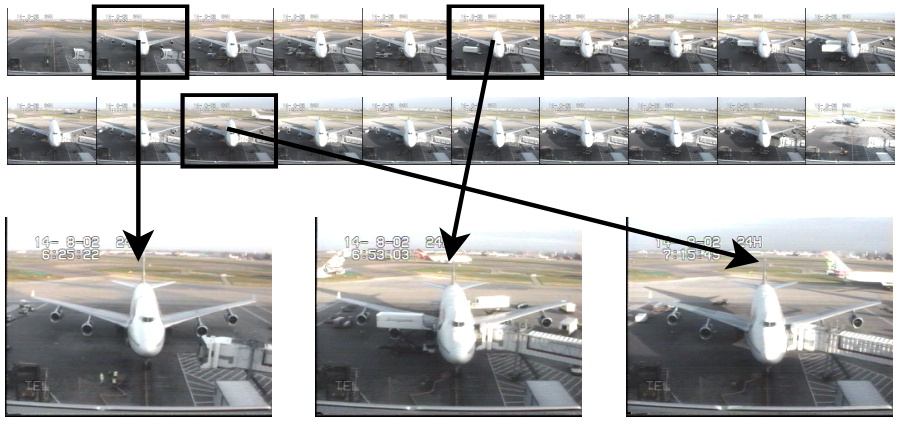
There are two main alternative paradigms for visual search, browsing and query-based retrieval, as illustrated in Figure 2.1 and are discussed in the following Sections. A useful analogy as to the relationship between browsing and retrieval was provided by Rui and Huang (2000). In essence, a prospective reader of a book has two distinct methods of evaluating the content without reading it at all. Firstly, they can look at the book's table-of-contents to get a general feel of the content and structure (browsing). Alternatively, they can use the index page to find specific sections of interest (retrieval).

2.1.1 Browsing, visualisation and summarisation

It is well understood that human cognition is very effective at quickly scanning visual data for important content. However, a problem with manually analysing image and video collections is the sheer scale of the task. Automated browsing, visualisation and summarisation systems are tasked with reducing this scale to enable a user to ascertain and evaluate the content more selectively and quickly. Three types may be distinguished:

- Browsing. Permits navigation towards a search goal.
- Visualisation. Provides a mechanism to visually explore the video documents. The set of video in a system is known as a searchspace.
- Summarisation. Reduces the task required to understand the content.

[1] See `http://www.bbcresearchcentral.com/`



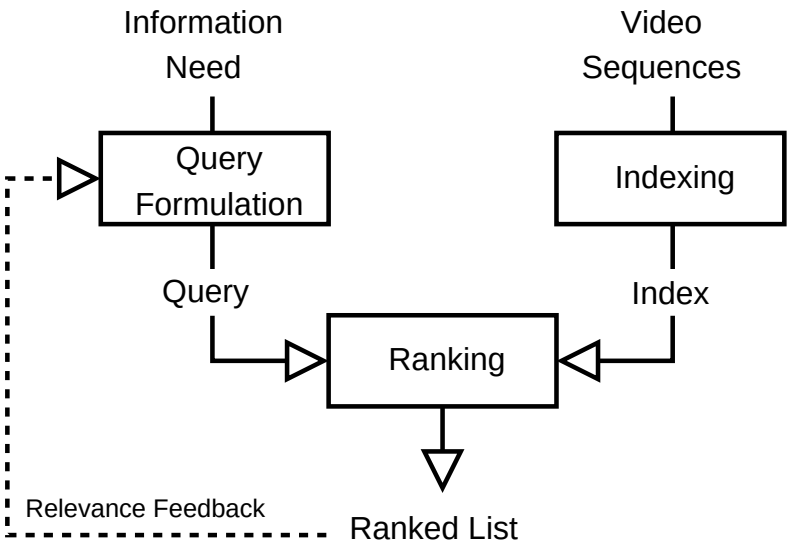


Figure 2.1: Two alternative frameworks for finding video content. (top) A browsing framework. The content is analysed and the important key-frames are shown to the user. The user can quickly scan for interesting content. (bottom) A query-based retrieval framework. The user expresses a requirement as a query, that is compared against an index using a retrieval function, to produce a ranked list.

Browsing systems permit the user to quickly drill-down in the search-space using a successive fractions *search tactic*. That is, by an iterative manual decision process the space is reduced until all that remains is relevant. It requires perceptual grouping in order to provide meaningful choices to the user. Visualisation systems are similar except that they are more focused on illustrating many items at once, using their similarities. Summarisation systems are concerned with producing abstracts of the content that can be seen more quickly. For image systems browsing and visualisation are more suitable. However, for video systems browsing and summarisation are more appropriate as a frame is only meaningful when shown directly after another frame (temporal context).

Considering the importance of image browsing and visualisation, it is remarkable that very few approaches are presented in modern literature. A simple approach is to present a grid of ordered image thumbnails to the user for review. In (Combs and Bederson, 1999) a zoom-able grid is used. However the optimum number of images or image resolution remain unclear.

A more sophisticated approach is to generate a self-organising arrangement using image similarities and clustering. In (Rodden *et al.*, 2001), a spatial arrangement is formed using a low-dimensional similarity space (see Figure 2.2). A caption-based similarity was compared with a visual-feature-based similarity by a user evaluated study. The caption-based technique was found to produce good results. However, the required manual labelling was subjective and time-consuming. On the other hand, a visual-feature-based approach was problematic due to that similar neighbouring images appeared to blend into each other, causing confusion.

A hierarchical image browsing system using visual feature similarity was proposed by (Lai and Tait, 1999). The system performs visual feature based clustering using colour. A hierarchy of similar images is presented to the user for navigation purposes. Unfortunately, this system is dependent on that the feature distribution being meaningful in some sense. In other words, that images close in feature space share similar semantics and should be presented together. However, this assumption does not always hold. In a recent study, Heesch and Rüger (2004) proposed that each image is represented as a

vertex in a directed graph and arcs are formed between two images if one is retrieved as the nearest neighbour of the other using a variety of visual features including colour histograms. Furthermore, Heesch and Rüger (2005) suggested that groups of densely connected images exhibit semantics leading to a semantic-based browsing without training. However, this representation is rather large and costly to compute.

In our study, three approaches to video browsing and summarisation are considered:

- An Intelligent Fast-Forward analogous to that provided by a video recorder.
- A static frame-based summary of content.
- Extraction of a excerpt, called a *video skim*. For a discussion of video skims, the reader is directed to Li *et al.* (2001).

Furthermore, these approaches are divided into those that exploit video structure and those that operate on unstructured video. By video structure, we mean a meaningful organisation into several layers of different granularity[2] and is widely considered beneficial as it facilitates non-linear access. Whereas unstructured video may contain sections, they do not reflect the content in any perceptive way.

In early work on unstructured video by Mills *et al.* (1992), the number of frames is iteratively sub-sampled to reduce the quantity of information presented to a user. The system is useful, but is limited because it does not consider the actual content of the video during the drill-down. In Tonomura and Abe (1990), several approaches are presented in a single workbench environment: variable speed, sampling flash, a rush browser, and space-time browser. The variable speed duplicates a Fast-Forward (FF) button; the sampling flash shows the key-frames using detected shots; the rush browser displays periodic frames irrespective of the structure; the space-time browser presents a frame sample of the structural units. However, a uniformly sampled approach does not account for the video content, i.e. low-action periods with little visual variation are over represented and high-action periods are under represented. To address these issues, Srinivasan *et al.* (1999) proposed an FF approach using non-linear frame sub-sampling based upon the

[2]Video structure is described in more detail in Section 2.4.

Figure 2.2: Screenshots from an image and video browsing system. (top) A self organising visualisation of image-space by (Rodden *et al.*, 2001). The images are shown according to their positions in low-dimensional colour space; (bottom) A video browsing tool by (Rui *et al.*, 1998). Key-frames are presented according the video structure. The user is able to navigate the structure and play the corresponding clips.

amount of motion in the frame. It is claimed that the result frame rate is between $10-15$ times faster than the original frame rate whilst retaining important content.

For structured video, a common approach is to use a key-frame for each segment, typically the first, last or mid-frame (Lew, 2001). In Rui *et al.* (1998) a table of contents is produced using the first and last frame of each shot/scene as the shot key-frames (see Figure 2.2). This system is shown to be effective and provides a user with a suitable navigation tool. In Uchihashi (1999), the key-frames are evaluated for importance using their length and novelty. A comic-book style video summary is then produced by resizing the frames according to their importance, and then using a temporally constrained packing algorithm. The system is shown to work well for highly structured domains with clean data, such as video of indoor lectures. In Arman *et al.* (1994), synthesised Rframes were generated to summarise the important visual properties of each shot. The abstractions are generated off-line before browsing begins. However, they are complex and require user ability to interpret. For example, it is difficult to correspond the motion data to the original video.

To present a visualisation of video activity, Zeng *et al.* (2002) computed a motion map for each shot using the level of temporal change. Unfortunately, the motion-maps do not capture the temporal order characteristics or the local directionality of action. In Iyengar and Lipman (2000), similar to image visualisation systems, the shots are clustered and a cluster browser presented to the user. The authors argued that it is not important whether the clusters make cognitive sense, rather the clusters provide a useful view of the video. In Ma and Zhang (2000), a semi-automatic face recognition system permits home video to be labelled and organised by the presence of known/named faces. This approach assumes that home video sequences usually contain a small number of repeating human faces. However, it requires manual labelling and the system presented only has 50 faces, indicating extensive labelling may not be straightforward and objective.

2.1.2 Query by example

For most document types including images, a common approach to performing retrieval is to adopt the query-based retrieval paradigm. Essentially, a system compares a query,

constructed to represent the user information need, against an index that contains representations of all the *retrievable documents*. For each document a query-document similarity is computed and a ranked list provided to the user with the most similar items first. The user then explores this list in order to find the desired content.

The processes in the query-based retrieval system are as follows:

Indexing The data is described off-line using a representation optimal for searching. The result is known as the *index*.

Query Formulation The user expresses information need in a numerical form consistent with the formation of the index. This expression is known as the *query*.

Ranking A comparison of the query against the index using a retrieval function. The result is a *ranked list* of items.

Relevance Feedback As retrieval is an iterative process, the user can examine the ranked list and mark documents as either *relevant* or not relevant (binary). The system can then improve the ranking by reformulating the query using an extrapolation. Alternately, the user can reformulate the query manually.

For text-based systems, the basic unit of content is *term* or word. The query formulation process consists of a user selecting a number of terms that are required. However, for image and video systems, there is no such generic basic unit of content. Hence, numerous visual features are computed to produce numerical document representations that can be compared in a feature space. Unfortunately, it is rather difficult for a user to express the information need in this numerical form due to its somewhat conceptually arbitrary nature. To overcome this problem, a new paradigm was proposed and has been widely adopted, the *Query-by-example* (QBE). A user presents an image, or a video clip, as an example of what is required of the system. The system automatically computes the features (query formulation) and performs the feature comparisons during ranking.

A perplexing problem with query-by-example is that a user is expected to provide a suitable example. Often, this is not easily possible because the user does not possess such

an example, or the user only has a vague idea of what they are looking for. The approach is also dependent on correct feature selection and a distance metric that corresponds to human cognitive perception. Therefore, it is considered that a truly effective retrieval system needs to provide both a browsing system for manually gauging the contents of the database and selecting a suitable example, and a retrieval system for performing visual queries to find content (Bimbo, 1999).

2.2 The semantic gap

The *Semantic gap* is an important issue in many computer vision systems, but particularly for indexing. It refers to the lack of coincidence between machine low-level digital representations of visual data and the human high-level cognitive understanding of the same data[3]. This is particularly important for the task of retrieval because the system is trying to find suitable visual data that matches the user search expectations. User studies suggest that image retrieval systems that operate using low-level visual features alone often do not satisfy user requirements (Enser and Sandom, 1995; Eakins, 1996; Enser and Sandom, 2003; Eakins *et al.*, 2004).

The major issues in visual indexing are characterised as:

	Sensory	**Semantic**
Gap	Features are only an approximation of the real world.	Features do not correspond to human understanding.
Ambiguity	The same thing can have many different visual appearances.	Similar visual appearances can have different meanings.

Related to the semantic gap, there is also the notion of a *Sensory gap* that refers to the fact that computer vision systems always deal with a digital approximation of a perceived world. A gap exists between the real-world and the computational descriptions that are derived during the recording process[4] (Smeulders *et al.*, 2000). This gap

[3] "The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation" (Smeulders *et al.*, 2000).

[4] "The sensory gap is the gap between the object in the real world and the information in a (computational) description derived from a recording of that scene" (Smeulders *et al.*, 2000)

is amplified by quantisation noise that a vision system must negotiate during modelling. The choice of the most suitable approximation is also considered as the feature selection problem. *Sensory ambiguity* is another problem, in that an object can have many different appearances in feature-space but still retain a single identity. Different appearances may be due to rotation, translation or scale variance, or due to different lighting conditions. The same object often exhibits different colours according to the lighting conditions, a problem known as colour constancy.

The *Semantic gap* refers to the fundamental differences between a digital and human representation of content. Whereas machines are inherently numerical, the human brain prefers concepts. Unfortunately, it is very difficult for machines to handle concepts and so a machine-human semantic gap exists. For example, in a video retrieval system the user may wish to perform a semantic search, e.g. find a video clip of "Tony Blair at the Whitehouse", whereas the system only has a colour, texture, shape and motion based representation. The problem of associating numerical representations with meanings is known as the symbol grounding or binding problem.. A *Semantic ambiguity* problem also exists, in that, even if a machine were able to converse in semantics, Human understanding can vary widely because of its subjectivity. A single concise definition of a word, or semantic, is not possible in most cases, despite recent efforts in the construction of ontologies. A successful visual search system must minimise the effects of these four issues, in particular the sensory and semantic gaps.

2.3 Indexing methods for visual search

The physical manifestation of an index is essentially a look up table of

[content identifier, content descriptor]

tuples. The content identifier contains information such as physical file location information and, for video, the segment start and end frame positions. This enables an application to search the index, but also find and present the actual content to the user. The content descriptor consists of a textual, or numerical, explicit explanation of the associated content. The descriptor is used for matching.

2.3.1 Text based approach

Owing to the success of early text-based information retrieval systems, for example (Rijsbergen, 1979; Salton, 1989), the first approach to visual indexing was reliant upon *Meta-data*. Textual descriptions of image content are generated manually, attached to the visual data, and then a standard text-based retrieval system can be employed to perform a text query. A good example of this kind is the probabilistic model (Baeza-Yates and Ribeiro-Neto, 1999). However, it quickly became apparent that manually generated meta-data was insufficient due to the lack of perceptual saliency and the subjective nature of a manual annotation (Bimbo, 1999). Also, it is somewhat unrealistic to expect an armada of manual annotators to sit, watch and analyse all the video output produced for all domains, and produce satisfactory and consistent meta-data.

Alternatively, a content-independent approach is to use the format, author's name, title, date, location, anchor text, and size as meta-data in order to perform retrieval using a deterministic matching system. To this end, the Moving Pictures Expert Group (MPEG) recently introduced the Multimedia Content Description Interface (MPEG-7) standard to hold both facts and visual features. However, the use of facts is not generally considered important to visual retrieval. Rather, approaches may exploit text that occurs within the visual data itself. In Lienhart (1996), artificial text is extracted from the video frames using Optical Character Recognition (OCR). Unfortunately, the majority of the text that can be extracted is not of use. In Smeaton (2001), the described Físchlár system uses closed caption and teletext information from broadcasters such as the British Broadcasting Corporation (BBC) and Radio Telefís Éireann (RTE).

In the Informedia project, Hauptmann and Smith (1995) deployed a Hidden Markov Model based speech recognition to extract a transcript as the basis of meta-data. Graves; Graves and Lalmas (2001; 2002) also proposed a system in which a transcript was divided amongst video segments and an Inference network used to perform ranking. All of these systems work with a degree of success. However, the text-based indexing paradigm for visual search is fundamentally flawed as a text-only representation cannot fully capture the perceptual properties of visual data (Bimbo, 1999; Colombo *et al.*, 1999).

2.3.2 Image feature based approach

Much effort has been made into developing methods for content-based image indexing and retrieval. Colour and texture features are most commonly used and are effective for image retrieval, whereas shape/edge features are effective in specialist domains (Bimbo, 1999; Lew, 2001; Bimbo, 1999; Smeulders *et al.*, 2000; Lew, 2001; Castelli and Bergman, 2002).

In the seminal work of Swain and Ballard (1991), a colour histogram representation was computed to support a histogram intersection similarity metric. Histograms are shown to offer invariance to translation, rotation, scale and partial occlusions. Later, the Query By Image Content (QBIC) system of Flickner *et al.* (1995) adopted a weighted histogram distance. In the VisualSEEK system of Smith and Chang (1997), the feature space was divided by spatially localised regions before histogram computation. In Jain and Vailaya (1995), a histogram of edge directions is added in order to perform combined colour and shape trademark retrieval. It was shown that a more robust result was achieved than either of the individual feature-based approaches. A major problem with colour-based systems is that changes in global illumination can drastically affect the feature space, caused by the colour constancy problem.

Unfortunately, image retrieval based upon holistic histogram matching is vulnerable to quantisation problems during the binning process. Fixed sized histograms do not achieve a good balance between the representation expressiveness and efficiency, i.e. the representation size is constant regardless of the content's perceptual complexity or importance. To overcome this problem, Rubner and Tomasi (1999) proposed a variable sized signature representation. The signature consists of a set of representative feature clusters found through vector quantisation. Additionally, a similarity metric called Earth Mover's Distance is used to compare signatures based upon the transportation cost (Rubner *et al.*, 2000). However, the estimation of the flow matrix is nontrivial.

An object-based image retrieval system, called Blobworld, was proposed by Carson *et al.* (1999). Firstly, a spatial image segmentation is achieved by clustering a combined colour and texture feature space using the Expectation Maximisation algorithm (Demp-

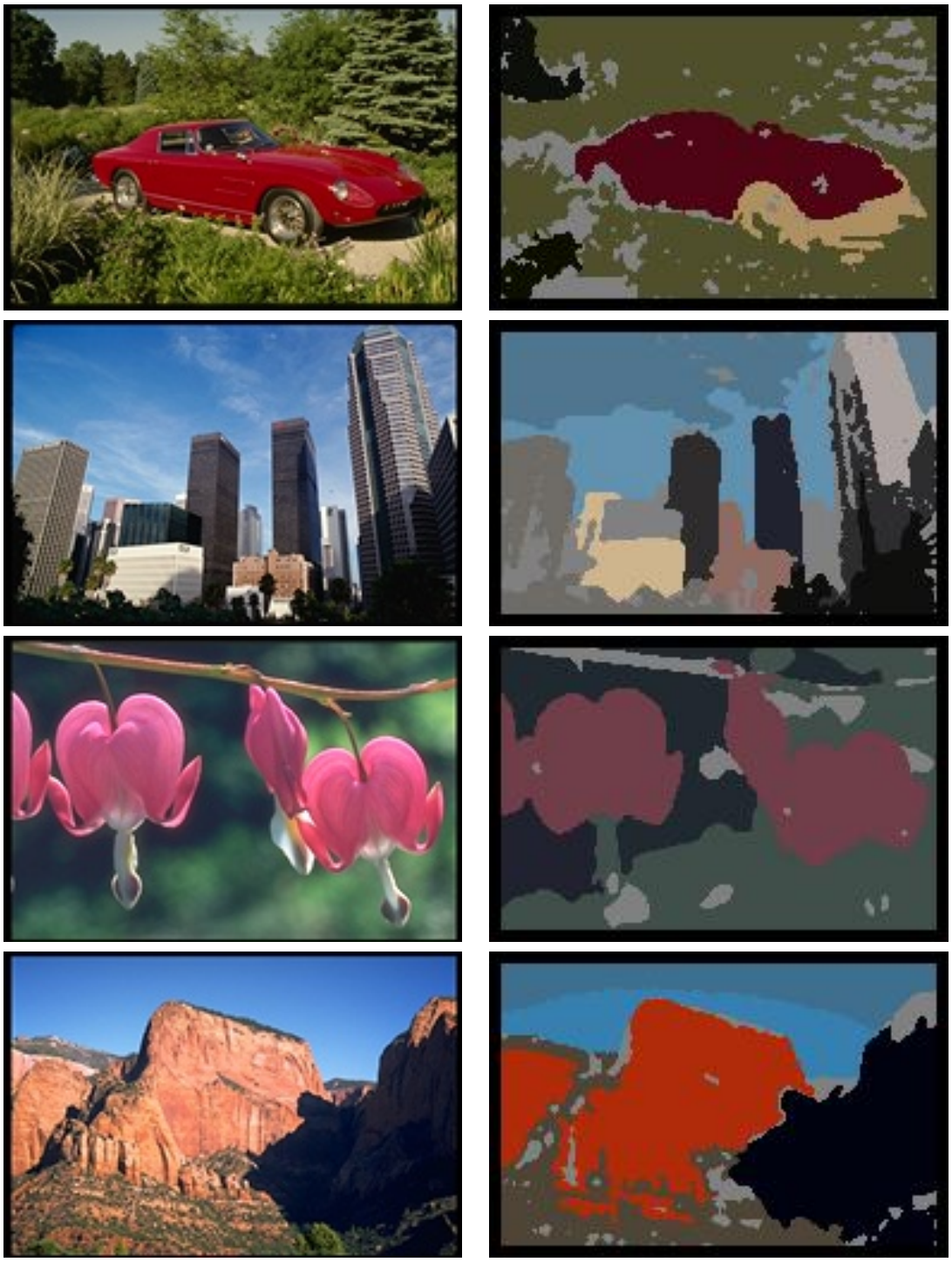


Figure 2.3: Four images and their spatial segmentations computed by the Blobworld. For the car image, the semantic is preserved in the shape. However, the city, flower and outdoor scenes result in segmentations that are not useful for retrieval. Images used courtesy of (Lui, 2002).

ster *et al.*, 1977). An object lookup table is then constructed for each blob using the colour, texture and shape of the segment. Object-based image retrieval is achieved by the selection of a query-blob from a user presented frame, and then blob comparison (Carson *et al.*, 2002). The approach is dependent upon the segmentation of meaningful blobs and on retrieval being object-dependent. In Figure 2.3, four image segmentations are shown. The car object produces a distinctive shape that is good for matching however the other segmentations are poor. Many succesful algorithms exist for image segmentation, for example Normalised Cuts (Shi and Malik, 2000) and the Watershed (Beucher and Meyer, 1993), however the general problem remains unsolved for all cases.

The choice of which features to use is a complex issue formally known as the feature selection problem. An optimal set of features are ones in which known different classes are maximally separable (Sebestyen, 1962). Feature selection for large data sets is also affected by the *Curse of dimensionality* problem (Bellman, 1961). As the size of the feature space increases the ability to find an optimal feature set diminishes. As a consequence, it is often the case for image and video indexing that a small number of features are chosen that is known to perform optimally for a specific domain.

2.3.3 Video feature based approach

Unlike an image, a video is a large, non-compact chunk of data. During indexing, the first task is to determine what elements are to be described as index items. Some indices are required to describe every frame, however in most cases frame groups called segments are extracted as index items (see Section 2.4). If no structure exists, a uniform temporal segmentation of a video can be used to provide equally divided small temporal units of content.

Following from the success of image retrieval systems, a common approach to video indexing and retrieval is to select a key-frame for a segment and then perform static image indexing. Commonly, the first, mid, last, or n^{th} frame of a segment is used. However, these frames do not consider the segment content and can therefore be unrepresentative. Many approaches to content-based key-frame selection have been reported in literature. In early work by Gunsel *et al.* (1997), a mean colour histogram is computed using all of

the frames, and the key-frame is selected as that with the closest colour histogram. Similarly, in Zhuang *et al.* (1998) a frame content space is clustered and the centroids from the largest clusters used. In Wolf (1996), an assumption is made that camera stillness is an indicator of frame importance. Therefore, optic-flow is computed and the frame associated with minimum flow field is used. Unfortunately, such an assumption only holds for manually created video. In Zhao *et al.* (2000), each frame is projected into a content space and frames at the corners are used.

An important consideration for key-frame selection is how many frames to employ. One approach is to use any knowledge about the visual characteristics of a segment, e.g. if the segment is zooming then the first and last frames can be used, but this both simplistic and ad hoc. In Porter *et al.* (2003b), several frames were selected according to frame overlap. Frames are selected that show different background material using a block-based motion algorithm. Rather than employing multiple key-frames, Sawhney and Ayer (1996) constructed a single mosaic of the background location using the shot frames, from which a static feature-based index is computed. However, this approach does not consider foreground objects or their temporal context.

In general, static features provide only a limited description of the segment content because they do not consider the *temporal context*. To overcome this problem, a number of approaches have been proposed that retain the temporal character of a video segment. In (Vinod, 1998), a shot activity histogram is computed and used. Each frame is represented using an optic-flow based estimation of the level of frame activity. In the VideoQ system of (Chang *et al.*, 1998a; 1998b), a Query-by-sketch (QBS) motion-trail based retrieval system is described. The sketch may contain colour, texture, shape or a spatial relationship between primitives, and their transformation over time. The QBS paradigm is attractive because it can solve the initialisation problem of QBE, described in Section 2.1.2. Unfortunately, it requires that the user has some sketching ability and a clear visualisation of the requirements. In Bimbo *et al.* (2000), 3D colour flows (blobs) are found and used. It requires that each frame is segmented to identify regions of homogeneous colour that are tracked over time. The approach is used to index television advertise-

ments that have a large amount of dynamic colour information. Similarly, in Sivic and Zisserman (2003), viewpoint invariant regions are found and tracked.

2.3.4 Semantic based approach

A semantic indexing system attempts to make the implicit knowledge in the scene explicit. This requires the detection and labelling of distinct objects, e.g. faces, cars or pedestrians, and behaviours, e.g. walk or run. Two classes of semantic may be distinguished:

Holistic semantics refer to the entire image, video, video frame or segment. For example, the whole image or video may labelled as indoor or outdoor, black-and-white or colour, have human faces or not.

Localised semantics refer to a particular spatial and/or temporal location. For example, a human face is present at (x,y,t) or a waving gesture occurs between $[t1,t2]$.

In Vailaya *et al.* (2001), images are classified as either indoor or outdoor using competing density estimations of visual features. Furthermore, once classified, pairs of classifiers are recursively applied to find more specific sub-classifications. Unfortunately, as the image progresses down the hierarchy, the error-rate cascades to unmanageable levels. It is also clear that a binary holistic classification is not useful for searching a large search space. In Wang *et al.* (2001), a holistic image classification is used to prune a gigantic search space prior to a more specific and expensive shape-based matching.

To avoid the use of hand-labelled data in supervised learning, Xie *et al.* (2003) adopted a video mining approach to automatically group and learn content phases in video. Low-level colour-based features are extracted and a hierarchical hidden Markov model was employed to perform temporal grouping and labelling. The classification rates were compared favourably against a hidden Markov model trained with hand-labelled data. However, the domain used was very constrained and the model size was small. It remains unclear as to whether the approach will scale to more meaningful data.

The ImageScape image retrieval system (Buijs and Lew, 1999; Lew and Sebe, 2000; Lew, 2000; Queries, 2002) learns a small number of localised semantics, such as [human

face, sky, stone, tree, water], using supervised training. The image descriptor consists of several spatial semantic indicators and retrieval is achieved using a Query-by-icon (QBI) paradigm: the user arranges a number of icons that correspond to the semantics on a palette; it is matched to the index using presence-of-semantic and spatial-position. Unfortunately, the small number of rather simple semantics result in a lack of expressiveness in the query language and matching process.

In Feng *et al.* (2002), pixels are labelled using a Multiple-Layered Perceptron (MLP) trained with hand-labelled outdoor data. To improve the result using spatial context, i.e. the sky occurs towards the top of an image, the local classification results are reprocessed through a Tree-Structured Belief Network trained using Maximum Likelihood (ML). Connected-components algorithm is used to find large blobs of similarly labelled pixels that become the semantics. Unfortunately, the approach suffers from large quantisation problems because of the enforced structure of the belief network.

The discovery of localised semantics in video is akin to dynamic scene understanding (Ullman, 2000; Gong *et al.*, 2000; Ng, 2002). An approach is required to perform background maintenance, object detection and tracking, and then object or activity recognition. The detection of changing pixels, called *temporal change*, provides information about the spatial positions of moving objects. However it is vulnerable to noise. Background maintenance reduces noise by modelling expected content of scene appearance, for example by a Gaussian Mixture model for each pixel (Stauffer and Grimson, 2000).

In Bobick and Davis (2001), the temporal characteristics and shape of pixel changes are modelled using a moment feature space and is used to recognise aerobic exercises. Unfortunately, the system requires clearly distinguishable activities given in clean datasets. It is not clear whether the approach could be trained using more realistic data. Hidden Markov Models (HMM) are a popular graph model used for temporal recognition because they offer dynamic time warping, efficient training algorithms, and clear Bayesian semantics (Rabiner, 1989). HMMs are known to perform well in controlled environments such as with gestures (Psarrou *et al.*, 2002), but are dependent upon a strict temporal order in the observed action. In (Brand *et al.*, 1997; Oliver *et al.*, 2000), a

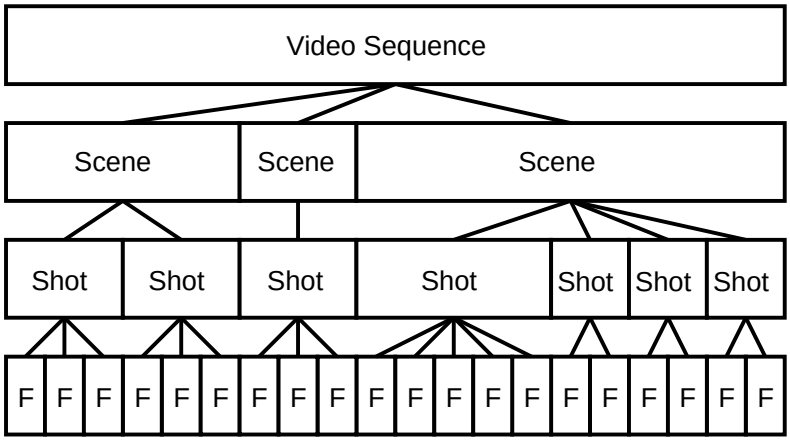


Figure 2.4: The four-level video structure. The frames are grouped into shots; the shots are grouped into scenes. Shot-breaks occur at camera capture discontinuity, whereas scene-breaks occur at semantic discontinuity. The top-level is the sequence.

Coupled HMM is used to successfully label outdoor activities. Similarly, in Gong and Xiang (2003), blobs of connected temporal change are found in outdoor surveillance data. These are identified using a dynamic Bayesian belief network - with the features primarily based upon spatial location and the size/shape of the bounding box. Unfortunately, the approach was limited to semi-structured scenes with repeating activity.

2.4 Video structure discovery

A video sequence consists of a set of temporally ordered frames that, when shown sequentially, the *Human Vision System* interpret as a moving image. Neighbouring frames are often similar, especially when a high number of frames per second was captured, leading to computational and perceptual difficulties. As Human understanding corresponds better to smaller and more semantic units and themes, a four-level hierarchy illustrated in Figure 2.4 is widely employed (Bimbo, 2000; Lew, 2001).

At the lowest level, the set of frames, a physical sequence is implemented. A *frame* is an atomic unit in the temporal domain and cannot be further divided. A *shot* is a group of frames that are captured continuously from the same camera without interruption. Shots are prevalent in highly structured video domains, such as newscasts, adverts, drama, entertainment, but less so in other domains such as sport and surveillance. However, for semantic-sensitive applications, shots still present a too low-level unit for Human

understanding. Shots are therefore grouped into *scenes*. A scene is a set of shots that exhibit a common semantic, thread or story-line. This hierarchy exploits the idea that, in most structured video, a sequence is artificially built during a manual editing process. As shots and scenes have the same physical structure, i.e. they both consist of a group of neighbouring frames, the generic term *segment* is used.

The first step in video indexing is to automatically discover its structure. The intermediate levels, shots and scenes, are discovered during *temporal segmentation*[5]. Here, the important works are now reviewed. For a comprehensive review of shot transition detection the reader is directed to (Lefèvre *et al.*, 2003).

2.4.1 Shot transition detection

The detection of shots has received widespread attention since the early 1990s with most effort concentrated on more commercial domains, for example in drama and television advertisements. We distinguish between two main types of shot transition:

- A sharp break, or cut, occurs when the change between shots occurs instantly.
- A gradual break occurs when, in the editing suite, an algorithm is used to visually enhance the transition, for example a wipe, fade or dissolve.

A robust method needs to address both transition types. In general, shot transition detection can be classified into five categories: pixel-based, histogram-based, block-based, compressed-domain, and model-based.

A number of early works detected sharp shot transitions using the level of difference in the pixels between frames. Early methods detected transitions by comparing the sum of intra-frame pixel-differences against a threshold (Nagasaka and Tanaka, 1991; Kikukawa and Kawafuchi, 1992). Similarly in (Zhang *et al.*, 1993), the number of pixels with change above a threshold is used. Such pixel-based comparison methods are highly sensitive to object and camera motions. As such, in (Zhang *et al.*, 1993) a

[5]In literature, this process may be referred to as video structure parsing, shot detection, scene detection, camera-break detection or shot transition detection.

3×3 smoothing filter is first applied to the image. Alternatively, in (Shahraray, 1995) a motion-compensated pixel difference is computed.

To provide greater perceptual robustness, a histogram representation can be used. A histogram can be computed for each frame, using either grey-scale or colour information, and is robust to camera and object motions. Frame comparison can then be made using a metric such as the histogram intersection or χ^2 distance (Lew, 2001). In the seminal work by Zhang *et al.* (1993), a twin comparison method was proposed to find sharp and gradual transitions in a single pass. However, a number of sensitive thresholds are needed to obtain a good result: one threshold to detect the sharp transitions; another to detect the gradual transitions using cumulative difference. This method is often reported in literature and is considered successful, for example (Boreczky and Rowe, 1996). However global histogram methods are known to fail when frames from different shots are close in representation space. Consequently, it will often miss a transition.

As pixel-based methods are considered too sensitive to noise and global histogram-based methods too sensitive to similar looking shots, an intermediate approach can be adopted by splitting the image into blocks. A block-based approach also has computational advantages. Nagasaka and Tanaka (1991) proposed an extension to their pixel-based approach that operated on frames divided into blocks of size 4×4. A transition is detected by (a) computing all the distances between the respective block histograms, (b) ordering the distances and retaining the lowest eight, and (c) comparing the average retained distance against a threshold. In a recent study, Porter *et al.* (2003a) used block-based motion estimation to track blocks through a sequence and to identify gradual transitions. It attempts to distinguish changes caused by transitions from those caused by camera and object motions at block-level. However, the method is not invariant to multiple different motions within a block. It is also computationally expensive.

A number of approaches have been proposed that partition video in a compressed domain using, for instance, the standards from the Moving Picture Experts Group (MPEG). This is potentially advantageous because the motion features are already computed during the temporal compression process. In Arman *et al.* (1993), the normalised inner

product of the coefficients between frames is compared. Similarly, in Zhang *et al.* (1994), the coefficients from corresponding blocks between frames are compared, using a number of thresholds in a modified twin comparison method. In both cases, the algorithms operate on the I-frames[6] leading to a loss of temporal precision. In Meng *et al.* (1995), the B-frame information is used.

It is important to note that all of the approaches above present a bottom-up style solution to the problem. A top-down approach is also possible, whereby implicit knowledge about the appearance of a gradual transition is encoded and recognised. For example, in (Hampapur *et al.*, 1995) a number of gradual transition types are modelled. However, we consider that such an approach is invalid in most unconstrained non-artificial cases.

2.4.2 Scene change detection

Once the set of shots is established, the next task is to examine them for semantic similarities in order to perform perceptual grouping into scenes. In general, there are three approaches: model-based, visual-similarity based, and temporal-context based.

In Aigraine *et al.* (1997), scene breaks are found using domain specific rules relating to editing techniques and film theory. For example, certain types of shot transition may indicate that the next shot is the beginning of a new scene. However, such rules are inflexible, require significant knowledge about the domain, and are only applicable to constrained sequences. A model-based approach is often too dependent on video content following pre-determined expectations.

Numerous works have proposed using visual-similarity to perform a bottom-up shot merging, in order to form scenes. In Rui *et al.* (1998), the first and last frames in the shot are used as key-frames. Colour histograms are extracted along with a measure of global shot activity. Shot similarity is then defined as a combination of feature similarity and temporal attraction. Shot groups and scenes are found using a time adaptive merging algorithm. The approach provides a basis for semi-automatic structure discovery using visual features. However, it is over dependant on colour leading to structural errors. In

[6]Refer to MPEG for more details on the compressed video format. In short, an I-frame is a complete image frame that occurs periodically in order to provide basis. A B-frame is encoded relative to the basis provided in both directions.

Hanjalic *et al.* (1999) it is argued that movies are organised around events, and that shots are either a part of an event (event shots) or serve for its description (descriptive shots), for example by showing the location of where an event is taking place. Links are formed between visually similar descriptive shots using an adaptive threshold. The shot transitions over which a low number of links pass is identified as a scene-break. This approach assumes that shot groups are encapsulated by similar shots. However, this does not often hold.

It is not sufficient to use visual-similarity alone when performing shot grouping, because shots in different scenes can be close in the feature space. In the literature, a number of approaches (Yeung *et al.*, 1996; Kender and Yeo, 1998; Lin and Zhang, 2000; Lin *et al.*, 2001) use temporal context when performing shot-scene assignment. By temporal context, we mean the examination of shot content in relation to its near and far neighbours over time. Once shots have been detected, a video can be represented as a sequence of symbols, e.g. *ABABCDCD*, where each letter corresponds to a shot label. Intuitively, the process of scene detection is required to find the points at which the future no longer looks like the past. So, in the example, the break is at time instant 4 because *A*s and *B*s become *C*s and *D*s. Such an approach requires shot clustering, labelling, and then a temporal analysis to discover the scene breaks.

In Yeung *et al.* (1996), the shots are represented using visual primitives, and hierarchical clustering is performed where, at each step, the two most similar clusters are merged. Shot similarity is computed using a feature-based metric and a temporal constraint. A *scene transition graph* is then constructed with nodes representing the clusters and edges capturing the level of temporal transition between nodes. The previous example is decomposed into a graph $A \Leftrightarrow B \Rightarrow C \Leftrightarrow D$. This graph is typically dense. Scene-transitions are identified where the graph is thin, i.e. in this example, between *B* and *C*, which is correct. A temporal constraint is introduced to handle longer sequences with repeating shot types. This approach was applied successfully to videos of situation comedy. However, it is a fundamentally discrete process reliant on accurate shot clustering and labelling. If small variations in visual features lead to a different shot label

assignment, the graph can be distorted and produce errors.

In Kender and Yeo (1998), a continuous approach is presented that, at each shot, estimates the level of similarity between the past and the future. As the video is parsed, a model of short-term memory is maintained using a shot buffer. Older shots are leaked from the buffer in a non-linear time-ordered manner. A shot recall value is computed for the incoming shot, that captures the similarity between it and the buffer. A normalised shot recall is used to compute a normalised measure of video coherence. Minima in coherence are identified as the scene-transitions. Unfortunately, the choice of buffer size has severe consequences for the results. Also, performing many shot-shot content comparisons can be computationally demanding.

A conceptually similar but simpler approach was proposed by Lin and Zhang (2000). The dominant colour motion in a shot is estimated and represented in a histogram. The histogram intersection distance is used for shot comparison. To establish the scene boundaries, at each shot the left and right shot attraction is computed using a local temporal context of three shots. If the attraction of a shot from the right (the future) is greater than the attraction from the left (the past), then a shot is allocated to a new scene. In their extended work (Lin *et al.*, 2001), the force competition approach, the ratio of the left and right attraction is used as a splitting force. A complementary merging force is computed using the similarity of the right shots to the left shots. Scene boundaries are generally found when the splitting force is at a maximum and the merging force is at a minimum. The approach is elegant. However, it relies on colour information being present and useful, and on a small temporal context. The implications of context size are not evaluated.

The shot and scene detection techniques outlined above are valid for constrained, well-structured broadcast quality sequences, such as film, news and sport programmes. However, the assumption that structure exists, and that the number of shots exceeds the number of scenes, do not necessarily hold for unstructured domains, such as home video or in surveillance. For example, during 24 hours of continuous capture, a surveillance camera may capture many natural scenes but will have no shot transitions.

2.5 Surveillance

In the digital age, Closed Circuit Tele-Vision (CCTV) surveillance cameras are ubiquitous. Although the exact number of cameras in the United Kingdom is not known, it is estimated to be around 4 million with $400,000$ of these in London (McCahill and Norris, 2002). These cameras exist to monitor and record activity and results in an extremely large amount of footage needing to be stored, sorted and processed reliably, in order to satisfy its purpose. The cognitive skills required by a successful surveillance operative include physical capacity, sensory recognition, perceptual processing, observation skills and sustained attention (Donald, 1999). Currently, these skills are compromised by relentless expansion. A fully automated or semi-automatic visual surveillance system is highly desirable and, in recent years, has attracted heavy investment[7].

In particular, current research is interested in the detection of abnormal phases of content. In Dee and Hogg (2004), an attempt is made to explain normal human behaviours in a car-park scene using a goal-based approach. It is assumed that human behaviour is always explainable, for example a person in the scene must walk to one of the exits or paying stations. However, the approach does require manual labelling and configuration. In Nait-Charif and McKenna (2004), unusual activities are detected in a home supportive environment, for example an elderly patient has stopped moving in a particular zone and intervention is required. This is achieved by, firstly, extracting object motion trajectories by grouping pixels of temporal change, a trajectory speed feature extraction, and then clustering to find spatial-zones of expected zero-motion. A rule based detection is then used to detect important zero-motion events, i.e. if the speed is less than a predetermined threshold in a particular zone then the status is abnormal.

One problem in surveillance scenes is that they often contain many static structures that cause occlusion, and so hampering tracking algorithms. Consequently, many

[7]The following projects are indicative: RETRIEVE - RealtimE Tagging and Retrieval of Images Eligible for use as Video Evidence; REVEAL - Recovering Evidence from Video by fusing Video Evidence Thesaurus and Video Meta-Data; GENERICK - Generation, Encoding and Retrieval of CCTV-derived Knowledge; ICONS - Incident Recognition for Surveillance and Security; VIGILANT - Intelligent Real-time Storage and Retrieval of Surveillance Video; INSIGHT - Video Analysis and Selective Zooming using Semantic Models of Human Presence and Activity.

surveillance video indexing systems avoid activity modelling and instead exploit only spatial and geometric information about the scene, for example (Makris and Ellis, 2002; Zhong *et al.*, 2004). In Greenhill *et al.* (2004), an occlusion landscape is built for a scene using a depth-based probability density function to model each pixel. The depth space is populated using estimates of moving object ground positions. The landscape is used during inter-frame correspondence to reduce occlusion effects.

2.6 Discussion

There is an urgent need for systems that can perform automatic analysis and pre-attentive filtering of surveillance video. Unfortunately, indexing is problematic owing to significant sensory issues:

- The quality of the scene capture can sometimes be poor because of the recording device and storage medium used. This leads to a weak feature landscape. For example, it has been suggested that very little useful information exists in the colour space captured by commercial CCTV (Gong and Xiang, 2003).

- Surveillance system operate in all weather conditions, 24 hours a day. This causes very large changes in scene appearance. (See examples in Figure 2.5).

- The rough nature of the domain: low-bandwidth during transmission; camera shake; dirty lens (although modern camera mounts may be equipped with wash-wipe facilities), all contribute to poor data quality.

- Elaborative multi-media data (synchronised audio, text and video) is not available. Multi-modal techniques, for example that exploit audio information, are not possible.

These issues are exacerbated by the fact that captured sequences are often very long, and contain little or no interesting content. Many visual representation approaches exploit colour, shape and texture information. However, in surveillance such features do not correspond to the main requirement (“what is happening”) are therefore are not useful.

Figure 2.5: Four frames from a road junction scene showing the effect of different lighting conditions. (top) The scene in normal conditions with diffuse lighting. (bottom left) The scene with direct sunlight. Strong shadows occur and are clearly seen. This effect hampers object detection in dark areas and also accentuates unwanted tree motions. (bottom right) The scene at night. It presents a significantly different visual appearance to the daylight scene.

Figure 2.6: An illustrative frame from two different outdoor surveillance scenes. Vehicles in the scene are constrained to operate on road surfaces and particular paths leading to very little useful trajectory information. Pedestrians can operate more freely but are small in the visual field leading to detection difficulties.

An active research area in computer vision is tracking. Although tracking will provide information on object motions in real-time, it is not computationally feasible to apply tracking algorithms, such as CONDENSATION (Isard and Blake, 1998), to such large-scale data. Also, segmented motion trails are not always informative when performing visual search. (See Figure 1.2 on page 12 for an example of trajectories that can be extracted from an indoor scene. See Figure 2.6 for examples of outdoor scenes in which trajectories are difficult to extract and constrained). Even if trajectory information could be extracted reliably, it would almost always follow the same trail, as in surveillance situations object motion often follows a similar constrained path. For example, cars are constrained to follow roads and pedestrians will usually follow the most sensible route that satisifies their objective (Dee and Hogg, 2004). A system that performs video analysis without specifically modelling object-level content or performs tracking can be highly effective and also computationally more robust.

As described in Section 2.4, temporal segmentation is an important initial phase of indexing. However, in a surveillance situation no artificially induced structure exists, i.e. there are no sharp/gradual shot transitions or meaningful shot groupings. We are therefore required to find a more subtle and fundamentally natural content change in long continuous video using action features. Nevertheless, if natural structure can be found in surveillance video it could be used to provide content access and visualisation.

In Chapter 3, an approach is presented for extracting action features from surveillance video. A video index is constructed that is used for the tasks of search and temporal segmentation. The approach is considered to be *pre-attentive* because it makes no assumptions about the scene content. In Chapter 4, a representation is presented that uses a wavelet-based descriptor to extract information about local regions (cells) of activity. An *Iconic index* is constructed that is used for temporal segmentation and sequence visualisation. An approach for video summarisation is also presented. In Chapter 5, an approach for the integrating of manual assessments with the index is presented. This forms a *Semantic graph* that is then used for sequence browsing.

Chapter 3

Pre-attentive video processing

Large volumes of surveillance video data exist and present a considerable and important indexing challenge. We consider the development of a *pre-attentive* system for indexing is a critical first step needed to facilitate visual search. By pre-attentive, we refer to a method that can operate quickly on large volumes of data, has no prior knowledge or artificial expectations of the content, and is able to operate without supervision. Such a system must determine what is pertinent and generate an index sufficient for later use by more directed algorithms. In effect, the system should filter out a large quantity of information while retaining the crucial parts.

As such, in this Chapter an action-based approach is proposed that generates a compact index, that is used for visual search, browsing and temporal segmentation. Firstly, in Section 3.2.1, a measure of temporal change is defined that is less sensitive to sensory problems and is sufficiently efficient and scalable. In Section 3.2.2, a frame representation is described that captures the spatio-temporal action context using a cellular grid. In Section 3.3, spatio-temporal connected cells, that are an indicator of scene content, are found and profiled to form an action-based index. In Section 3.4, the visual structure and temporal context of scene action is compared and a cost-based similarity metric formed. Finally, outdoor surveillance scenes are used to demonstrate our approach.

3.1 A sequence and its segmentation

To begin, the terms *scene* and *sequence* are clarified: a scene is a real-world environment in which action occurs; a sequence is a digital approximation of the scene content as observed and captured from a digital capture device. To perform scene analysis, it is approximated, digitised and stored into a number of sequences. A video sequence is defined as a set of *frames*:

$$\mathcal{F} = \{F_1, F_2, \ldots, F_n\} = \{F_t : \forall t \in [1,n]\} \tag{3.1}$$

where n is the number of frames[1]. Note that $\forall t$ refers to all frames. When the frames are viewed in strict sequential order they provide a digital approximation of the scene content. Each frame consists of a square grid of pixel positions:

$$F_t = \{F_t(x,y) : \forall x \in [1,X]; \forall y \in [1,Y]\} \tag{3.2}$$

where $\forall x$ and $\forall y$ to refer to the full spatial ranges. For clarity, a pixel position is referred to as $F_t(x,y) = F(x,y,t)$.

Considering the need for video structure as discussed in Section 2.4, in this Chapter we form an index and perform temporal segmentation. The temporal segmentation consists of a number of break positions:

$$\mathcal{B} = \{b_i : \forall i \in [1,m]; m < n; b_i \in [1,n]\} \tag{3.3}$$

where each break position b is the frame number of a discontinuity in the sequence, i.e. each b_i is in the range $[1,n]$. The number of breaks m is less than the number of frames n. The frames in the range $[b_i, b_{i+1}]$ are known as a *segment* and provides a larger and more semantic unit of sequence content (than a frame).

[1]Set notation is used throughout this thesis. For example, $X = \{x_i : \forall i \in [1,n]\}$ indicates the set X is comprised of n items. The term $[1,n]$ is used to refer to all values in the range 1 to n. Constraints appear after the colon and multiple constraints are separated by a semi-colon.

To perform a comparison of two temporal segmentations, a similarity metric is defined that considers the number of close break positions:

$$\sum_{i=1}^{i\leq m} \left(\min\left(\left\{|p_i - q_j| : \forall j \in [1,m]\right\}\right) > T_{within}\right) \tag{3.4}$$

where P and Q are the segmentations being compared, m is the number of breaks in each, and T_{within} is a matching distance for breaks. The result of the metric is between $[0,m]$ representing the number of mis-aligned breaks. Aligned segmentations will produce a low score. To compute the similarity between a segmentation and a set of segmentations, all the similarities are computed and the mean used.

3.2 Frame based video indexing

Owing to the nature of surveillance, and as we are more interested in "what is happening" in the scene rather than "what is present", we focus on action-based feature extraction methods. In the following Sections our approach for indexing a surveillance sequences at the frame level is described. We use the scenes illustrated in Figure 3.1.

3.2.1 A measure of reliable temporal change

The first task is to extract important features from a sequence by processing each frame in turn. Each pixel is initially represented as a vector, $F(x,y,t) = \langle RGB \rangle$, that represent the intensity of red, green and blue captured at that position. To begin, the frame is converted to the Hue Saturation Value (HSV) colour space:

$$H = 180\frac{0.5(R-G)+(R-B)}{((R-G)^2+(R-B)(G-B))^{1/2}} \tag{3.5}$$

$$S = 1 - \frac{3}{R+G+B}\min(R,G,B) \tag{3.6}$$

$$V = \frac{R+G+B}{3} \tag{3.7}$$

The values of Hue, $H \in [0,360^o]$, and Saturation, $S \in [0,1]$, provide perceptually meaningful colour information, and Value, $V \in [0,255]$, records the brightness.

The *Waving hand* scene shows a number of hand signals - squares, figure eights, triangles - used to simulate changing content behaviour.

The *Pets carpark* scene, obtained from the Performance Evaluation of Tracking and Surveillance workshop (PETS), shows a carpark scene containing car, bicycle and pedestrian activity. See (Ferryman, 2003).

An *Aircraft docking station* in a busy airport scene, obtained during the Incident Recognition for Surveillance and Security project (ICONS), shows an aircraft docking station at Heathrow airport, London. The sequence shows aircraft arrival and departure, unloading, loading, and many other natural activity content. See (QMUL, 2002).

Figure 3.1: Three different scenes used.

By observation, we find that the colour information computed in outdoor surveillance scenes is not expressive or consistent enough (at least with the majority of the current analogue systems). This is due to changeable holistic lighting conditions such as cloud coverage, the poor quality camera equipment used during surveillance capture, and frequently reused storage media. We also find that surveillance frames are mainly grey due to a high proportion of concrete, brick, tarmac and street furniture content. Consequently, it can be argued that colour is not sufficient for visual search tasks in outdoor video. This lack of useful colour is illustrated in Figure 3.2. The HS space computed for ten images from an image dataset is compared to the HS space computed for ten frames from an Aircraft docking scene. It can be seen that, in contrast to that computed for the images, the colour information computed for the surveillance frames is lacking richness. Henceforth, in this work, the HS information is discarded and the approach is built using the brightness information, V. It must be stated that numerous other colour spaces and brightness estimation techniques exist, for example YUV. However, we use the HSV brightness value due to its computational simplicity and common use in literature (Bimbo, 1999; Gong *et al.*, 2000).

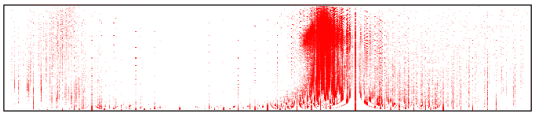
Assuming a fixed camera position, as commonly found for surveillance mounts, a sequence action is approximated using pixel-wise difference between successive frames. The thresholded *temporal difference* as computed for frame t:

$$\left\{ \left(|F(x,y,t) - F(x,y,t-1)| > T_{diff} \right) : \forall x \in [1,X]; \forall y \in [1,Y] \right\} \tag{3.8}$$

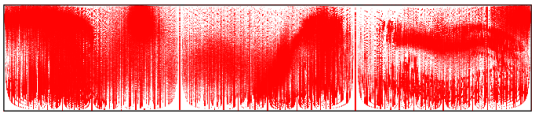
provides the position, shape and intensity of activity in the scene at a frame t. T_{diff} is a threshold that can be tuned according to the application. Such an approach is popular in literature because of its and inexpensive cost (Bobick and Davis, 2001; Gong and Xiang, 2003) in comparison with a background maintenance approach such as (Stauffer and Grimson, 2000). Unfortunately, as seen in Figure 3.3, the approach is vulnerable to sensory problems that produce rogue active pixels. This is due to camera shake, image coding and transmission errors, quantisation and sampling problems, and also the potential presence of a surveillance time-stamp and other embedded meta-data.

(a) A mountain scene.

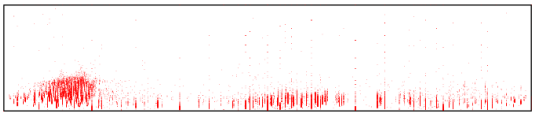
(b) Nine further natural images.



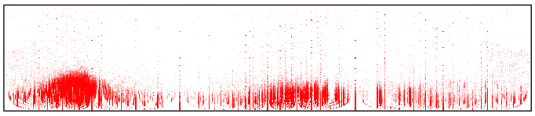
(c) Hue-Saturation for a.



(d) HS for a and b.



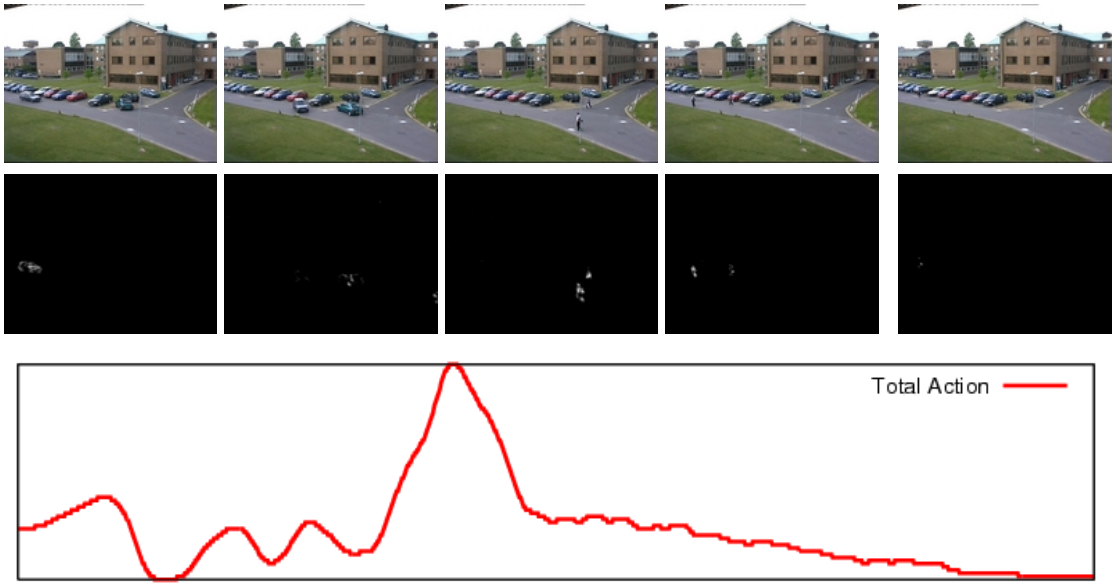
(e) HS for an Aircraft frame.



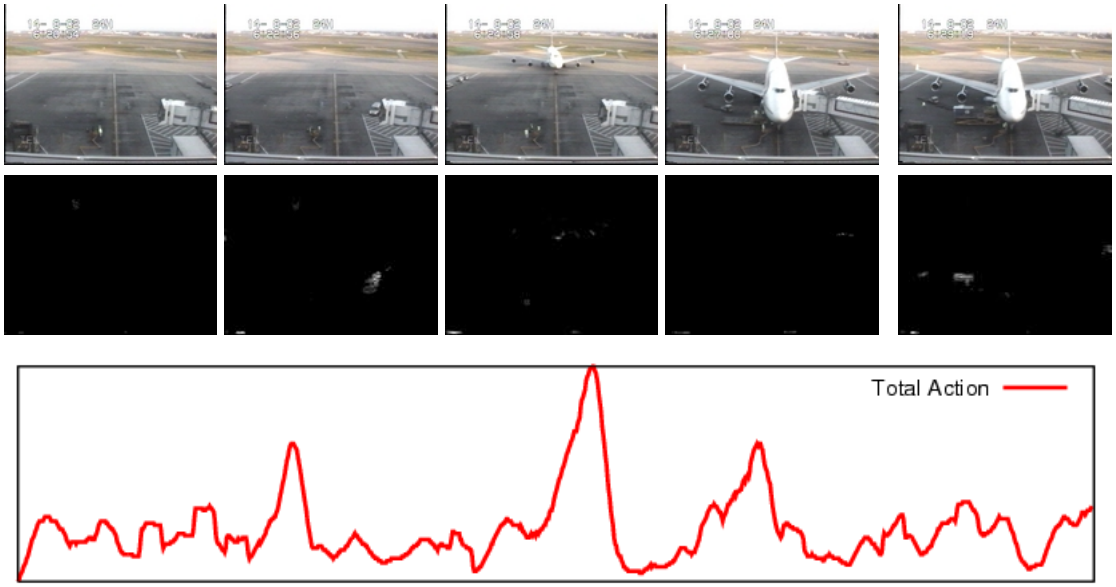
(f) HS for 10 Aircraft frames.

Figure 3.2: A comparison of the colour space available in image retrieval with visual surveillance scene. In (a) we show a mountain scene from a dataset of images. Nine more images are shown in (b). We computed the Hue-Saturation-Value (HSV) for each image and we show the HS colour space for a the mountain scene image in (c) and all ten images in (d). Value is discarded because it is concerned with brightness rather than colour. The x-axis corresponds to Hue (0-360^o) and the y-axis corresponds to Saturation $[0,1]$. In (c) it can be seen that the mountain scene contains a distinctive formulation of colour saturation. Furthermore, in (d) it can be seen that the variation of colour saturation amongst the dataset is high. This clearly illustrates that colour is able to discriminate in this dataset. In (e) we show the HS colour space for a frame at the Aircraft docking scene shown in Figure 3.1 and in (f) we show the HS colour space computed for ten such frames (that were not close in temporal space). It can be seen that the saturation is low, meaning that very little colour information exists. It is also seen that the addition of new frames provides little new information from which to discriminate. This is because all the frames contain highly similar colour content due to similar background information. We conclude that colour is insufficient for searching this scene.

Figure 3.3: The computation of action features illustrated in an extract from the Pets scene. (top) Two frames showing a car moving through the scene. (second row) The temporal difference. Although it captures the position and shape characteristics of the motion, it is vulnerable to noise. (third row) The thresholded temporal difference as in Equation (3.8). The important information is filtered, however noise is prevalent. (bottom) The Sustained temporal change as in Equation (3.9). It can be seen that the position and shape characteristics are present, and noise is removed. Crucially, the same action in the scene can be observed to produce a similar set of features.



The total scene action for 1000 frames at the Pets scene.



The total scene action for 1000 frames at the Aircraft scene.

Figure 3.4: An illustration of the total scene action metric of Equation (3.10). We show the total scene action for two outdoor scenes computed using 1000 frames. We also show the frame and Sustained temporal change content from Equation (3.9) for 5 equidistant positions. (Pets) The peak in scene action is seen to occur when two cars negotiate over a carpark place. (Aircraft) The peak in the scene action is seen to occur when the aircraft arrives. In both cases, the measure provides a useful indicator of when the scene was active and when the scene was *not*. This can be used to either focus a search towards or away from particular frames.

We achieve noise reduction using independent spatio and temporal filtering. Firstly, frames are spatially smoothed using an approximated Gaussian filter in order to reduce the effect of pixel outliers caused by the sensory problems. Inexpensive temporal filtering is then employed to smooth the result:

$$D_{\alpha,\beta}(x,y,t) = \begin{cases} \min(D_{\alpha,\beta}(x,y,t-1)+\alpha,1) & \text{if } |F(x,y,t)-F(x,y,t-1)| > T_{diff} \\ \max(D_{\alpha,\beta}(x,y,t-1)-\beta,0) & \text{otherwise} \end{cases} \tag{3.9}$$

where α and β are accumulation and decay factors. The computation is initialised with zero action, $D_{\alpha,\beta}(x,y,0) = 0; \forall x; \forall y$. The choice of parameters α, β and T_{diff}, is made according to domain and computational factors. A suitable selection is discussed in Section 3.6. The result value $D_{\alpha,\beta}$ is between $[0,1]$ where a high value indicates that a period of *sustained change* has taken place. Henceforth $D_{\alpha,\beta}$ is called the "Sustained temporal change". The approach provides a robust feature landscape for outdoor scenes as seen in Figure 3.3. Also, the independent spatial and temporal filtering is efficient compared to a combined spatio-temporal filtering, for example the approach by Chomat *et al.* (2000). This is because the computation at each frame uses the result of the previous frame, meaning that little computation is duplicated.

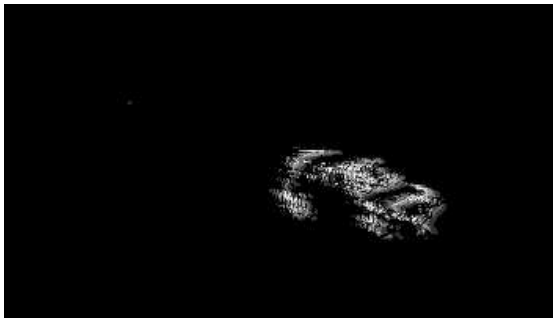
We also compute the following estimate of total scene action:

$$TotalD_{\alpha,\beta}(t) = \sum^{X}\sum^{Y} D_{\alpha,\beta}(x,y,t) \tag{3.10}$$

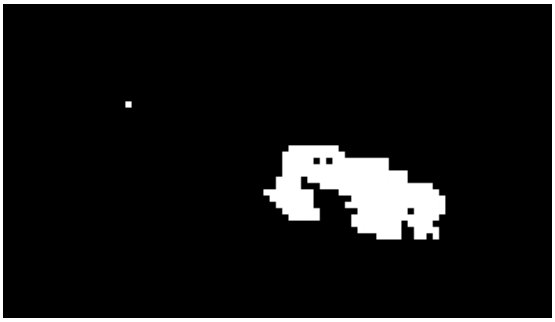
as seen in Figure 3.4. It provides an indication of the frames in which no action is occurring and is a useful pre-attentive indicator.

3.2.2 Grid based frame descriptor

As regions provide greater spatial context than individual pixels and correspond better to the moving object content, the image space is divided into a regular, static grid of square cells of equal size. A binary measure of cell activity is computed for each cell using a



The Sustained temporal change.

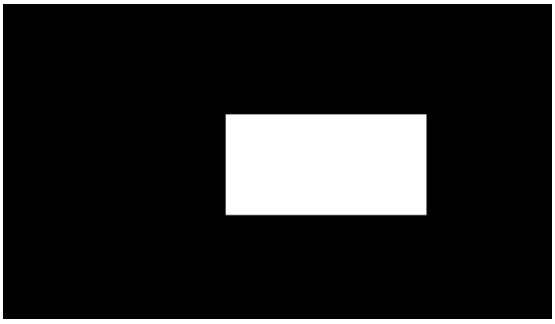


The grid with cellsize $\lambda = 4$.

The grid with cellsize $\lambda = 8$.

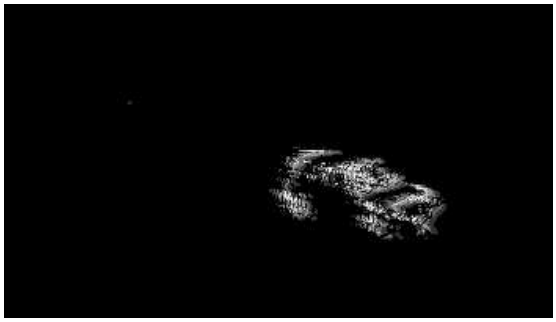
The grid with cellsize $\lambda = 16$.

The grid with cellsize $\lambda = 32$.

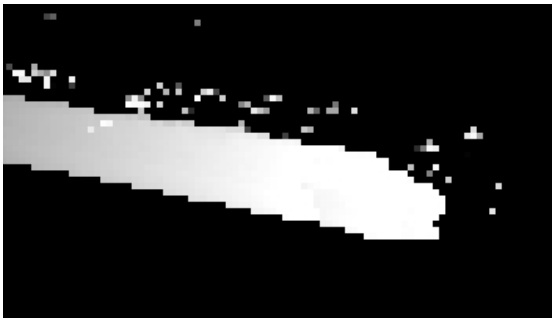


The grid with cellsize $\lambda = 64$.

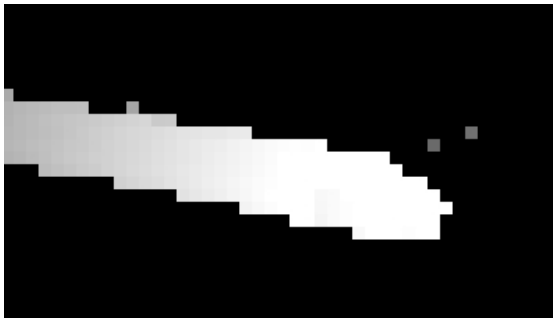
Figure 3.5: The grid computed using various cellsizes. Using the Pets scene extract from Figure 3.3. It can be seen that the grid preserves the location and shape context of the activity. The finer granularity - those with a smaller cellsize - provide more detail, at the cost of being a larger representation and more computationally expensive.



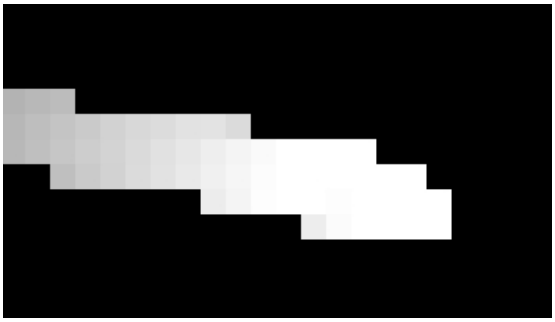
The Sustained temporal change.



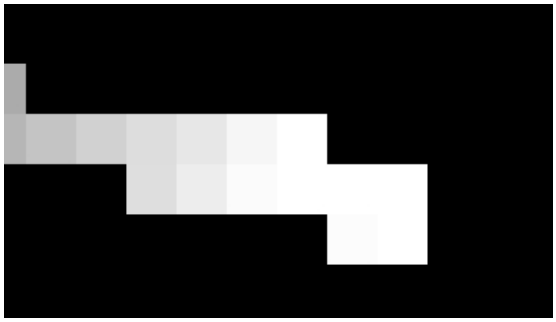
The history with cellsize $\lambda = 4$.



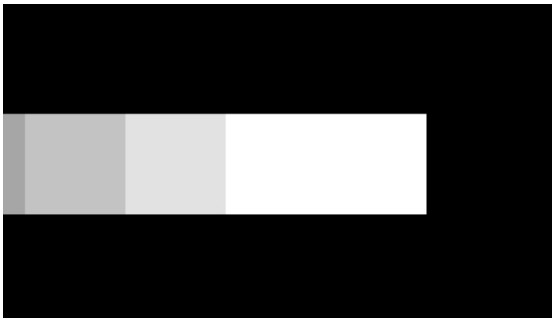
The history with cellsize $\lambda = 8$.



The history with cellsize $\lambda = 16$.



The history with cellsize $\lambda = 32$.



The history with cellsize $\lambda = 64$.

Figure 3.6: The history grid computed using various cellsizes. Using the Pets scene extract from Figure 3.3. It can be seen that the history preserves a temporal memory of recent occurrence. The lighter colour squares correspond to the current grid action, as Figure 3.5. The darker colour squares correspond to previous action in memory: the darker the square the more distant the cell action.

ratio-of-occupancy of active to inactive pixels:

$$Cell_{\alpha,\beta,\lambda}(cx,cy,t) = \left(\sum_{i=0}^{i<\lambda} \sum_{j=0}^{j<\lambda} \left(D_{\alpha,\beta}(cx\lambda + i, cy\lambda + j, t) > T_{pixel} \right) \right) > T_{cell} \quad (3.11)$$

where λ is the cellsize, cx and cy are the cell spatial position, T_{pixel} and T_{cell} are a pair of thresholds used that ensure the block is marked active only if considerable activity is present. A suitable choice is discussed in Section 3.6. λ is selected according to the desired coarseness of the representation: a larger value produces a more compact representation suitable for large-scale surveillance indexing. The result is an efficient and compact indicator of scene action. An example of the result using different cellsizes can be seen in Figure 3.5. We also compute the total number of active cells as:

$$TotalCell_{\alpha,\beta,\lambda}(t) = \sum^{CX}\sum^{CY} Cell_{\alpha,\beta,\lambda}(cx,cy,t) \quad (3.12)$$

where CX and CY are the number of cells in each dimension.

3.2.3 Frame spatio-temporal context

Whereas Equation (3.11) captures the spatial position and visual structure of active cells in the scene, an important consideration is the history of cell activity. The history provides information on the spatio-temporal context of action and improves the potential understanding of neighbouring cells. We therefore compute a measure for each cell using a temporal displacement of previous cell activity:

$$\begin{aligned} &History_{\alpha,\beta,\lambda}(cx,cy,t) = \\ &\qquad t - \max\left(\left\{ i : \forall i \in [t - T_{delay}, t-1]; Cell_{\alpha,\beta,\lambda}(cx,cy,i) = 1 \right\} \right) \end{aligned} \quad (3.13)$$

resulting in a value in the range $[0, T_{delay}]$. A low value indicates recent cell activity, a high value indicates cell inactivity, and a value of T_{delay} indicates that the cell has not been active within current memory. T_{delay} is the largest delay permitted. An example of the computation can be seen in Table 3.1 and the result can be seen in Figure 3.6.

0	0	0	1	0	1	1
1	1	0	0	1	1	1
1	0	1	1	0	0	3
1	1	0	0	0	0	5
0	0	0	0	0	0	T_{delay}
t-6	t-5	t-4	t-3	t-2	t-1	*History*

Table 3.1: An illustration of the computation of the cell history. We show the value of *Cell* from Equation (3.11) for the positions $[t-6, t-1]$ and the *History* result from Equation (3.13). It can be seen that recently active cells produce a low result.

t=1	t=2	...	t=n
TotalD(t)	TotalD(t)		TotalD(t)
Cell(x,y,t)	Cell(x,y,t)		Cell(x,y,t)
History(x,y,t)	History(x,y,t)		History(x,y,t)

Table 3.2: A summary of the frame index. For each frame in the sequence, an index item is computed using the total frame activity from Equation (3.10), the cellular grid from Equation (3.11), and the cell history from Equation (3.13).

To summarise, we compute and stored a frame-based index $FrameIndex(\mathcal{F}, \theta)$ for a sequence $\mathcal{F}$ as illustrated in Table 3.2. The tuning parameters and thresholds $\theta = \{T_{diff}, \alpha, \beta, \lambda, T_{pixel}, T_{cell}, T_{delay}\}$ are summarised in Table 3.5. In Figure 3.7 a number of index items computed for the Aircraft scene are shown. Note that for pragmatic purposes, index items are omitted if the frame has little activity, i.e. if $TotalD$ is low or $TotalCell(t) = 0$.

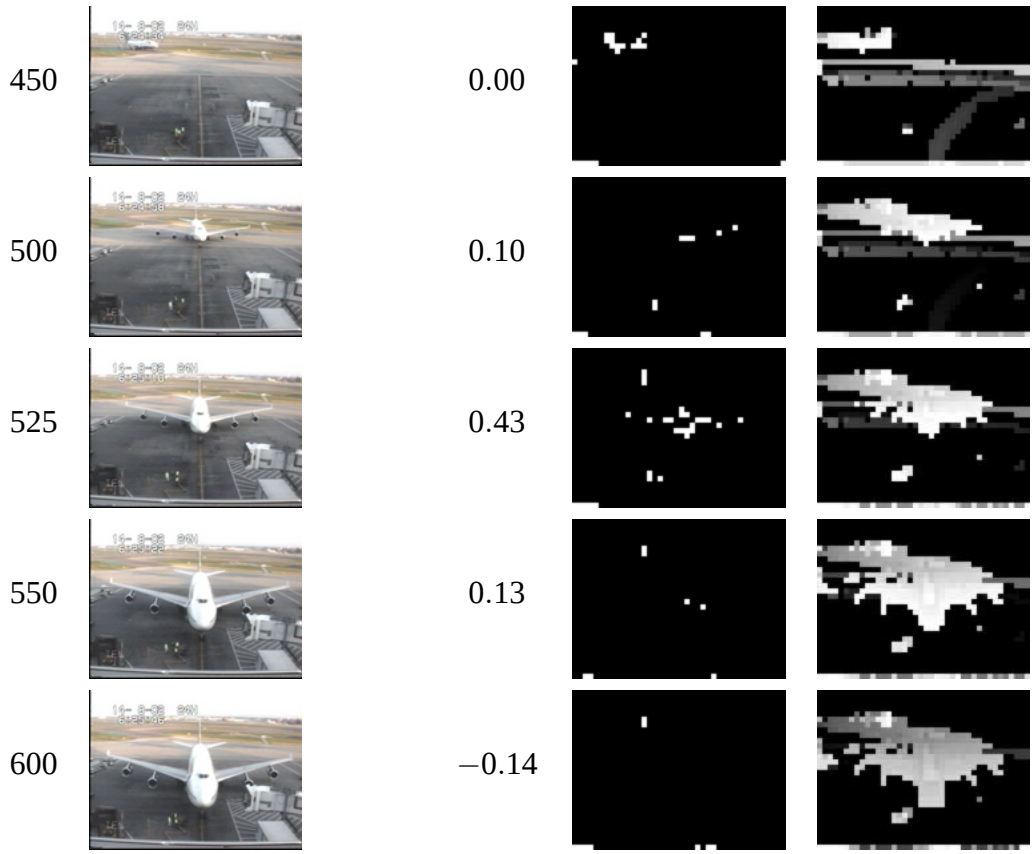


Figure 3.7: A demonstration of the frame index approach for the Aircraft1 scene. (left) The scene content, as seen in the frames, shows an aircraft approaching and docking with the docking station. (right) The index captures the level of activity in the scene, the current action, and the temporal context of action. The aircraft docking activity is clearly seen, however we also see the remnants of previous scene action allowing interpretation to be achieved in context.

3.3 Activity based video indexing

Although the cellular frame descriptors of Section 3.2.2 assume that each frame contains independent action, in reality the content of a scene action may span both spatial and temporal dimensions. The following Sections describe our approach for indexing a surveillance sequence at the activity level.

3.3.1 Significant activity segmentation over space and time

We consider a scene activity to consist of a set of spatio-temporally connected and overlapping active cells. To this end, and to reduce representational sparseness, a temporally extended connected-components algorithm is employed that detects, or *segments*, activities from the sequence. As input, the binary cellular grid description from Equation (3.11) computed for each frame is used. The result is a set of activity descriptors, each comprised of an activity identifier, start and end times, and profile. The segmentation algorithm is described below and detailed in Algorithm 3.1 on page 59:

1. A result set and current *memory* of activities are initialised as empty sets.

2. The active cells are computed for the next frame as Equation (3.11). The regions of connected active cells are found using the 4-way connected components algorithm (Gonzalez and Woods, 1992).

3. Each region of active cells is compared against the memory. The region is assigned to the activity in memory with the most spatial overlap in the previous time instant. If no overlap exists, a new activity is initialised using the region and added to memory. At this point the activity representation is a cellular grid where a value of 1 indicates that the cell is active and connected to the activity.

4. Each activity is analysed to see whether it has finished and, if so, whether it is significant enough to be added to the index. An activity that is alive in frame $(t-1)$ but not the current frame t is evaluated for spatial size using $T_{spatial}$ and temporal duration using $T_{temporal}$.

```
procedure ActivitySegmentation(Sequence)
Initialise Result
Initialise Memory
for Each frame do
    Compute the activity grid for the frame
    Regions = Compute the connected components for the grid
    for Each region r in Regions do
        Activity = best matching activity in Memory and Regions[r]
        if No matching activity found then
            Memory.AddActivity(new Activity(Regions[r]))
        endif
        else
            Activity.AddRegionToActivity(Regions[r])
        endif
    endfor
    for Each activity a in Memory do
        if a exists in frame (t-1) but not t then
            if Memory[a].Size > Threshold then
                Result.AddActivity(Current[a])
            endif
            Memory.RemoveActivity(a)
        endif
    endfor
endfor
return Result
```

Algorithm 3.1: An algorithm to extract (segment) spatio-temporally connected 'activities' from a set of frame cell grids. For each frame, the algorithm performs a standard spatio connected components algorithm to find the connected regions. Each region is assessed: if it matches the spatial position of an existing activity it is added to it; otherwise, it is assigned to a new activity. The detection of an activity is finished if no regions in the current frame are assigned to it - it is then assessed for size and added to the result index if sufficient.

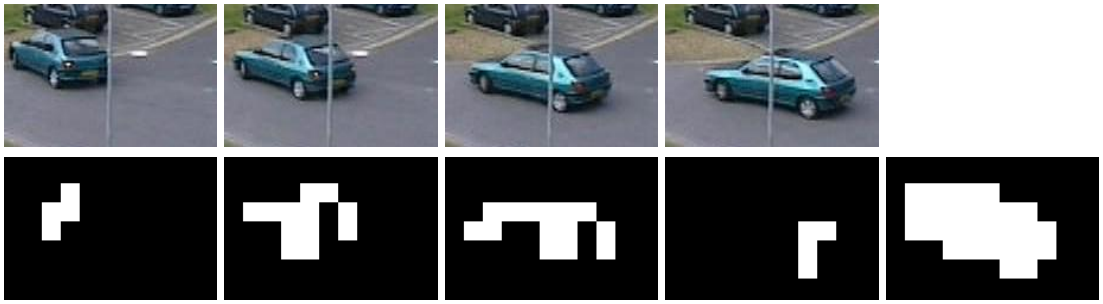


Figure 3.8: An illustration of the activity segmentation process. The action in several frames is estimated. Overlapping spatio-temporal action are assigned to an activity. The activity representation consists of a binary cell membership.

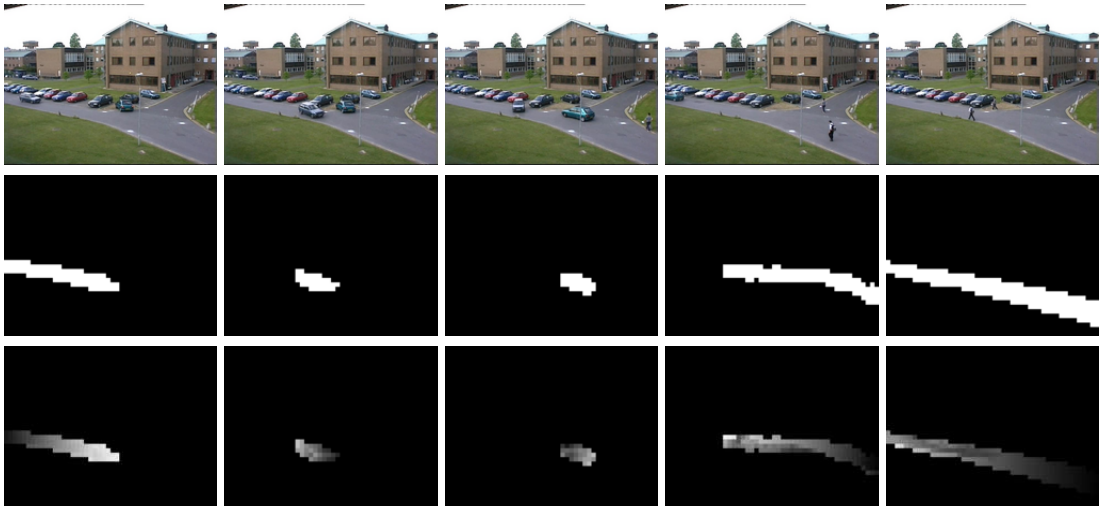
The result of the segmentation process is a set of activities, $\mathcal{A} = \{a_i : i \in [1, na]\}$, where na is the number of activities. Each is described by $a = \{id, start, end, shape\}$. The shape is a binary grid indicating which cells are a member of the activity. The activities found during the segmentation process correspond to large and important scene action. Figure 3.8 shows a number of frames and activity regions (spatially connected active cell positions) computed for a car parking activity in the Pets scene. The result activity shape is shown, and consists of a binary indication of the cell membership to the activity.

3.3.2 Spatio-temporal activity profiling

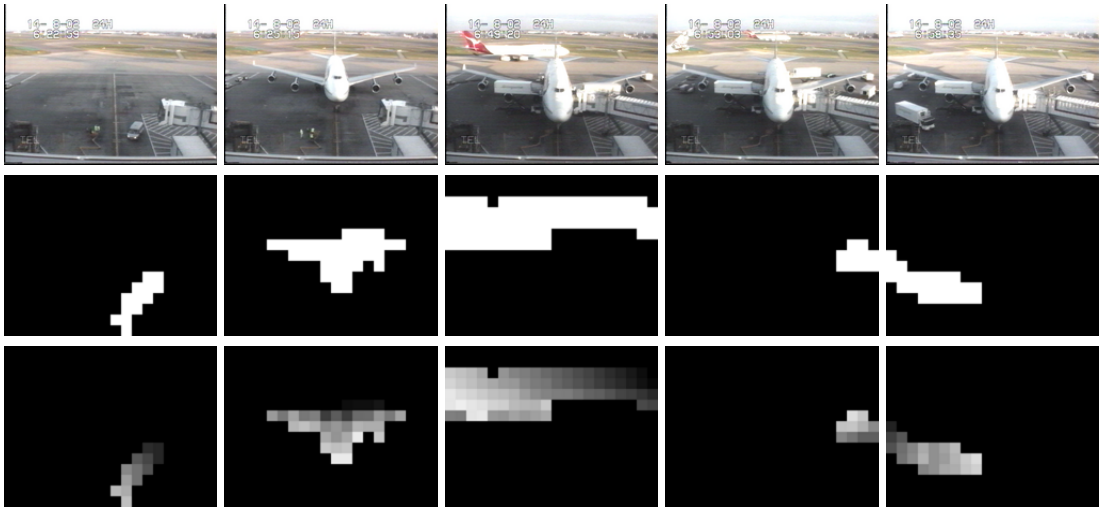
Following the segmentation of the activities from the sequence, an activity profiling method is used to capture the spatio-temporal character of each. We define the *shape* information, found during segmentation, using a Binary Shape Profile (BSP):

$$BSP_{\lambda,\alpha,\beta}(cx, cy, a) = \begin{cases} 1 & \text{If active and connected} \\ 0 & \text{Otherwise} \end{cases} \tag{3.14}$$

where a value of 1 indicates that the cell was active and spatio-temporally attached to the activity a. In other words, the cell was in a region of active cells attached to the activity at some point during its duration. The temporal size (duration) of the activity is computed as $Tsize = (end - start)$ and the spatial size as $Ssize = \max(\{|BSP| : \forall t \in a\})$ which is the largest activity size at any time instant.



Illustrative profiles computed for the Pets car-park scene.



Illustrative profiles computed for the Aircraft docking scene.

Figure 3.9: Illustration of the spatio-temporal activity profiling for two scenes. For each: (top) The mid-frame for the activity. (middle) The computed Binary Shape Profile. (bottom) The computed Temporal Order Profile. It is clear that the Binary Shape Profile captures the location and visual structure of action, and the Temporal Order Profile retains a time-scale invariant estimate of the temporal order. Dark cells generally occur towards the beginning and light cells towards the end.

The BSP stores information about the presence of activity, but not its temporal character. Therefore, in order to retain the temporal order within an activity (to capture the directionality) each cell's typical activation time within an activity context is computed as the Temporal Order Profile (TOP):

$$TOP_{\lambda,\alpha,\beta}(cx,cy,a) = \begin{cases} \frac{whenCellActive(cx,cy,a)}{(end-start)} & BSP(cx,cy,a) = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.15}$$

where $(end - start)$ is the temporal length of an activity, and *whereCellActive* is a function that returns the average cell activity time:

$$whenCellActive(cx,cy,a) = \frac{\sum_{t=start}^{t\leq end} \begin{cases} t-start & \text{if } Cell_{\lambda,\alpha,\beta}(cx,cy,t) = 1 \\ 0 & \text{otherwise} \end{cases}}{\sum_{t=start}^{t\leq end} Cell_{\lambda,\alpha,\beta}(cx,cy,t)} \tag{3.16}$$

where *whenCellActive* returns a value between $[0,(end-start)]$. A low value indicates that a cell was generally active towards the beginning of the activity whereas a high value indicates it was active towards the end (see Table 3.3). The TOP value for each cell is scaled into a range of $[0,1]$. Figure 3.9 shows examples of the spatio-temporal profiling for two outdoor scenes.

To summarise, we compute and stored an activity-based index $ActivityIndex(\mathcal{F},\theta)$ for a sequence $\mathcal{F}$ as illustrated in Table 3.4. This requires the computation of the cellular grid index of Section 3.2.2 upon which spatio-temporal segmentation is performed to find activities. Each is then profiled. The tuning parameters and thresholds $\theta = \{T_{diff}, \alpha, \beta, \lambda, T_{pixel}, T_{cell}, T_{spatial}, T_{temporal}\}$ are summarised in Table 3.5.

3.4 Similarity metrics for visual search

In Sections 3.2 and 3.3 frame-based and an activity-based indexing methods were developed. However, in order to perform search, a metric is required that is able to compute the perceptual similarity between index items. Our approach is to initially compute an inexpensive comparison of item spatial location and visual structure. Then, temporal

0	0	0	1	0	1	$(4+6)/2=5$
1	1	0	0	1	1	$(1+2+5+6)/4=3.5$
1	0	1	1	0	0	$(1+3+4)/3=2.67$
1	1	0	0	0	0	$(1+2)/2=1.5$
0	0	0	0	0	0	0
t=1	t=2	t=3	t=4	t=5	t=6	*whenCellActive*

Table 3.3: An illustration of the computation of the average cell activity time. We show the value of *Cell* from Equation (3.11) for the positions $[t-6,t-1]$ and the *whenCellActive* result from Equation (3.16). It can be seen that cells generally more active towards the end of the activity ($t=6$) produce a higher score.

a=1	a=2	...	a=m
[start, end]	[start, end]		[start, end]
[Ssize, Tsize]	[Ssize, Tsize]		[Ssize, Tsize]
BSP(a)	BSP(a)		BSP(a)
TOP(a)	TOP(a)		TOP(a)

Table 3.4: A summary of the activity index. For each activity segmented from sequence, an index item is computed using the starting and end positions, the spatial and temporal sizes, the Binary Shape Profile and the Temporal Order Profile.

T_{diff}	A threshold used during the computation of temporal change in Equation (3.8).
α	An accumulation factor used in Equation (3.9). Used to highlight pixels exhibiting sustained change.
β	A decay factor used in Equation (3.9). Used to de-highlight pixels that are no longer active.
λ	The grid cell-size first used in Equation (3.11).
T_{pixel}	A threshold used to determine whether a pixel is active or not according to its current level of sustained change. Used in Equation (3.11).
T_{cell}	A threshold used to determine whether a cell is active or not according to the number of active pixels that it contains. Used in Equation (3.11).

(a) Parameters used for the computation of action features.

T_{delay}	The maximum time delay since last cell action. Used during frame profiling in Equation (3.13).

(b) Parameters used for the computation of a frame index.

$T_{spatial}$	A spatial threshold used to reduce the number of activities retained during segmentation. Activities with a maximum spatial size less than the threshold are not retained.
$T_{temporal}$	A temporal threshold used to reduce the number of activities retained during segmentation. Activities with a temporal length less than the threshold are not retained.

(c) Parameters used for the computation of an activity index.

Table 3.5: A summary of the tuning parameters used during frame and activity indexing.

features are used to provide temporal context. To summarise, the following features are used to provide context in each index:

	Spatial context	**Temporal context**
FrameIndex	*Cell* from Equation (3.11)	*History* from Equation (3.13)
ActivityIndex	BSP from Equation (3.14)	TOP from Equation (3.15)

In the following Sections the terms P and Q are used to refer to the two items that are being compared. For mismatching binary cell positions, $P(cx,cy) \neq Q(cx,cy)$, a Z is used to refer to the grid with zero action cell. In other words, if there is a activity mismatch then $Z = P$ if $P(cx,cy) = 0$ and $Z = Q$ if $Q(cx,cy) = 0$.

3.4.1 Spatial similarity using transformation cost

The first step is establish a similarity using the geographic location of action and its visual structure. The result of $Similarity(P,Q)$ will be high for two items if they exhibit similar features. The following evidence based metric is proposed:

$$Similarity(P,Q) = \exp\left(-\frac{Negative(P,Q)}{Positive(P,Q)}\right) \tag{3.17}$$

to evaluate the ratio of *Negative* evidence (the two structures are not similar) to *Positive* evidence (the two structures are similar). The use of two forces, unlike using *Positive* alone, provides a framework in which an evaluation of the match between two structures is counter-balanced by the evaluation of non-match. This permits richer structures to be compared and also provides size invariance. Furthermore, we define:

$$Positive(P,Q) = \sum^{CX}\sum^{CY} |P(cx,cy) = Q(cx,cy) = 1| \tag{3.18}$$

$$Negative(P,Q) = \sum^{CX}\sum^{CY} |P(cx,cy) \neq Q(cx,cy)| \tag{3.19}$$

meaning semantically that *Positive* is the "number of matching active cells" and *Negative* evidence is the "number of non-match cells" (one active, one inactive). The similarity

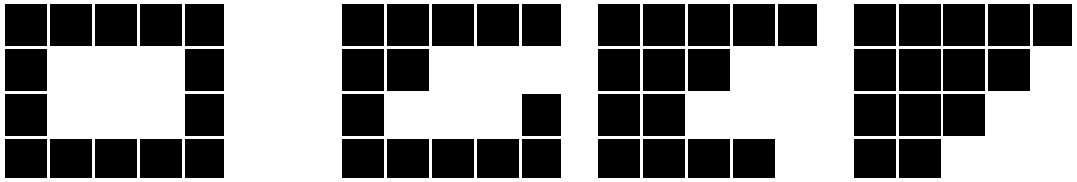


Figure 3.10: An illustration of the evidence based similarity metric. For a given Shape, we show three similar shapes along with the values of the basic positive evidence (matching active cells) and negative evidence (number of mis-matching cells). For example, for the first shape, there are five matching active cells and two cell mis-matches. As the second and third shapes become less similar, so this is reflected in the similarity score. However, it is clear that the approach is vulnerable to small spatial translations.

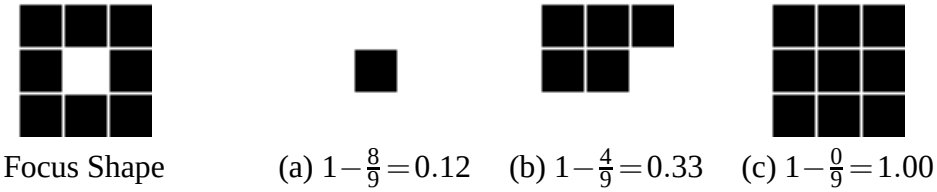


Figure 3.11: An illustration of the estimation of negative evidence accounting for small spatial translations. Considering only the middle cell in the Focus Shape, we show how the surrounding cell contents is used. In (a) all the surrounding cells are active, so the mis-match is explained by a local neighbourhood translation and the negative evidence is low. In (b) some surrounding cells are active. In (c) there are no surrounding cells active, so the mis-match cannot be explained and the negative evidence is high. The overall result is that the estimation of negative evidence is invariant to small spatial translations.

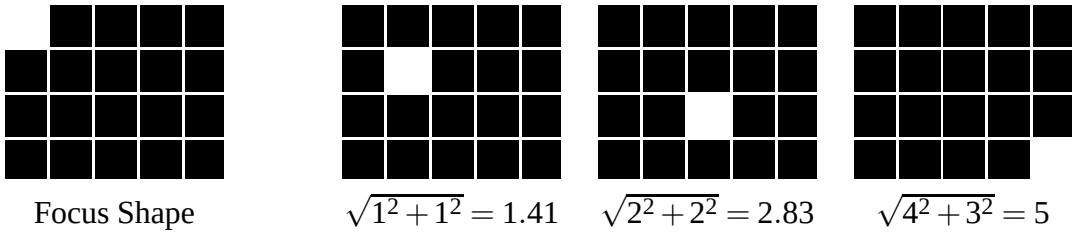


Figure 3.12: An illustration of the estimation of negative evidence account for large spatial translations. Considering the active cell in the given Shape, the negative increases if the corresponding closest active cell is distant.

result is therefore high if the number of matching cells is high and the number of mis-matching cells is low. See Figure 3.10. As can be seen, although criteria (3.18) and (3.19) are able to identify similar content, the binary nature of the *Negative* evidence is vulnerable to small spatial translations. We therefore propose using the level of localised activity when estimating the negative impact of each inactive mis-matching cell:

$$Negative(P,Q) = \sum^{CX}\sum^{CY} \begin{cases} Score(cx,cy) & \text{if } P(cx,cy) \neq Q(cx,cy) \\ 0 & \text{otherwise} \end{cases} \tag{3.20}$$

$$Score(cx,cy) = \frac{\left(\sum_{i=-1}^{i<=1}\sum_{j=-1}^{j<=1} Z(cx+i,cy+j)\right)}{9} \tag{3.21}$$

For each such mis-matching cell with zero activity, a score is computed using the level of action in its local neighbourhood. If those cells are inactive also, the negative score will remain high. However, if the cells are active, the mis-match can be explained as a local translation, and the negative score is lower. See Figure 3.11 for an illustration.

Alternatively, to consider a larger spatial context at a greater computational cost:

$$Score'(cx,cy) = \frac{\min\left(\{dist(cx,cy,i,j) : \forall i \in [1,CX]; \forall j \in [1,CY]; Z(i,j) = 1\}\right)}{dist(1,1,CX,CY)} \tag{3.22}$$

where *dist* is a ground distance between cell positions such as the Euclidean distance, $dist(x1,y1,x2,y2) = \sqrt{(x1-x2)^2 + (y1-y2)^2}$. The result is that the negative score for each mis-matching cell as the scaled distance to the nearest active cell. See Figure 3.12 for an illustration.

3.4.2 Exploiting temporal context

Although the spatial approach is able to identify similarly located and overlapping action, it does not yet exploit the temporal information stored in the index. Temporal information is now proposed to enhance the result. When comparing items in the FrameIndex, the cell's Negative evidence can be reduced if the cell was active in recent history. An

adjustment is therefore computed using:

$$HistoryAdjust(P,Q) = \sum^{CX}\sum^{CY} \begin{cases} \frac{History(cx,cy,Z)}{T_{delay}} & \text{if } P(cx,cy) \neq Q(cx,cy) \\ 0 & \text{otherwise} \end{cases} \tag{3.23}$$

where *History*, defined in Equation (3.13), is the time delay since the inactive cell was last active. The result is a boost to the similarity of frame index items in which there is a temporal translation of action.

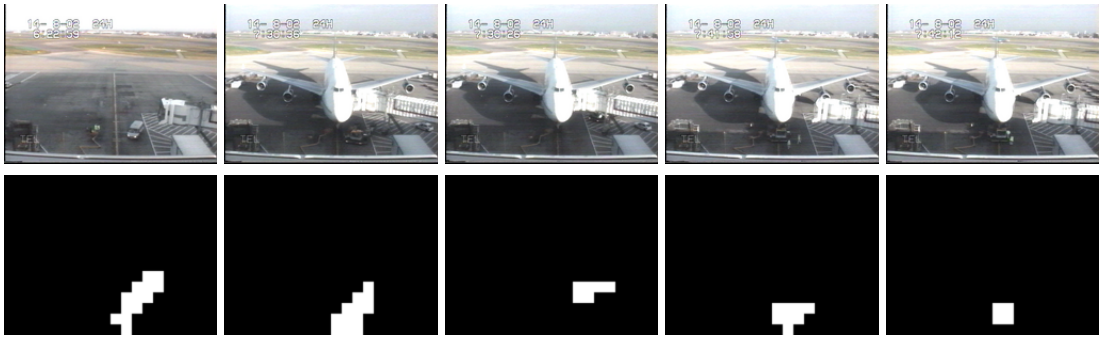
When comparing items in the ActivityIndex the temporal order is used:

$$TOPAdjust(P,Q) = \frac{TOPSim(P,Q) + TOPSim(Q,P)}{2} \tag{3.24}$$

$$TOPSim(P,Q) = \frac{\sum^{CX}\sum^{CY} \begin{cases} 1 - dist(cx,cy,i,j) & \text{if } P(cx,cy) > 0 \\ 0 & \text{otherwise} \end{cases}}{\sum^{CX}\sum^{CY} P(cx,cy) > 0} \tag{3.25}$$

where TOP is the Temporal Order Profile from Equation (3.15). The selection of (i, j) is the cell position in Q with the closest temporal order to the cell being examined in P, i.e. that minimises $|TOP(x,y,P) - TOP(i,j,Q)|$. The overall result is that items with similar temporal order produce a higher similarity.

To illustrate the similarity metrics, Figure 3.13 shows a query activity with the four most similar other activities found in an Aircraft docking sequence. The spatial context similarity finds activities that occur in a similar spatial location to the query. This is considered a good result considering the conceptual simplicity of the metric. For the temporal context similarity, those elements with a more similar directionality (top-right to bottom-left in the figure) are ranked higher. One advantage of decomposing the similarity estimation into two separate steps is that the temporal comparison can be restricted to items that are found to have similar spatial similarity. For example, in our system the spatial similarity is computed for all items and the temporal adjustment computed only for the most similar 25%. This can reduce the search time.



Rank 1 (Query) Rank 2 Rank 3 Rank 4 Rank 5

(a) Using the Binary Shape Profile.

Rank 1 (Query) Rank 2 Rank 3 Rank 4 Rank 5

(b) Using the Temporal Order Profile.

Figure 3.13: Demonstration of the similarity metrics. We show a query activity with the top four most similar activities found in the Aircraft docking scene. Note that the item used as the query is ranked first in all cases. (a) The comparison is made using the metric in Equation (3.17) using the spatial criteria given in Equations (3.18) and (3.19) in Section 3.4.1. For the activities we show the Binary Shape Profile. It can be seen that the similarity metric produces good results considering the spatial location of activity. (b) The comparison is made using the temporal adjustment given in Equation (3.24) in Section 3.4.2. For the activities we show the Temporal Order Profile. It can be seen that the similarity metric produces good matches considering the directionality of action.

3.5 Explanation based partitioning

Let us now consider how to use an index for performing temporal segmentation on a long, continuous sequence. In essence, we wish to discover points of temporal discontinuity in the sequence - at which the "future bears little resemblance to the past" - where these breaks provide key points for defining the structure of video content.

3.5.1 Localised temporal coherence

In traditional multimedia indexing, frames are automatically grouped into shots that are then grouped into scenes, thus providing a rich structure. When shots exist, they can be extracted and grouped using the correlation between the past and future (Kender and Yeo, 1998; Lin *et al.*, 2001). Reported approaches have proved successful for those sequences in which several sources are manually chopped and edited to simulate a story. However, surveillance video is fundamentally different. We are not reconstructing a sequence that was artificially constructed, rather a completely natural scene is being captured from which content is extracted without prior knowledge. Surveillance video is continuous, with no shot breaks, hence it is proposed to monitor the continuity of scene action in order to discover the points of discontinuity.

A generic measure of video coherence is computed at each index item:

$$Coherence(t) = \frac{\sum_{i=1}^{w/2} \text{median}\left(\forall_{j=1}^{w/2} Similarity(I_{t-i}, I_{t+j}) \right)}{w/2} \tag{3.26}$$

where w is the window size, I_i is the i^{th} index item, and *Similarity* is a metric. The similarity between the past $\{t-w/2,\ldots,t-1\}$ and the future $\{t+1,\ldots,t+w/2\}$ items is computed and modelled using the median. The median is used because it is known to be less sensitive to outliers (Weisstein, 2006b). The result values are in the range $[0,1]$ with a high value indicating sequence continuity.

When using FrameIndex, the computation of coherence is susceptible to low levels of scene action due to the linear population of the buffer. As all frames are used, including those with little or no action, the buffer will eventually fill with unimportant content.

This leads to bias in the level of coherence. We therefore employ an *expanding window* solution illustrated in Figure 3.14. The total scene action value is normalised using the approach in Appendix B and frames with a value > 0 are considered eligible to enter the buffer. The buffer window is expanded in either direction until it is filled. As can be seen in Figure 3.14 (c), the approach is more robust to periods of low action. Furthermore, the buffer is populated with a sub-sample of frames.

Note that when using the ActivityIndex the coherence for the activities in the sequence is discovered, however to obtain a corresponding temporal location t in the sequence we use the starting point of the activity.

3.5.2 Finding significant coherence minima

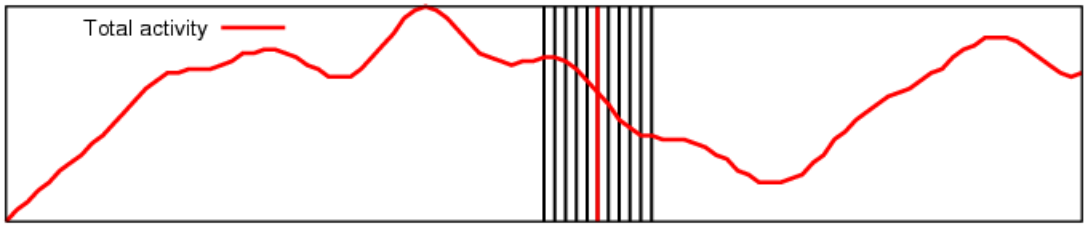
Once the coherence is computed, the minima are automatically found and marked as these are the breaks in continuity. An approach often used in literature is a *sliding window* method, for example (Sundaram and Chang, 2000), whereby a window is used to provide a local context during analysis. The following method is used:

- Candidate minima are discovered. A window of fixed size is moved across the graph of coherence values. Points at which the central value is equal to the lowest in the window are used.

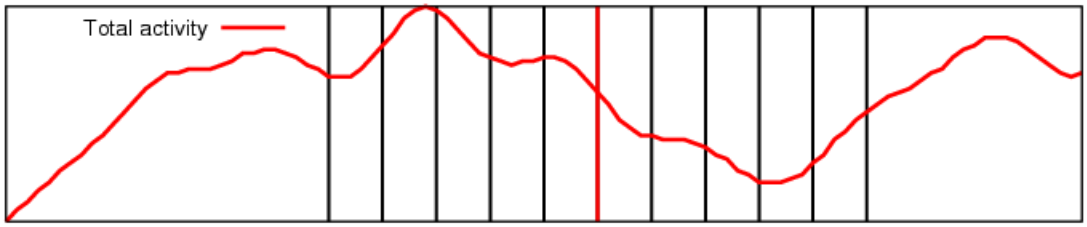
- Candidate minima that are too close are resolved. This situation occurs when multiple candidate minima have the same value, for example when the coherence has a wide minima. The candidate is retained that is the furthest from a candidate in the opposite direction.

- Candidate minima are pruned. The minima are ordered by coherence value and the desired number retained. This is selected according to the length of the sequence and the required granularity of content, i.e. how many scenes are to be detected.

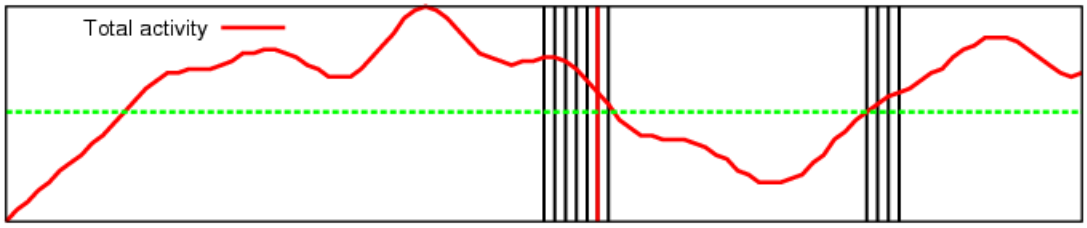
The process requires the number of desired breaks as input. We find that retaining the number of breaks according to sequence size is sufficient.



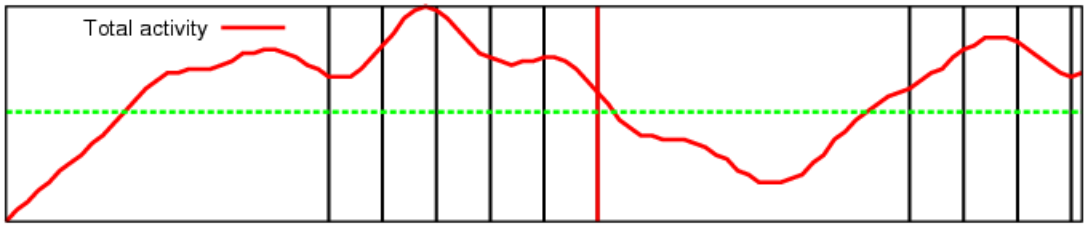
(a) A linearly populated buffer.



(b) A sub-sampled buffer.



(c) A thresholded buffer. Frames are only included if their normalised total activity is above a threshold.



(d) A sub-sampled thresholded buffer.

Figure 3.14: The frame buffering using a window size of 10. In (a) we see a normally populated buffer where 5 frames in either direction are used. We use vertical bars to indicate the frames included in the buffer. In (b) we show that by sub-sampling the frames included in the buffer we can achieve much wider window coverage. This corresponds to an increase in the temporal context during the buffer computation. In (c) we show the thresholded approach. Frames are only included if they contain a level of action. This removes the low-action frames that occur frequently in long surveillance sequences. In (d) we show a combination of the sub-sampled thresholded case. It is able to provide a wide and meaningful temporal context.

3.6 Experiments

In this Chapter approaches have been presented for the estimation of action features from a sequence (Sections 3.2.1-3.2.2), the computation of a frame-based index (Section 3.2.3), the computation of an activity-based index (Section 3.3) , and the computation of a coherence based temporal segmentation (Sections 3.4-3.5). We now demonstrate the indexing and temporal segmentation approach.

A sequence and its manual segmentation

We captured a long surveillance sequence that observes the Aircraft docking station scene shown in Figure 3.1. The sequence is roughly 1.5 hours of footage, sampled and digitised at 2Hz, resulting in $11,000$ frames of size 320×240. Let us call this sequence Aircraft1. Upon manual inspection, the following eight salient scenes were identified:

frames 0-400	empty dock
400-600	aircraft arrival
600-2,700	passengers dis-embark and unloading
2,700-5,700	plane re-stocked
5,700-7,500	period of inactivity
7,500-8,750	final loading
8,750-9,500	engines examined
9,500-11,000	aircraft departure

In the sequence, a plane arrives at the dock, is restocked, loaded, examined, and then departs. An illustrative frame from each of these manually identified scenes is shown in Figure 3.15. Unsurprisingly, it is extremely difficult to identify the scene content from such a static frame presentation even though this approach is a commonly employed for video summarisation.

Note that a manual segmentation is subjective, not guaranteed to be correct, and not guaranteed to be consistent with segmentations produced by other observers. To demonstrate, four manual segmentations were collected from observers that were not familiar with the scene. In Figure 3.16, the positions of the breaks in the ground truth

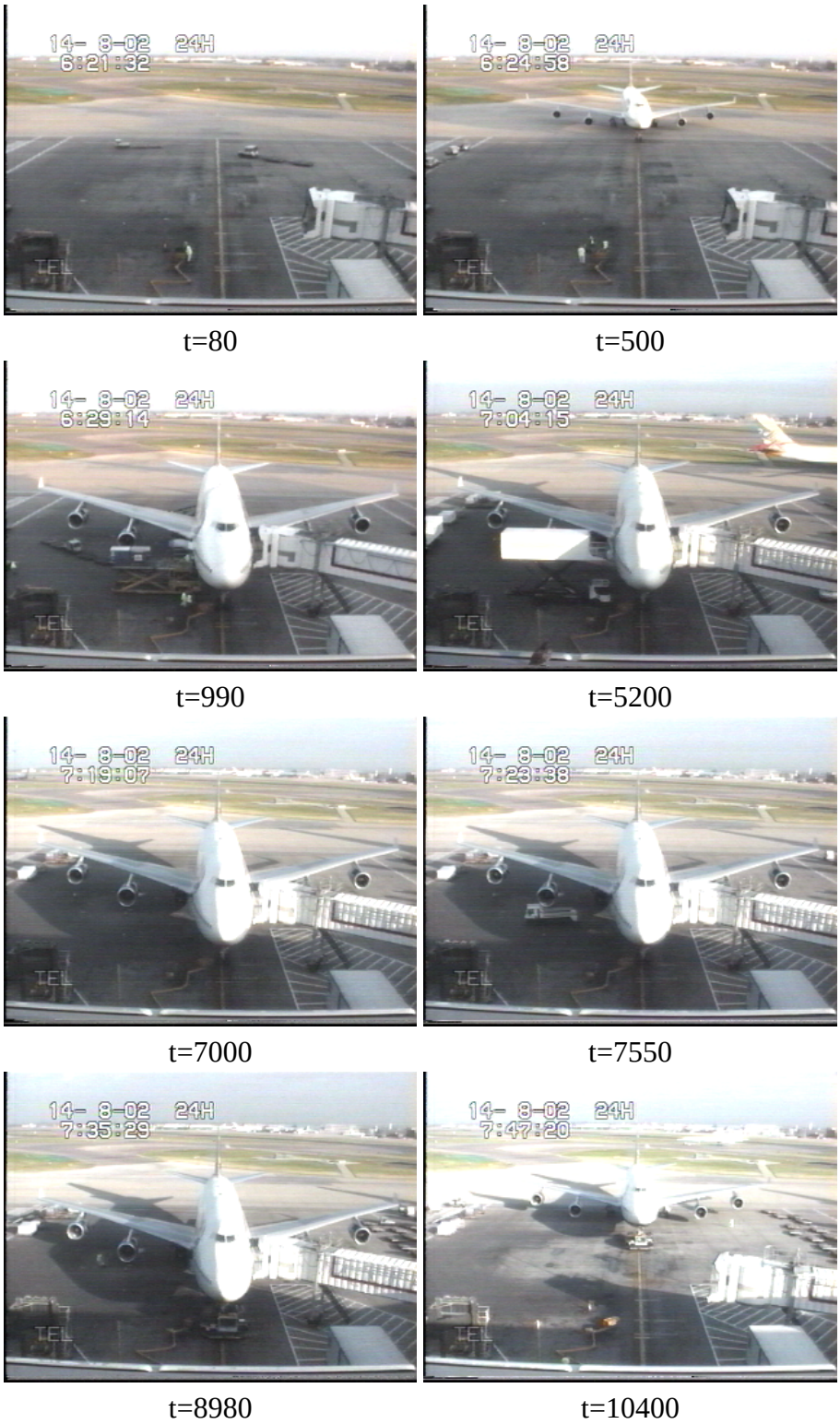


Figure 3.15: An illustrative frame from each of the eight manually identified scenes. It is difficult to determine the scene content.

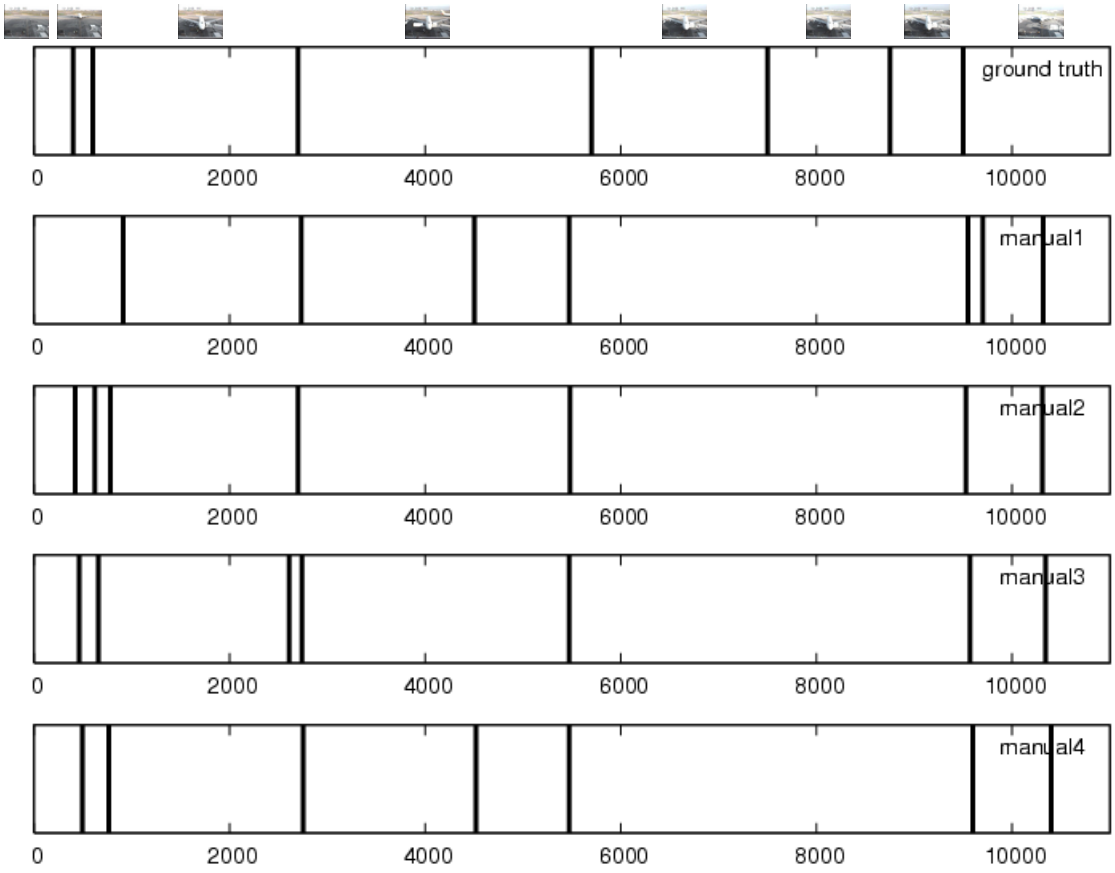


Figure 3.16: An illustration of the manual segmentations for the Aircraft1 sequence. (top row) We show the ground truth, as explained on page 74, along with frame position indicators that correspond to the eight frames shown in Figure 3.15. (second+ rows) We show four manual segmentations produced by different observers with no previous surveillance or segmentation experience. Each manual segmentation is semantically meaningful as each break point required justification during the experiment. It can be seen that segmentations are consistent with each other and can be used for evaluation.

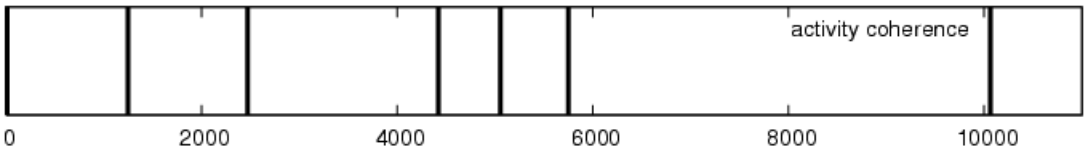


Figure 3.17: The activity coherence result for the Aircraft1 sequence. In comparison to the manual breaks of Figure 3.16, we can see that the break points are well positioned. In particular, the activity coherence breaks at the approximate frame positions $t \in \{1100, 2400, 5000, 10100\}$ are located at manual breaks.

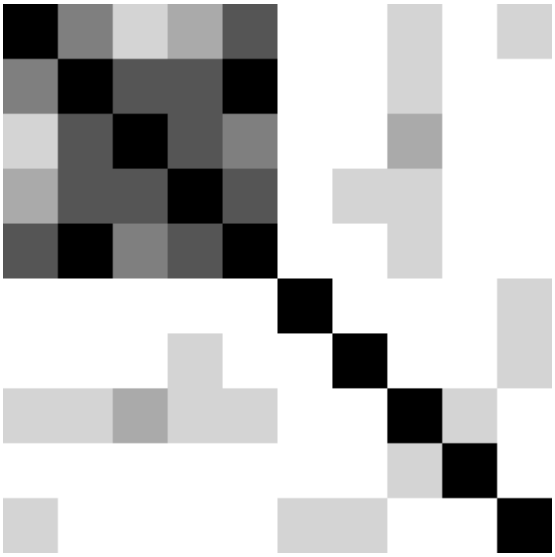


Figure 3.18: A similarity matrix that shows the intra-set and inter-set similarities between five manual and five random segmentations. Positions $1-5$ are the manual segmentations and $6-10$ the random segmentations. Each position shows the similarity between the items, where black is similar and white dis-similar. Each item is identical to itself as seen on the identity diagonal (top-left to bottom right). It can be seen that the manual segmentations are similar to each other (as shown by the top-left quarter of the matrix having dark cells), the random segmentations are not similar to each other, and also the intra-set similarity is low.

and four manual segmentations are shown (for the 11,000 length sequence). It can be seen that the segmentations have some similar and dissimilar tendencies.

In order to examine the consistency of the five manually produced temporal segmentations (ground truth plus four manual alternatives mentioned above), five random segmentations were generated and then a similarity matrix computed between the ten. See Figure 3.18. Each random segmentation was generated using random numbers between $[1,n]$ for each break, with the additional constraint that breaks should be more than 20 positions distant from each other. During the similarity computation a matching distance of $T_{within} = 50$ was used in Equation (3.4). It can be seen that the manual intra-set similarities are high (as indicated by the dark cells at the top-left of the matrix) whereas the random intra-set similarities are low and the inter-set are also low. This illustrates that the manual segmentations are consistent and can be used for evaluation.

Computation of action features using generic parameters

The first task was to compute the action features, namely the measure of reliable temporal change in Section 3.2.1 and the grid based frame descriptor from Section 3.2.2. This requires a selection of tuning parameters summarised in Table 3.5 (a). To find a suitable and generic set of parameters, a number of preliminary experiments were conducted using sequences showing the Hand waving, Pets carpark, and Aircraft docking scenes (see Figure 3.1 on page 48), in which the parameter values were varied. It was found that, with the exception of the cellsize λ, the value of each parameter does not effect the computation time and so a generic choice is possible. The following settings provide a good result in all scenes:

$$T_{diff} = 5$$
$$\alpha = \beta = \frac{50}{255}$$
$$T_{pixel} = 50$$
$$T_{cell} = 15\%$$

Also, changing the parameters by small values produce very similar results meaning that the approach is not overly parameter sensitive.

The choice of cellsize λ determines the computation time and available storage capacity. A generic choice of $\lambda = 16$ was found to capture sufficient detail in the different scenes, however it can be changed according to the anticipated size of moving object content within the scene and the computational and storage limitations.

Frame indexing and temporal segmentation

For the Aircraft1 sequence, a FrameIndex was computed using the parameters described above and $T_{delay} = 255$. This value was chosen as it provides a sufficiently sized history (255 frames at 2Hz equals approximately 2 minutes of action) and also corresponds to 255 grey-levels stored and displayed using an image raster format. The frame similarity metric used was defined in Section 3.4 by Equations (3.17), (3.18), (3.21) and (3.23).

In order to perform partitioning, frame coherence was computed using a window size

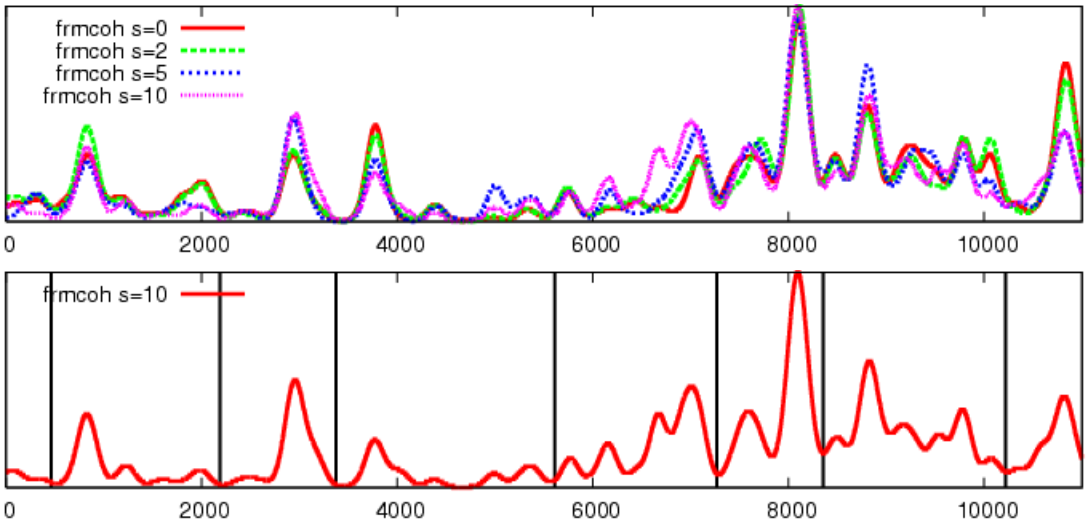


Figure 3.19: The frame coherence result for the Aircraft1 sequence. (top) The frame coherence using subsample size of $[1, 2, 5, 10]$. The sub-sampled version provides a similar result at much reduced computational cost. (bottom) The frame coherence result with subsample size of 10 with the 7 detected minima giving a corresponding 8 scenes - as per the manual segmentation experiment.

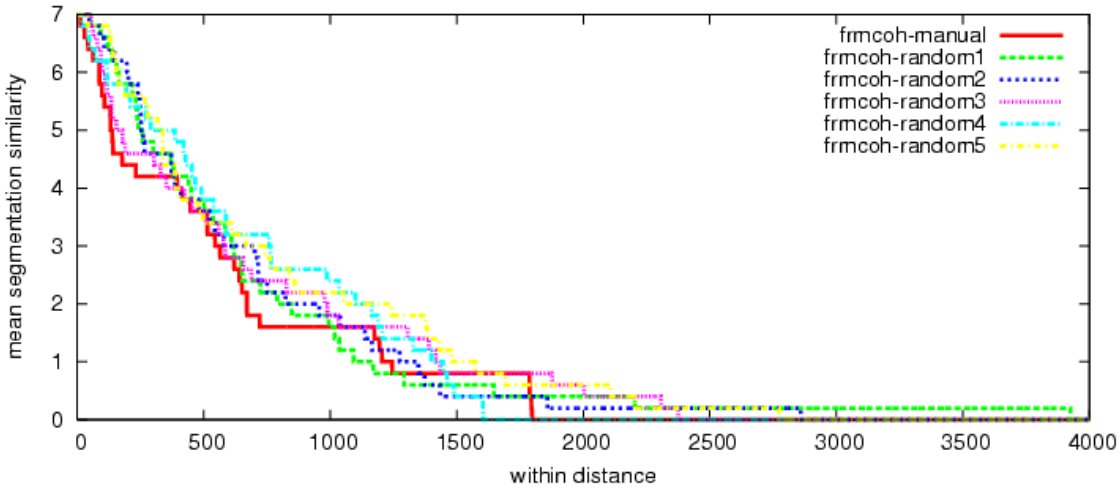


Figure 3.20: A comparison of the frame coherence segmentation against one set of manual segmentations and five sets of random segmentations. (Each set containing five segmentations). The graph shows the mean similarity between the frame coherence segmentation and each set, using different values for the threshold T_{within} of Equation (3.4). It can be seen that the frmcoh-manual comparison is not discernible from the frmcoh-random comparisons.

of $w = 100$ in Equation (3.26) and subsample sizes of $[1, 2, 5, 10]$ as described in Section 3.5.1. It was found that increasing the size of the window further was not possible as the number of frame similarities needed increases exponentially with window size. This is because an increase in size of $+1$ results in number of similarities needed by $w/2$, as the new frame in the "past" must be compared against every frame in the "future" according to Equation (3.26). The median must also be computed on a larger set requiring more computational expense (Weisstein, 2006b).

The result computed for the different subsample sizes is shown in Figure 3.19 (top). It can be seen that sub-sampling can be used to produce a similar result at a fraction of the computational cost. The detected breaks are shown in Figure 3.19 (bottom) as vertical bars. It can be seen that the frame coherence approach is able to partition the sequence, but is sensitive to level of activity. The impact of a small window size is that the temporal context considered in the coherence computation is small. This leads to temporally localised discontinuity detection rather than long-term content change detection.

In order to evaluate the temporal segmentation automatically produced using the frame coherence, it is compared against the manual and random sets using a varying matching distance in the segmentation similarity metric of Equation (3.4). The segmentation was compared against one manual set and five random sets, each consisting of five segmentations. In Figure 3.20 the similarities are shown: the x-axis corresponding to the increase in the size of the matching distance; the y-axis is the the mean similarity between the frame coherence segmentation and the test set. A desirable result is one in which the similarity falls quickly, as this corresponds to more breaks being aligned within a smaller match distance. In the Figure, it can be seen that the manual result is not easily discernible from the five random results, although it descends more quickly than most. This suggests that the frame coherence segmentation is as similar to the manual segmentations as those randomly generated.

Activity indexing and temporal segmentation

For the Aircraft1 sequence, the ActivityIndex was computed using the parameters used previously for the frame indexing along with activity thresholds:

$$T_{spatial} = 3$$

$$T_{temporal} = 10$$

used in the spatio-temporal activity cropping of Section 3.3.1. We found through initial experimentation that these were sufficient for retaining a sufficient number of activities in the index. The activity segmentation is able to find and segment the scene action efficient and effectively. For the Aircraft1 sequence a total of 363 activities were discovered, a number of which are displayed in Figure 3.21.

Owing to the compactness of our index representation, a small window size can be used during the computation of the coherence. The activity similarity metric used was defined in Section 3.4 by Equations (3.17), (3.18), (3.22), (3.24) and (3.25). Figure 3.22 shows the activity coherence produced using the window sizes, 4, 12 and 20. It can be seen that the larger window sizes produce a more consistent score, because as more activities are added to the buffer's past and future elements the past-future comparison is more robust to short-term action changes. By observation, we found that a window size of 12 performed sufficiently well for a number of different scenes, so is used for experiments. To illustrate the coherence computation, in Figure 3.23 the maxima and minima are marked and five activity Temporal Order Profiles (TOP) from the buffer past/future are shown for these positions. For the maxima, it can be seen that the activities have similar spatial locations, visual structure and directionality. For the minima, the past action bears little resemblance to the future, leading to the low video coherence score.

We used the activity coherence to compute eight scenes using seven minima. In Figure 3.24 the coherence for the 363 items is shown along with the detected minima. The activity coherence and minima for $11,000$ frame positions is also shown, using the coherence value for an activity as the value for all the frames for that activity. In Figure 3.17 on page 76, the positions of these breaks are shown (on the same page as

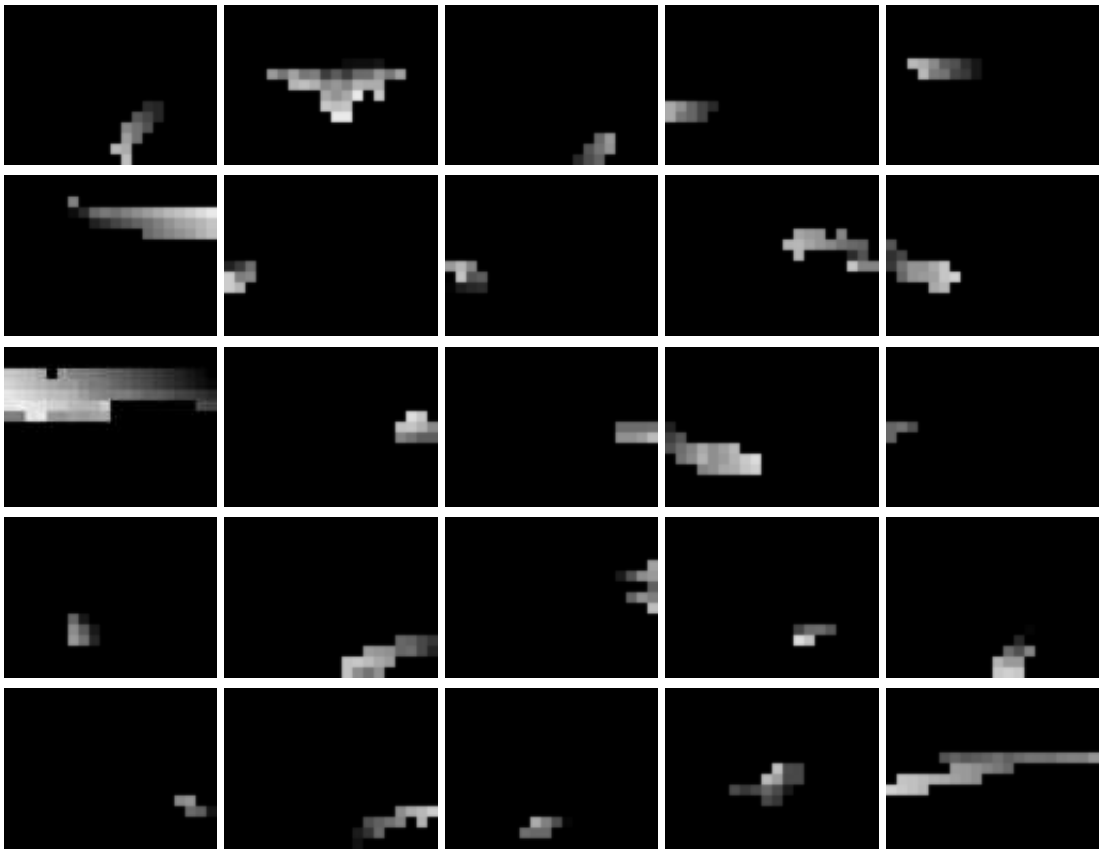


Figure 3.21: A number of activity Temporal Order Profiles computed for the Aircraft1 sequence. Different activity contents - arrival, loading, unloading - are clearly observed.

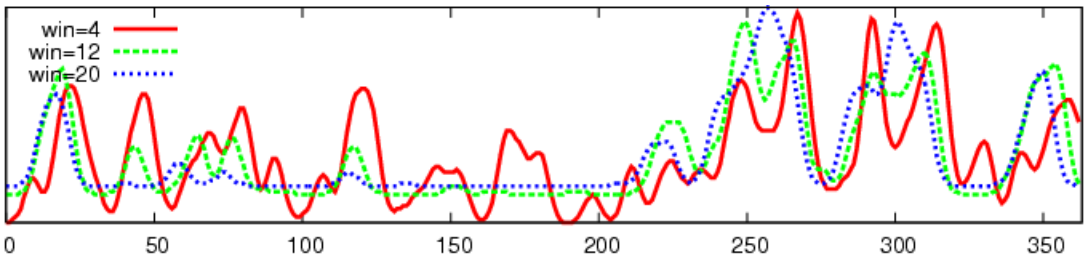
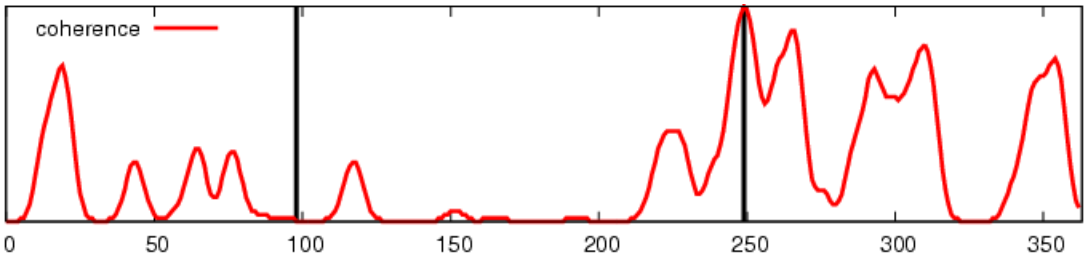


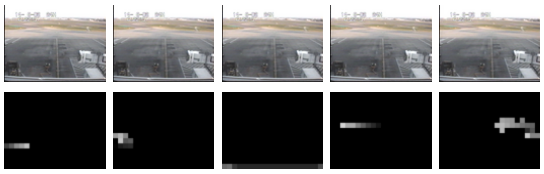
Figure 3.22: The activity coherence result produced with three window sizes. We can see that the result using the wider window is smoother, as more similar items in the past-future comparison are found.



(a) The activity coherence. An illustrative maxima and minima are marked.

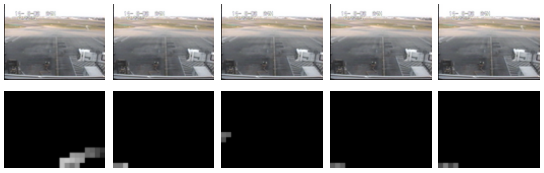
89 91 93 95 97

(b) Example of the past at the minima



99 101 103 105 107

(c) Example of the future at the minima



240 242 244 246 248

(d) Example of the past at the maxima

250 252 254 256 258

(e) Example of the future at the maxima

Figure 3.23: An illustration of the computation of coherence. (a) The activity coherence produced for the Aircraft1 sequence. We mark illustrative minima and maxima positions for which we show the content of the activity buffer. Comparing (b) and (c), it can be seen that the past contains little resemblance to the future. This results in the coherence minima. Comparing (d) and (e), it can be seen that the past contains similarities with the future. This results in a high level of coherence.

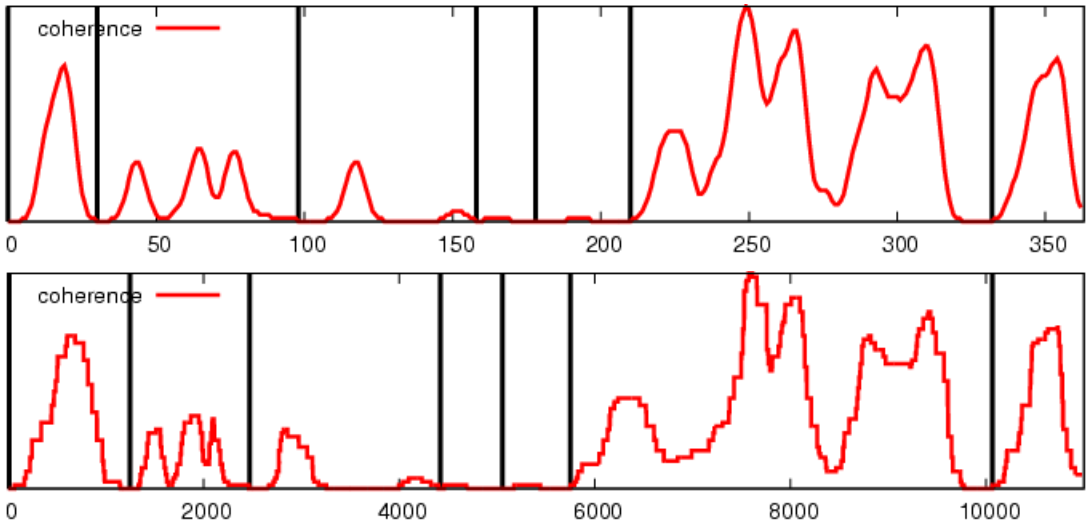


Figure 3.24: The activity coherence as applied to frame positions. (top) The activity coherence computed in which the detected minima are marked. (bottom) The activity coherence is displayed according to sequence frames. All frames within an activity are given the coherence score - leading to a mini-plateau effect - providing the real temporal/frame positions of the breaks.

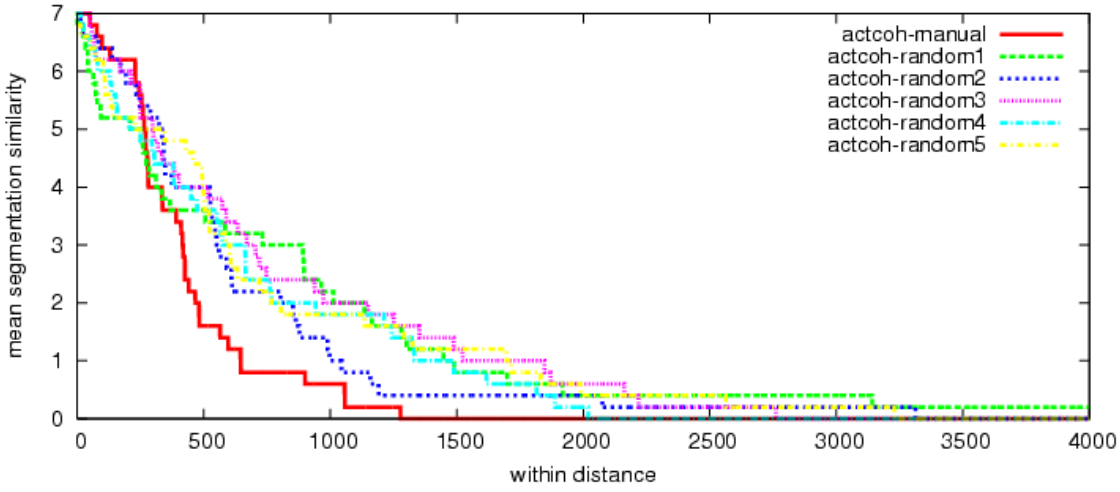


Figure 3.25: A comparison of the activity coherence segmentation against one set of manual segmentations and five sets of random segmentations (compared to Figure 3.20). It can be seen that the actcoh-manual comparison produces a much steeper drop than the actcoh-random comparisons, meaning that the activity coherence segmentation is much more aligned to the manual segmentations than those randomly generated.

the manual break positions for visual comparison purposes). It can be seen that the breaks reasonably lineup with the manual breaks, meaning that the activity coherence approach is producing a temporal segmentation comparable to those manually produced. Upon further analysis, it was found that the mis-placed breaks tend to be due to an over-segmentation that occurs during inactive periods.

Similar to the frame coherence result, the activity coherence temporal segmentation was compared to the manual segmentations and five random sets of segmentations (as computed for the frame coherence and shown in Figure 3.20). The result is shown in Figure 3.25. It can be seen that the activity coherence segmentation is much more aligned to the manual segmentations than the random segmentations. This suggests that the activity coherence approach produces a segmentation similar to those produced manually.

3.7 Discussion

In this Chapter, the important problems of forming a pre-attentive sequence index and performing temporal segmentation of surveillance video were addressed. Using spatio-temporally smoothed temporal difference - the Sustained temporal change - a grid-based frame descriptor was computed to explicitly represent the spatial location and history of scene action. This frame information was used to form an index, however it was found to be large because of representational sparseness. We therefore performed spatio-temporal activity segmentation to extract significant regions of connected, active cells. Each activity was profiled, to capture its spatial structure and temporal-order information, and a compact index formed.

In order to search the indices, a transformation-cost based similarity metric was proposed. The metric estimates the similarity in spatial location and visual structure of action using estimates of the negative and positive evidence, and then the action asynchronicity and temporal order to reduce dis-similarity estimates. In effect, the metric is able to explain the inconsistency between two representations using spatial and temporal factors. It was found that the metric was able to find, using the query-by-example paradigm, similar activities by spatial location and directionality. However, the represen-

tation and similarity approach are primarily dependent on the spatial location of scene action. Whilst useful in most situations, it can be argued that visual appearance of action is more desirable. We will develop such an approach in the next Chapter.

To perform automatic video partitioning, a sliding window based approach was proposed for comparing the past action against the future. At each position the coherence is computed and the coherence minima, points at which there is little resemblance between the future and the past, are used as breaks in a partitioning. When using the FrameIndex, the approach was found to be computationally expensive as the number of similarities needed increases with window size. This leads to a small window size with insufficient temporal context. To overcome this obstacle, an expanding window mechanism was proposed. However, we conclude that a frame-based approach is not optimal. When using the ActivityIndex, the coherence computation was found to be much more efficient and reliable, because the index has retained only the important scene content and is more compact. This approach is computationally undemanding, operates without the use of colour information, requires no training and can detect primitive scene changes in surveillance video. The approach result was compared against a manual segmentation, and it performed favourably when compared to randomly generated segmentations. However, we note that it is difficult to perform a quantitative evaluation because of the inherent subjectivity in manual assessments.

Chapter 4

Iconic video indexing

In this Chapter, a video representation is developed that can be used to perform video indexing of unstructured surveillance video. Rather than a geographic location approach, as in Chapter 3, we aim to extract and use the local visual appearance of action. This provides a translation invariant mechanism for scene action comparison facilitating visual search with no assumptions on scene content. This is critical if a diversity of surveillance videos with varying scene content is to be analysed.

Motion can be observed as orientations over time and can be analysed using orientation sensitive motion-sensors (Chomat *et al.*, 2000) or through static orientation analysis of spatio-temporal image-slices (Ngo *et al.*, 2002). However, reported approaches operate on the scene visual appearance rather than action leading to representational redundancy. Following the success of the temporal-change based approaches in literature (Bobick and Davis, 2001; Gong and Xiang, 2003), wavelet-based orientation filters are employed to analyse the appearance of local action. Commonly occurring action types, called icons, are used to form an *Iconic visual vocabulary* used for frame description. Visual search is achieved using known histogram matching metrics. Temporal segmentation is achieved by monitoring the cumulative appearance of the Iconic terms during a sequence. Finally, a video summarisation is produced using the discriminant action in the scene.

4.1 The need for discriminant visual context

In order to perform successful visual search on a wide variety of different surveillance search spaces, it is critical that:

- the index representation is uncommitted, and
- the search mechanism is efficient and generic.

By *uncommitted* we refer to the system having little advance knowledge, so very few assumptions are made about the expected scene contents. This is because search tasks are often retrospective, and search criteria are often developed after the data has been captured and archived. Additionally, the large scale of video data-banks demand a compact representation and efficient metric.

An important visual characteristic to be captured is a *discriminant visual context*, i.e. a local region of activity able to successfully establish content similarity. A good discriminant visual context is similar in nature to a good search term used in text retrieval systems. For example, the word “visual” returns approximately 141 million documents when used in Google, whereas the word “the” returns approximately 3.4 billion documents. This demonstrates that, although “visual” returns plentiful candidate documents, it is more discriminant than “the” and is thus a better search context.

4.2 A wavelet-based sequence descriptor

4.2.1 Wavelet analysis of temporal change

During indexing, we are interested in the appearance of *regions of temporal change* rather than individual pixels as they contain more contextual information, are more semantically pertinent to visual search tasks, and will produce a more compact representation more suitable for long sequences. Firstly, a robust temporal change $D_{\alpha,\beta}(x,y,t)$ is computed as described in Section 3.2.1. The temporal change space is divided into a regular, static grid of square cells of equal size. The cell-size λ determines the granularity of the descriptors and is chosen according to the scene layout and index size requirements.

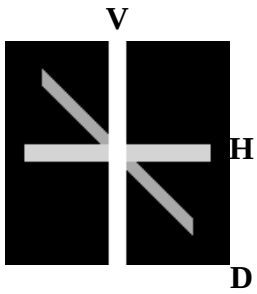


Figure 4.1: The visualisation of a cell. The lines represent to the Vertical, Horizontal and Diagonal energies. The amount of energy is depicted by line length and colour.

Moments of Haar wavelet coefficients have been shown to be effective for texture analysis and provide a good compromise between computational complexity and effectiveness (Unser, 1995). Comparable approaches such as Gaussian derivatives and Gabor wavelets offer little improvement in result (Oren *et al.*, 1997). The action content of each cell is described using the Haar basis function:

$$\Psi(x) = \begin{cases} 1 & 0 \leq x < 0.5 \\ -1 & 0.5 \leq x < 1 \\ 0 & otherwise \end{cases} \tag{4.1}$$

applied using a wavelet transform $\phi_i^j(x) = \phi(2^j x - i)$, where x input is translated using the number of scales j and position i. The result of the wavelet is a division of input according to four sub-bands: low-high (LH), high-low (HL), high-high (HH) and low-low (LL). Using the result, a 3D feature vector, $\psi(x,y,t) = \langle \psi^1 \, \psi^2 \, \psi^3 \rangle$, is formed using the mean of the coefficients in the LH, HL and HH bands. The LL band information is not used as the band result is equivalent to a sub-sample of the original data. The level of overall energy is given by the sum of the coefficients, $|\psi|$.

When applied to the Sustained temporal change of Equation (3.9) for a cell, ψ provides an estimate of the *action energy* in the vertical, horizontal and diagonal directions. This gives an estimate of a cell's localised visual structure and action directionality. Henceforth, the cell action is visualised using the icon explained in Figure 4.1. Figure 4.2 shows the coefficients produced for the scene extract in Figure 3.3 on page 51.

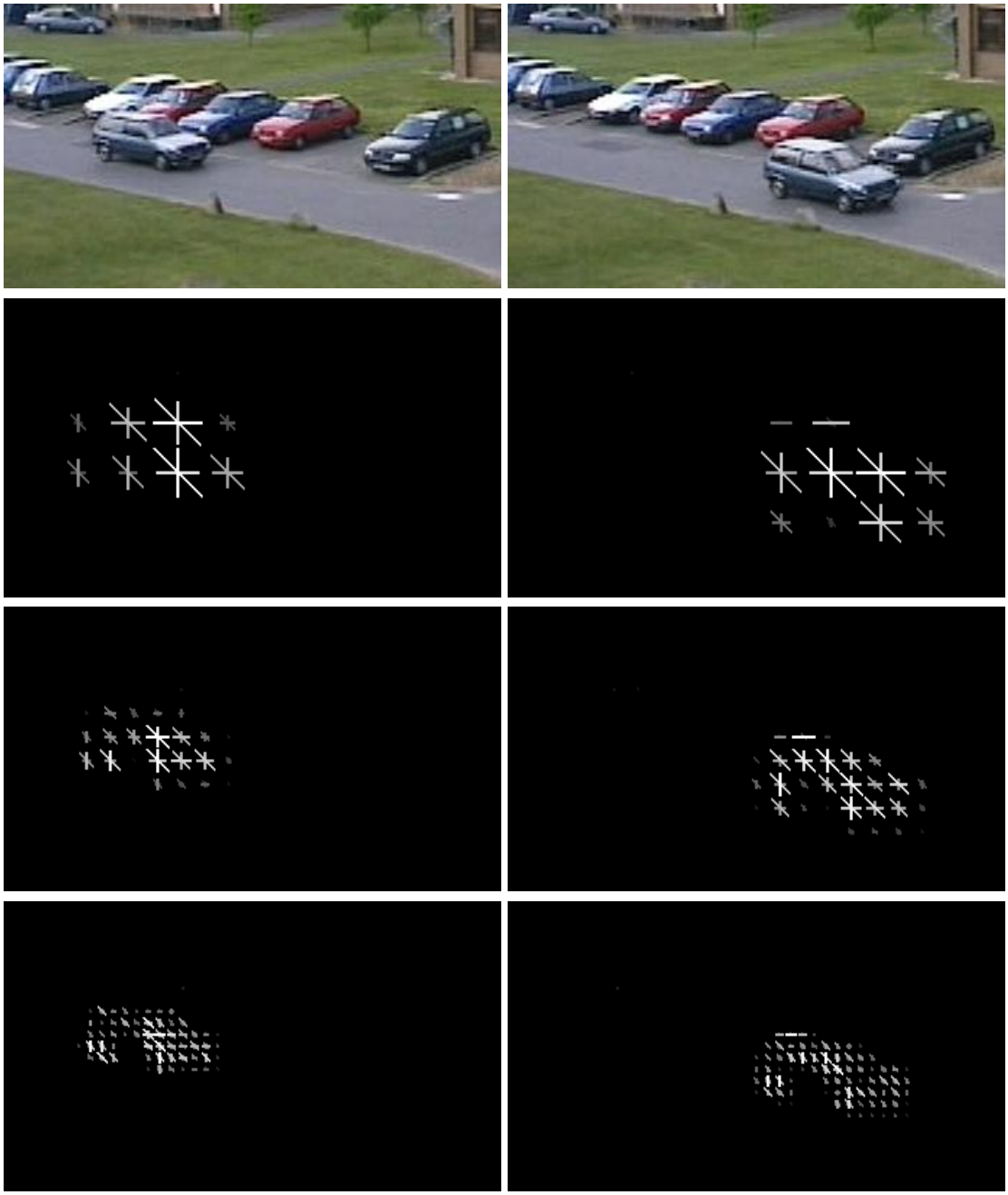


Figure 4.2: The Haar coefficients computed for an extract from the Pets scene from Figure 3.3 on page 51. We show the iconic visualisation (see Figure 4.1) for three cell-sizes: top=32, middle=16, bottom=8. The Sustained temporal change computed for an image is divided into a grid of equally sized cells and the Haar wavelet analysis is performed on each. The computed coefficients correspond to the amount of energy in the vertical, horizontal and diagonal directions. It can be seen that an object motion can be holistically described using the set of coefficients computed for it.

4.2.2 An iconic visual vocabulary

To provide a certain degree of perceptual robustness to any matching process, a compact, invariant scene descriptor is now formed. A 3D feature space is computed for a sequence using the cell descriptors. This space is then clustered using mixture-model based clustering, to find κ common patterns of local visual appearance. The centroids are each called an *Iconic term*, θ, because it is an important element of visual context within the scene. The set of iconic terms form an *Iconic visual vocabulary* used for scene description, $\Theta = [\theta_1, \theta_2, \ldots, \theta_\kappa]$. The choice of vocabulary size κ is critical - our approach is in Section 4.3. Each frame is described using a *Frame occurrence histogram* (FOH) with elements representing the number of Iconic term occurrence:

$$FOH_t = \left\{ f_t^k : \forall k \in [1, \kappa] \right\} \tag{4.2}$$

$$f_t^k = \sum^{CX}\sum^{CY} \left(\min^{j<\kappa} \left(dist\left(\psi(cx, cy, t), \theta_j \right) \right) = k \right) \tag{4.3}$$

where t is the frame being described, $\forall k$ refers to all Iconic terms, CX and CY are the number of cells in each dimensions, and $dist$ is the Euclidean distance between the cell feature vector ψ and the Iconic term θ. The result of Equation (4.3) is the number of occurrences of icon k in frame t. The representation captures a translation invariant and perceptually robust description of what is happening in the scene. See Figure 4.4.

An advantage of using a histogram based frame representation, is that a standard histogram similarity metric can be used for frame comparison and retrieval. The histogram intersection measure of Swain and Ballard (1991) is adopted:

$$1 - \frac{\sum_i^\kappa \min(P_i, Q_i)}{\sum_i^\kappa q_i} \tag{4.4}$$

where $P = \{p_i : i \in [1, \kappa]\}$ and $Q = \{q_i : i \in [1, \kappa]\}$ are histograms of the same size that are being compared. The metric is widely used for matching of colour histograms in image retrieval systems (Lew, 2001) owing to computational efficiency and low susceptibility to the curse of dimensionality (described in detail on page 31).

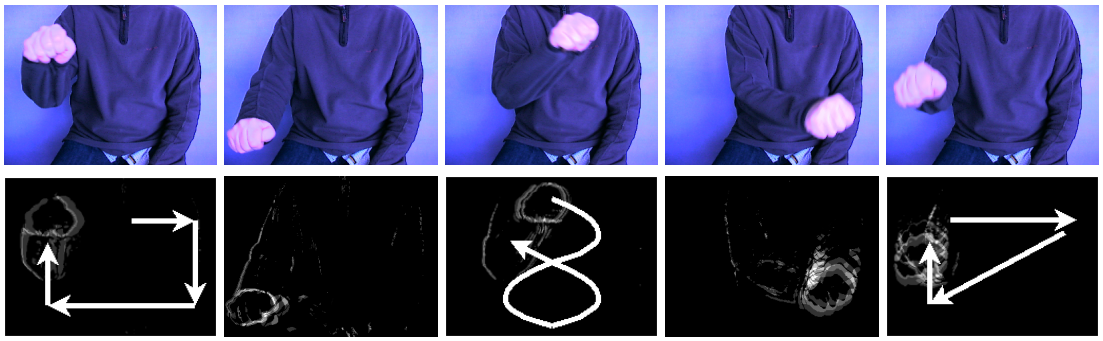


Figure 4.3: The content of the Waving1 sequence. The sequence was generated that contains three scenes for illustrative purposes. The hand signals begin with square movements, then figure eight movements, then triangle movements.

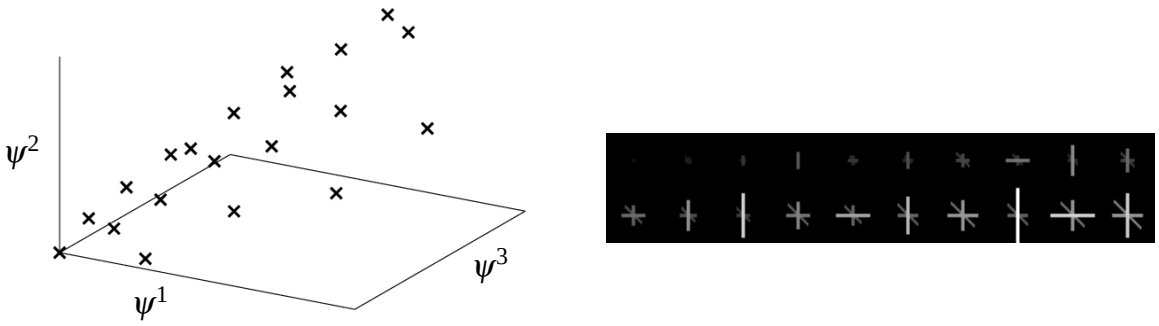


Figure 4.4: The computation of the Iconic visual vocabulary. (left) The 3D Haar coefficient space computed for the Waving1 sequence is clustered using the k-means algorithm. Here we show the resultant centroids. (right) Each centroid corresponds to a position in the coefficient space. Each is an *Iconic term* in the vocabulary and is henceforth used for describing the sequence content.

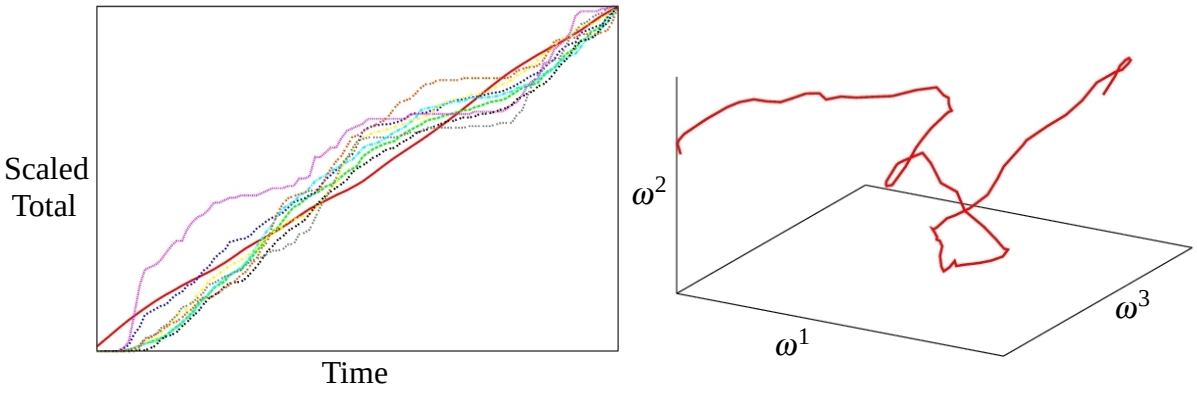


Figure 4.5: An analysis of the iconic term occurrence during the sequence. (left) A cumulative histogram showing the occurrence of the Iconic terms throughout the sequence. Variations in the histogram correspond to various content that stimulate different coefficients. (right) A projection using the first three principal components. We call this projection the *Video scene trajectory*. It captures the important thematic change from the cumulative histogram, and hence the varying scene content.

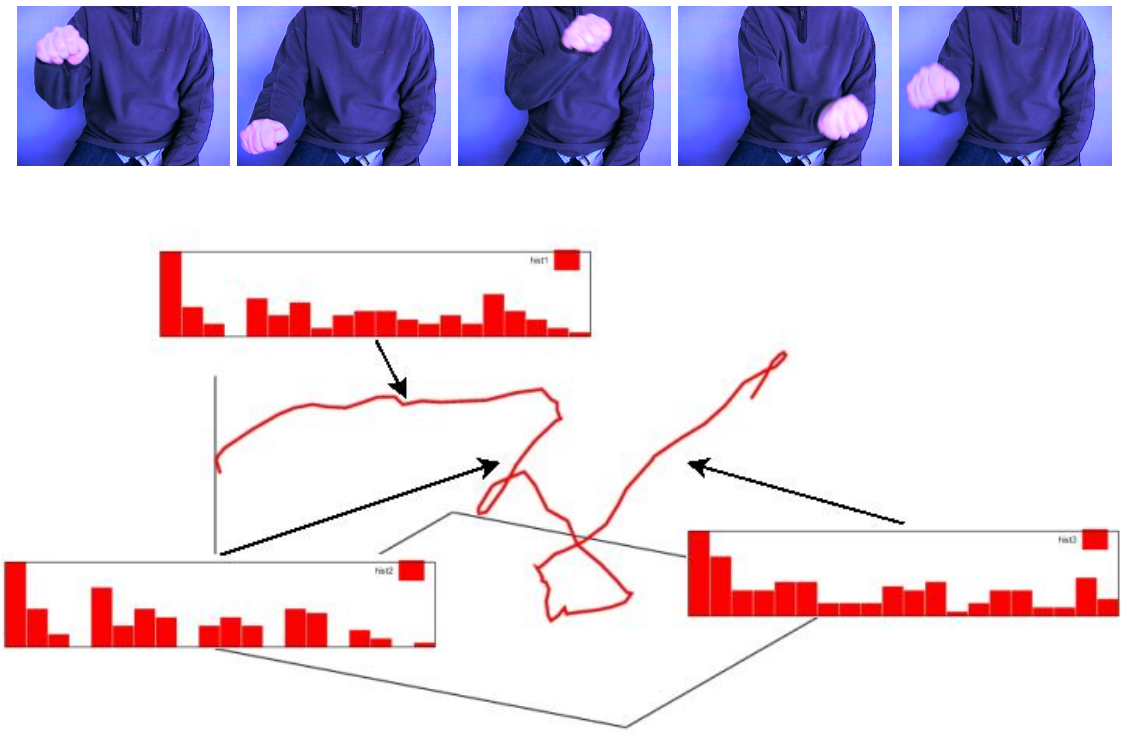


Figure 4.6: The Video scene trajectory produced for the Waving1 sequence shown. The sequence content is constructed with three scenes - squares, eights, triangles. The varying content produces different frame histograms; captured in the variations in cumulative totals shown in Figure 4.5; the themes of which are captured in the trajectory.

4.2.3 A cumulative analysis of iconic appearance

The key to our approach is that a scene can be defined as having a *similar profile of filter responses* throughout its period. Over a long period of time, similar content will stimulate patterns in the iconic occurrence histogram of Equation (4.3). As content changes, different patterns will occur. The approach for detecting content change by a cumulative analysis of the iconic appearance histograms is now described.

A continuous representation that captures long term content and thematic change is formed. A scaled cumulative histogram is used:

$$Cumul(t,k) = \frac{Cumul(t-1,k) + f_t^k}{Cumul(n,k)} \tag{4.5}$$

where $Cumul(t,k)$ gives the cumulative occurrence total at frame t for Iconic term k. $Cumul(0,k) = 0$ for all classes. f_t^k is the occurrence level of term k in the frame, as in Equation (4.3). It is clear that Cumul increases monotonically with t for each class. The result is scaled between $[0,1]$ for each point using the value at the last frame n. See Figure 4.5. One problem with the approach is that some filter responses are common to all scenes and are not helpful for content change detection. The use of a scaled cumulative histogram is able to reduce the effect of this noise.

4.2.4 A video scene trajectory

The variations in the scaled cumulative histogram capture different long-term frame content changes, however the dimensionality of the frame descriptors κ is high. This results in difficulty the detection of important change. Also, many of the Iconic terms are unimportant with respect to the content, for example a term that captures background noise. Therefore, in order to provide focus, the principal subspace of the scaled cumulative histogram is computed using Principal Components Analysis, a well known technique for data analysis and dimensionality reduction (Press *et al.*, 1992). The result eigenvectors with the corresponding highest eigenvalues preserve the important cumulative effects.

The scaled cumulative histogram is projected into its subspace using the first ω eigenvectors. When $\omega = 3$, the result is called a *Video scene trajectory* (VST) for the se-

quence, because it is easily visualised as a three dimensional trajectory. Figure 4.5 shows an Iconic term scaled cumulative histogram and a VST computed for a sequence with three clearly distinctive periods of content. Figure 4.6 show the trajectory along with an example of the frame histograms that occur at the different phases. It can be seen that the frame histograms shown are distinctive. The result VST provides a structure that clearly corresponds to the changing content in the sequence.

4.3 Model order selection using entropy

The choice of vocabulary size κ must *maximise the potential discrimination ability* of the representation as search tasks must quickly discriminate amongst a large dataset. For each candidate vocabulary, the quality of the contained terms is analysed and those more suitable for visual search are retained. Similar to text retrieval where a few words are found to possess statistical power for searching, we aim to discover the powerful visual elements in the scene. Firstly, a *Term occurrence histogram* for each term is built by concatenating the frame term occurrence f^i for each frame:

$$O(\theta_k) = \left\{ o_t^k : \forall t \in [1,n] \right\} \tag{4.6}$$

where $o_t^k = f_t^k$. o is used to distinguish the term occurrence histogram (the distribution of the term occurrence throughout the sequence) of Equation (4.6) and the frame occurrence histogram (containing the number of term occurrences in that frame) of Equation (4.2). The histogram provides information about the distribution of the term occurrence in the sequence. To allow comparison between low-frequency and high-frequency terms, it is scaled using $(\sum^{t<n} o_t^k) = 1$. The total term occurrence is $|O(\theta_k)| = Cumul(n,k)$.

If a term is popular, i.e. the $|O(\theta_i)|$ is high compared to other terms, it does not necessarily mean that the term is good for searching. The opposite is usually the case. For example, in text retrieval the words [“and”,“the”,“if”] occur commonly but are clearly unsuitable for searching as they lack context or any actual meaning. Similarly, unpopular words [“bivariate”,“condition”,“giraffe”] are unsuitable because they are too infrequent.

A prominent solution in text indexing is the removal of the highest/lowest frequency terms (Rijsbergen, 1979). In our approach, the most frequent and most infrequent terms from a candidate vocabulary being assessed are removed.

A measure of a term's discriminant ability is desired. In text indexing, a classic approach is the *inverse document frequency*, computed as:

$$idf(\theta_k) = \log\left(\frac{n}{n_k}\right) \tag{4.7}$$

where n is the total number of documents and $0 \leq n_k \leq n$ is the number of documents in which term k occurs. Unfortunately, in our case the computation of a binary icon-frame membership n_k is not practical as the frames are spatially large and likely to contain all the terms. Another possibility is the computation of the normalised fourth order moment of the term occurrence histogram, the kurtosis, as it provides a measure of the distribution peakedness (Weisstein, 2006a). However, this approach is not possible as the temporal order of occurrence is not significant.

Our approach is to use the homogeneity of the term occurrence distribution computed using Entropy:

$$E(\theta_k) = -\sum_{t=1}^{t<n} o_t^k \log_2(o_t^k) \tag{4.8}$$

A term that occurs evenly throughout the sequence (e.g. background) is bad for searching will produce a high entropy score. A term for which the distribution is peaked at certain positions is good for searching and produces a low score. See Figure 4.8.

A good term for searching is one that occurs frequently, is discriminant, and is of a significant size. To this end, the normalised values of $|O'(\theta_i)|$, $E'(\theta_i)$ and $|\psi'(\theta_i)|$ are computed using the approach in Appendix B. The term *quality* is then estimated using:

$$Q(\theta_i) = |O'(\theta_i)| + |\psi'(\theta_i)| - 2E'(\theta_i) \tag{4.9}$$

Figure 4.9 illustrates four terms, their term occurrence histograms from Equation (4.6), and a textual description of their properties. It can be seen that the term quality provides a compromise between frequently occurring terms and those that provide maximum dis-

criminance. During indexing, a vocabulary size κ is chosen that maximises the mean term quality, by max $\frac{\sum^{k\in\kappa} Q(\theta_k)}{\kappa}$.

4.4 Partitioning by trajectory approximation

For a sequence that is represented using a Video scene trajectory, we now wish to partition the sequence into meaningful segments to solve the temporal segmentation problem. Considering that the trajectory is smooth when the action content in the scene is stable, the key trajectory alterations are detected and these positions used as the breaks. To this end, a linear piecewise approximation of the trajectory is generated that retains the key vertices using the Discrete Curve Evolution (DCE) algorithm (DeMenthon *et al.*, 2000):

1. The *relevance* of each vertex on the trajectory is computed using:

$$rel(t) = dist(t-1,t) + dist(t,t+1) - dist(t-1,t+1) \tag{4.10}$$

   where *dist* is the Euclidean distance. The relevance score *rel* is low if the point can be removed from the trajectory without significantly increasing the reconstruction error.
2. The vertex with the least relevance is removed.
3. Repeat until the required number of vertices remain.

The final retained vertices are the points deemed most necessary for reconstruction, and are used as the break points in the temporal segmentation. The algorithm is found to be conceptually intuitive, efficient and effective. However, it does operate on the whole trajectory at once. If an online process was required, alternative algorithms could be exploited (Keogh *et al.*, 2001). We show a trajectory and its approximation in Figure 4.6. It is clear that the approximation retains the shape and character of the original, using a few key points. Considering the trajectory is continuous when no change is occurring, these points can be used as break points.

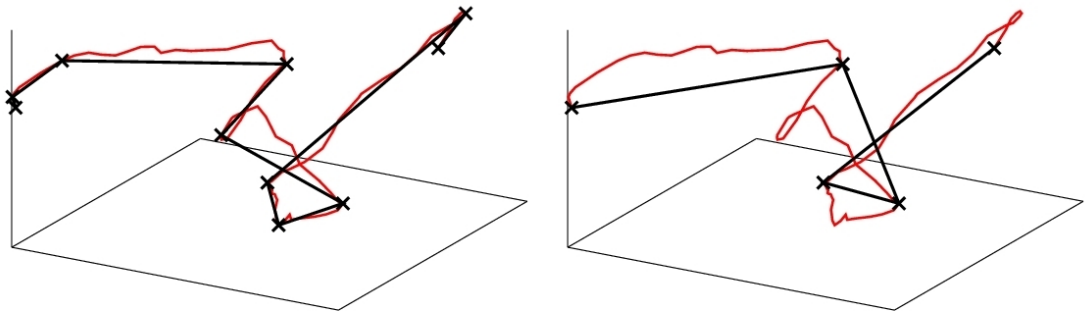


Figure 4.7: A trajectory and it's approximation using 10 points and 5 points. It can be seen that the shape and character of the trajectory is maintained. The points found can be used to divide the sequence into phases of similar directional content.

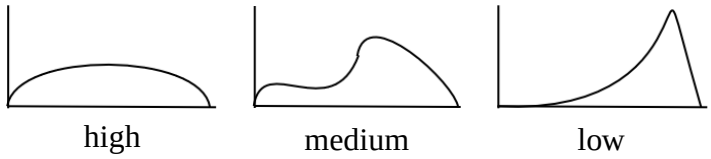


Figure 4.8: Three histograms corresponding to high, medium and low entropy. High entropy indicates that the term occurrence is evenly distributed and is bad for searching. A low entropy indicates that the term occurrence is peaked and is thus discriminate.

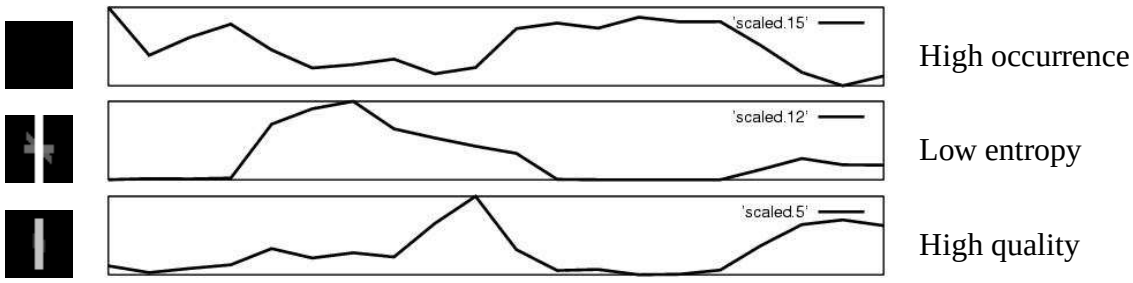


Figure 4.9: An illustration of the term evaluation. We show four iconic terms, their scaled term occurrence histograms, and a textual description. The *high occurrence* term corresponds to background activity. The *low entropy* term corresponds to a highly discriminate term - it matches against half the sequence and does not match at all against the other half. This is good for searching as the non-matching half can be quickly discounted. The *high quality* term provides a balance between the two desirable properties.

4.5 Visualisation using discriminant action

Once a sequence has been segmented, a video summary that illustrates the scene content is required in order to solve the video visualisation problem. Commonly, the first, last, mid or n^{th} frame or frames are used as the segment summary. However, such a static representation is found to be ineffective because it does not describe "what is happening" in the scene. Therefore, in our approach a Segment summary frame is constructed for each segment that emphasises the particular action that occurred within it. For a segment the set of active pixels are used that best represent its action. Each pixel is evaluated using the criteria:

1. How active the pixel is in the segment.

2. How good the pixel is for describing a segment considering the sequence. In other words, the *discrimination ability* of the pixel.

The motivation is that we wish to use the pixels that are most discriminative, i.e. are best for describing the unique content in the segment. This is similar to the *term frequency, inverse document frequency* term weighting strategy in text indexing (Rijsbergen, 1979).

For a sequence, each pixel position is evaluated for its discrimination ability:

$$DiscriminationAbility(x,y) = \log\left(\frac{n}{\sum_{t=1}^{t<n} D(x,y,t) > T_{act}}\right) \tag{4.11}$$

where D is the Sustained temporal change from Equation (3.9), T_{act} is a threshold to determine a level of significant pixel activity. A high value indicates that the pixel is rarely active. To compute the set of representative active pixels for segment s, each pixel is evaluated as:

$$SegmentActivePixel(x,y,s) = \frac{\sum^{t\in s}(D(x,y,t) > T_{act})}{\max\ SegmentActivePixel(\forall x, \forall y, s)} \tag{4.12}$$

where $t \in s$ corresponds to the set of frames in the segment. The result is scaled to the range $[0,1]$ where a high value indicates that the pixel was highly active in the segment.

Each pixel is scored using

$$Score(x,y) = DiscriminationAbility(x,y)SegmentActivePixel(x,y,s) \tag{4.13}$$

and the top τ% of pixels with the highest scores are Top discriminative active pixels (TDAP). These provide indication of the most interesting segment action. A sequence summary is formed by computing a Segment summary frame (SSF) for each segment in the sequence. The SSF is computed using the TDAP and the first frame from the segment to provide a visual context:

$$SSF(x,y,s) = \begin{cases} 255 & \text{if } (x,y) \in TDAP(s,\tau) \\ \gamma F(x,y,s) & \text{otherwise} \end{cases} \tag{4.14}$$

where $F(x,y,s)$ is the starting frame of activity s and γ is a scalar between $[0,1]$ used for reducing the frame emphasis. $\gamma = 0.5$ is found to work sufficiently well. Figure 4.19 shows, for two outdoor scenes, the *DiscriminationAbility*. Additionally, the total level activity for a segment - i.e. *TotalD* from Equation (3.10) - and the computed Segment summary frame are shown for a segment detected at that scene. It can be seen that the SSFs successfully summarise the most pertinent action in each segment.

4.6 Experiments

In this Chapter approaches for video indexing, temporal segmentation and summarisation have been presented, using the action features computed from a sequence. The video index was built by computing the Haar wavelet coefficients for the scene action (Section 4.2.1), forming an Iconic visual vocabulary for scene description (Section 4.2.2), that was then used to form a Video scene trajectory (Sections 4.2.3-4.2.4) that captures the changing scene content. Temporal segmentation was achieved by a process of trajectory discretisation (Section 4.4). A video summary was formed using the segmentation and the discriminant scene action (Section 4.5).

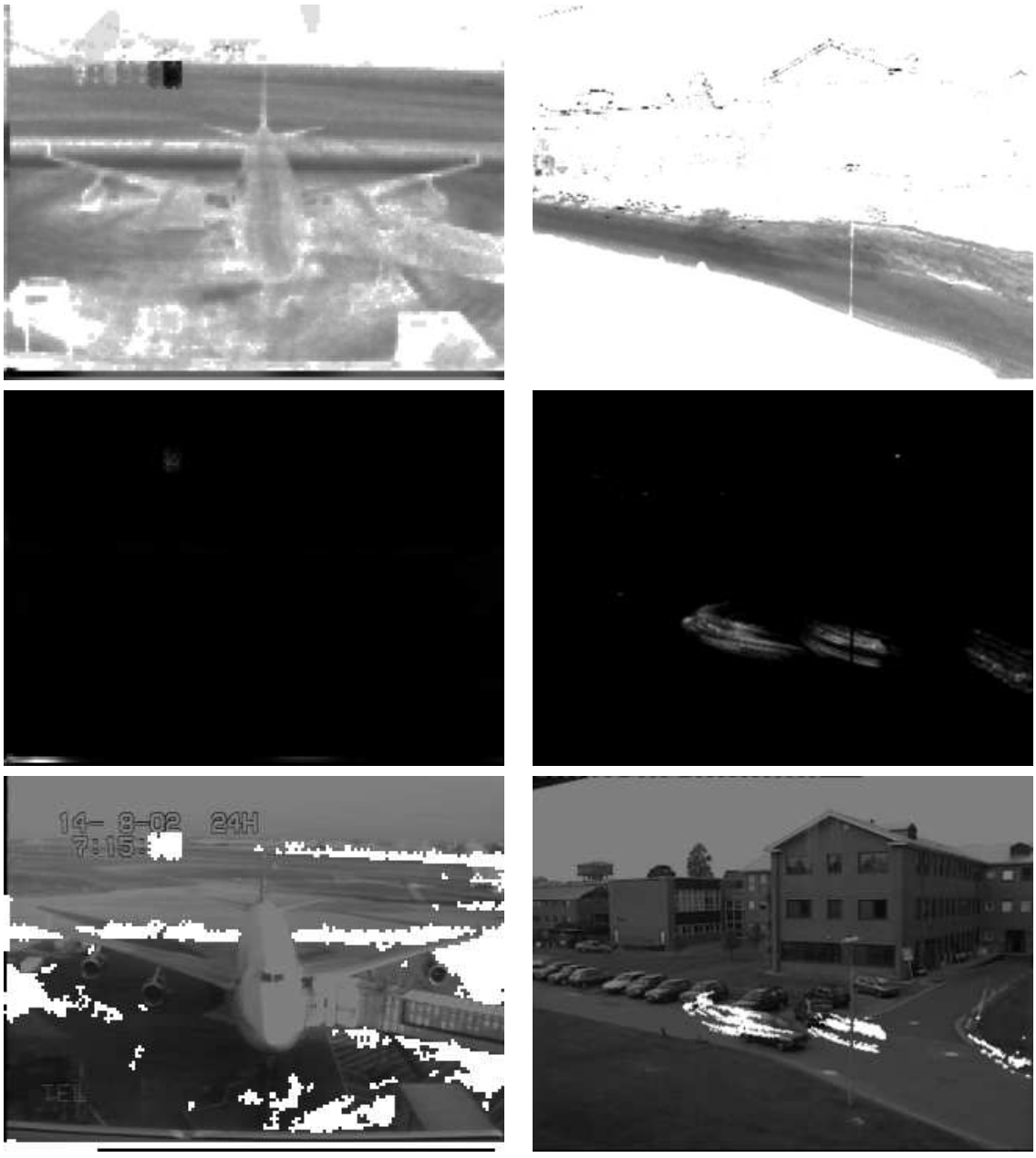


Figure 4.10: Action-based summaries for two scenes. (top) The *DiscriminationAbility*. A high/white value occurs when the pixel is rarely active and consequently is of more interest in the summary. (middle) The *TotalD*. The total level of activity for a detected segment. (bottom) The Segment summary frame. The static frame context is combined with the Top discriminant active pixels (TDAP). It provides both information about the static visual appearance of the scene, and also the important scene action.

Test sequences

We tested our approach using a variety of surveillance scenes that show indoor, outdoor, artificial and natural content. This variety is typical of a surveillance system and highlights the need for an efficient approach for partitioning and visualisation. The sequences used are detailed below:

Aircraft1-5 showing the Aircraft docking scene shown in Figure 3.1 on page 48. Each sequence is of spatial dimension (320×240). The sequence lengths of the five sequences are $t = \{11000, 6470, 2869, 2642, 2434\}$.

Pets1-2 showing the Pets carpark scene shown in Figure 4.11 on page 103. The spatial dimension is (760×540) and lengths $t = \{3061, 3064\}$. The two sequences capture the same action recorded concurrently from two separate camera mounts.

Pets3-4 are as Pets1-2 with $t = \{2688, 2688\}$.

Courtyard1 shows an outdoor courtyard scene shown in Figure 4.11. The spatial dimension is (760×540) and length $t = 2982$.

Selection of parameters

To avoid the need for tuning parameters to each individual sequence, the representation, trajectory and visualisation are all computed using a single set of parameters. This is a realistic situation in that the system can process a new surveillance sequence with no knowledge or expectations of the content. Each sequence was processed as follows:

1. We compute the Sustained temporal change developed in Chapter 3 using the parameters from that Chapter. Namely, $[\alpha = 50, \beta = 50, T_{diff} = 5]$ are used when computing Equation (3.9) to produce a reliable and efficient estimate of what is changing in the scene

2. An Iconic visual vocabulary was computed using a medium sized cell $\lambda = 16$ (see Section 4.2.1) and the number of clusters $\kappa = 20$ (see Section 4.2.2). These selections were found to produce a compact frame-based representation that was able to retain important scene content.

The *Pets carpark* scene, obtained from the Performance Evaluation of Tracking and Surveillance workshop (PETS), shows a carpark scene containing car, bicycle and pedestrian activity. See (Ferryman, 2003). This view is from camera mount 1.

The *Pets carpark* scene. This view is from camera mount 2. The same action is shown as that observed from mount 1.

An *Outdoor courtyard* scene. The scene contains pedestrians, a fountain, and vehicle content. The foreground also contains moving trees. The main problem with interpreting this scene is that no structured action occurs, i.e. the content seems almost random.

Figure 4.11: Two scenes used. Note that the Pets carpark scene is observed from two separate camera mounts, capturing the same action concurrently.

3. To detect long term sequence change, the Video scene trajectory was computed (as Section 4.2.4) using three principal components, $\omega = 3$, and approximated by the Discrete Curve Evolution algorithm (see Section 4.4) using 10 points. Using three dimensions produces a trajectory that is intuitive to compute and visualise. It was found that ten vertices are sufficient to approximate a sequence whilst capturing the important changes, although this can be easily changed as required.

4. A video summary is produced by computing the Segment summary frame (as Section 4.5) for each segment using the most discriminant pixels, $\tau = 25\%$.

We found this set of parameters through experimentation and tuning, to provide a balance between compactness, efficiency and expressiveness. It must be stated that minor changes in the parameter values makes little difference to the result, and so we consider the choice to more about domain context and computational limits. To verify this assertion, a Video scene trajectory was computed for the Pets1 sequence using various values of cell-size (λ) and vocabulary size (κ). The result, shown in Figure 4.12, reinforces our view that the approach is relatively parameter insensitive.

Computation of Video scene trajectories

Figure 4.13 illustrates the result produced using the Aircraft3 sequence. It shows:

- The Video scene trajectory in three dimensions.
- The trajectory approximation using ten vertices to encapsulate nine scenes.
- The resultant nine Segment summary frames along with the frame ranges for each segment.
- In order to aid manual interpretation, a number of the Segment summary frames are shown on the trajectory at the corresponding segment positions.

Upon inspection of trajectory, it can be seen to contain several long continuous periods of similar or *continuous directionality*. Upon manual analysis of the sequence content, these correspond to periods of similar content. For example, in Segment 3 a large number of related unloading activities are grouped; in Segment 7 the loading activities are

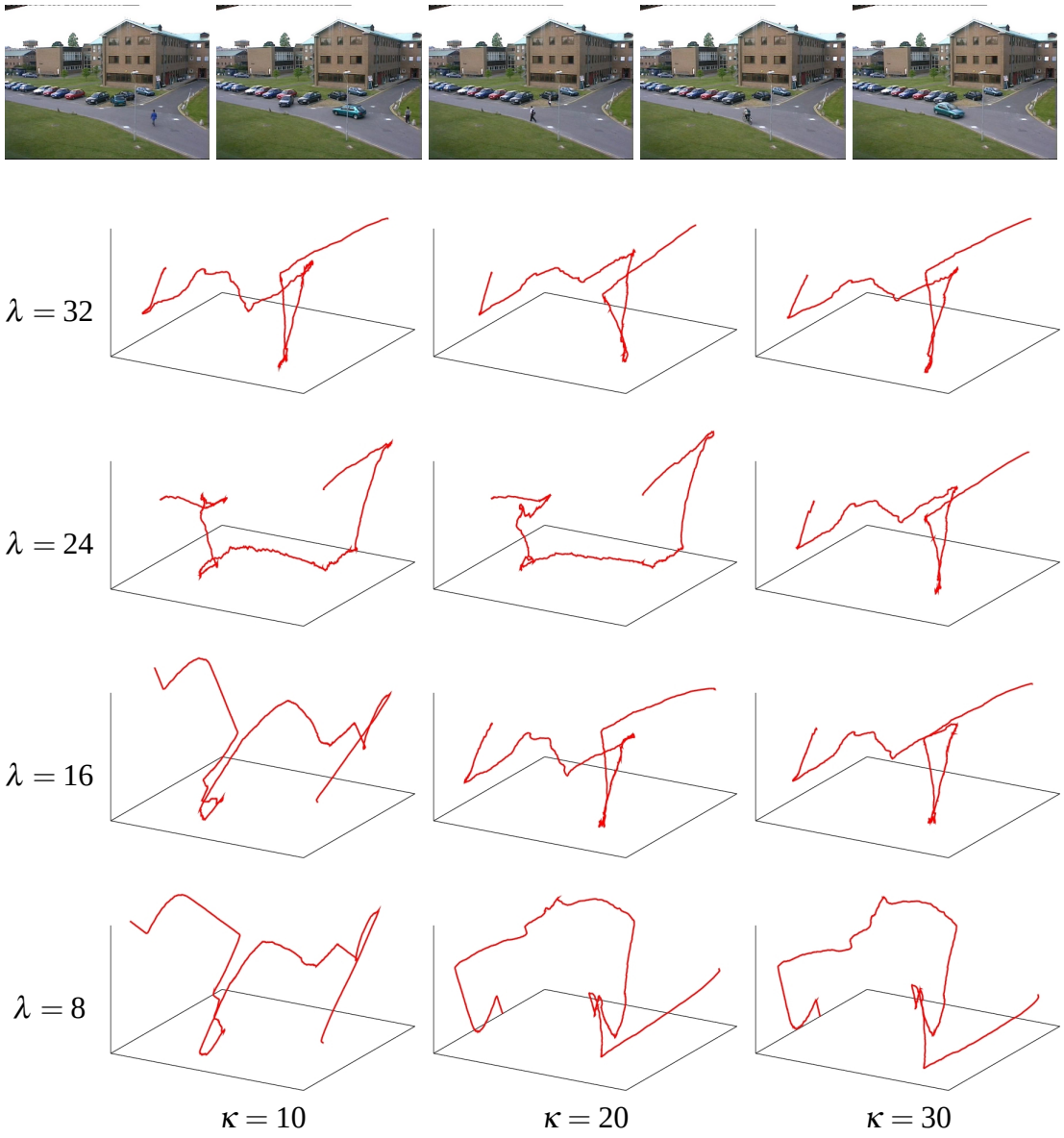
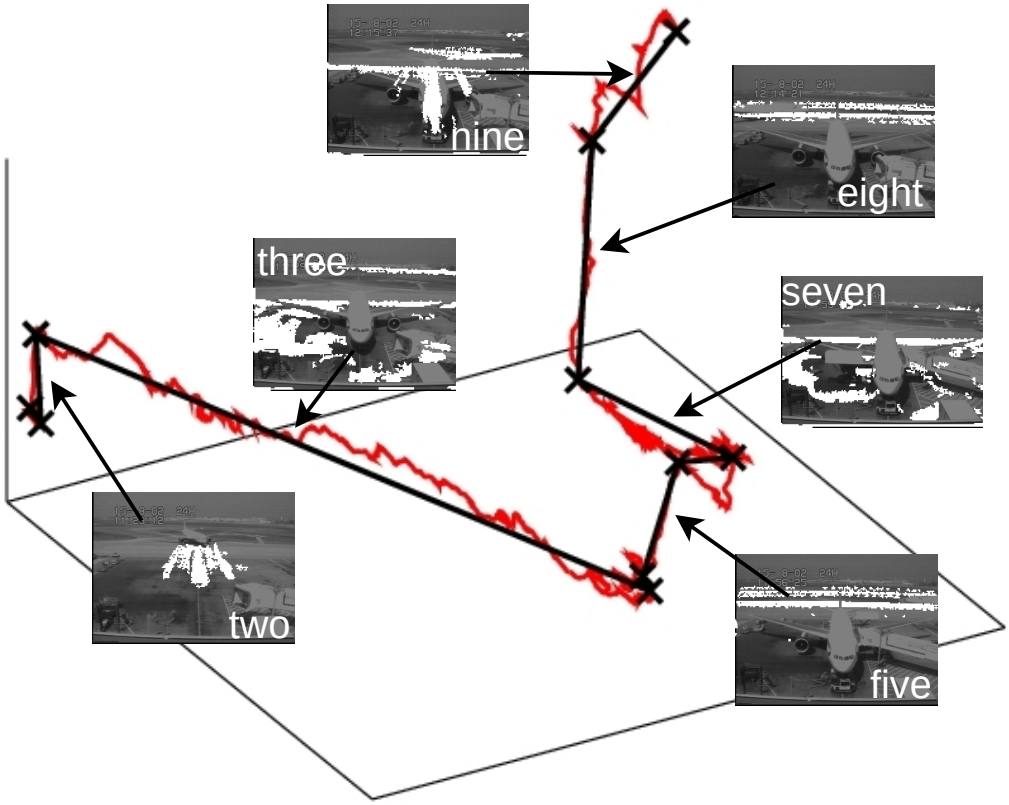


Figure 4.12: The robustness of cell-size (λ) and vocabulary size (κ) as demonstrated using the Pets1 sequence. In each case the trajectory contains a similar profile, with similar main phases and changes. This highlights that the result is not over sensitive on the selection of parameters.




Segment 1: 0-141


Segment 2: 141-200


Segment 3: 200-3104


Segment 4: 3104-3536


Segment 5: 3536-3648


Segment 6: 3648-3965


Segment 7: 3965-5784


Segment 8: 5784-5946


Segment 9: 5946-6468

Figure 4.13: The Video scene trajectory and segment based summary produced for the Aircraft3 sequence. The trajectory is approximated using 10 points leading to 9 segments. It can be seen that the approximation closely follows the structure of the trajectory and, assuming that the trajectory captures the scene content change, the retained vertices are the break points.

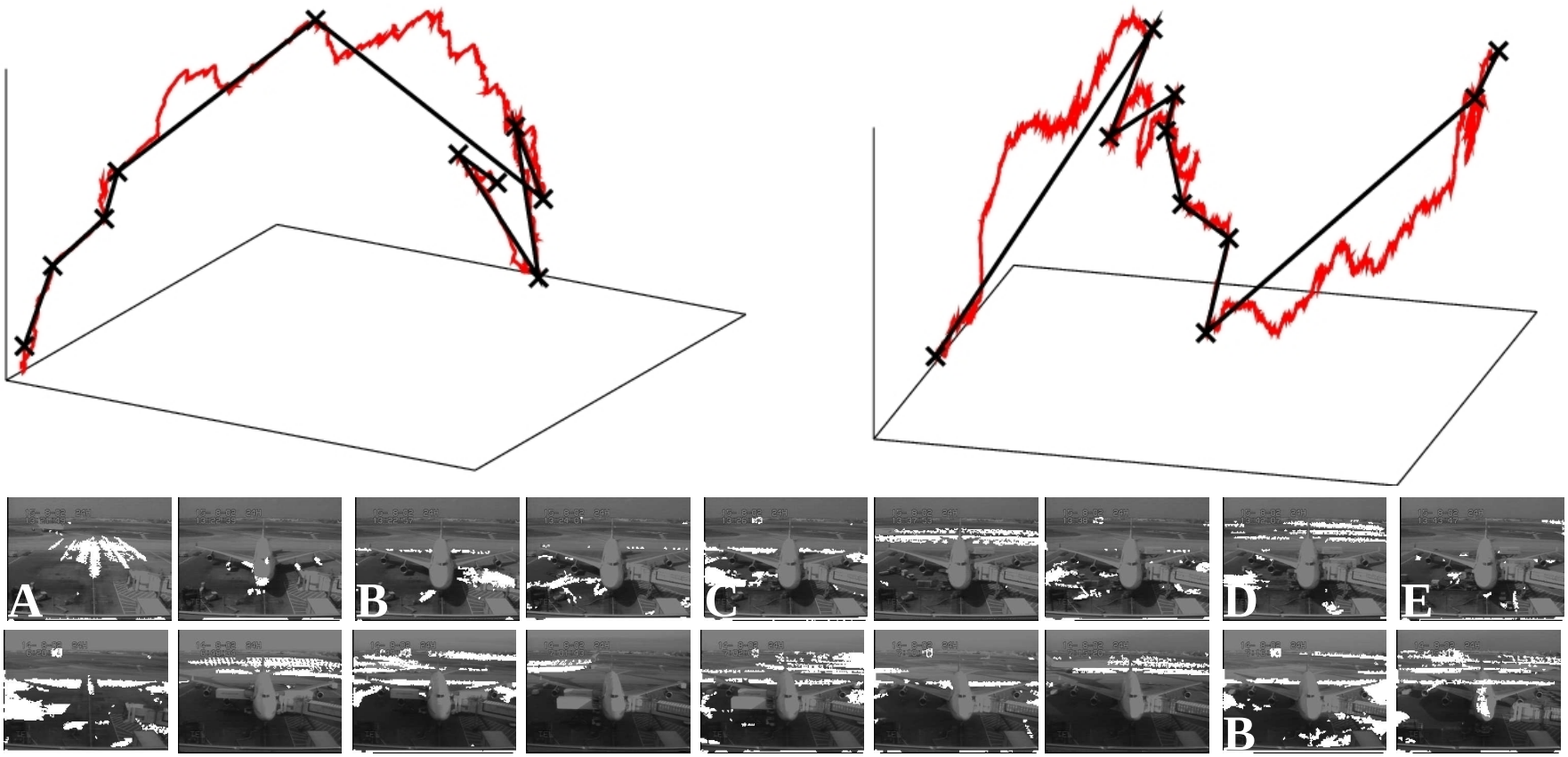


Figure 4.14: The trajectories and summaries produced for the Aircraft1-2 sequences. Note that the segment summary frames provide an indication on the scene content and can be quickly interpreted by a trained operator. For example, to provide instant information on the status of the loading bay: empty-full loading-unloading. For a discussion of the marked summary frames, ABCDE, refer to the text.

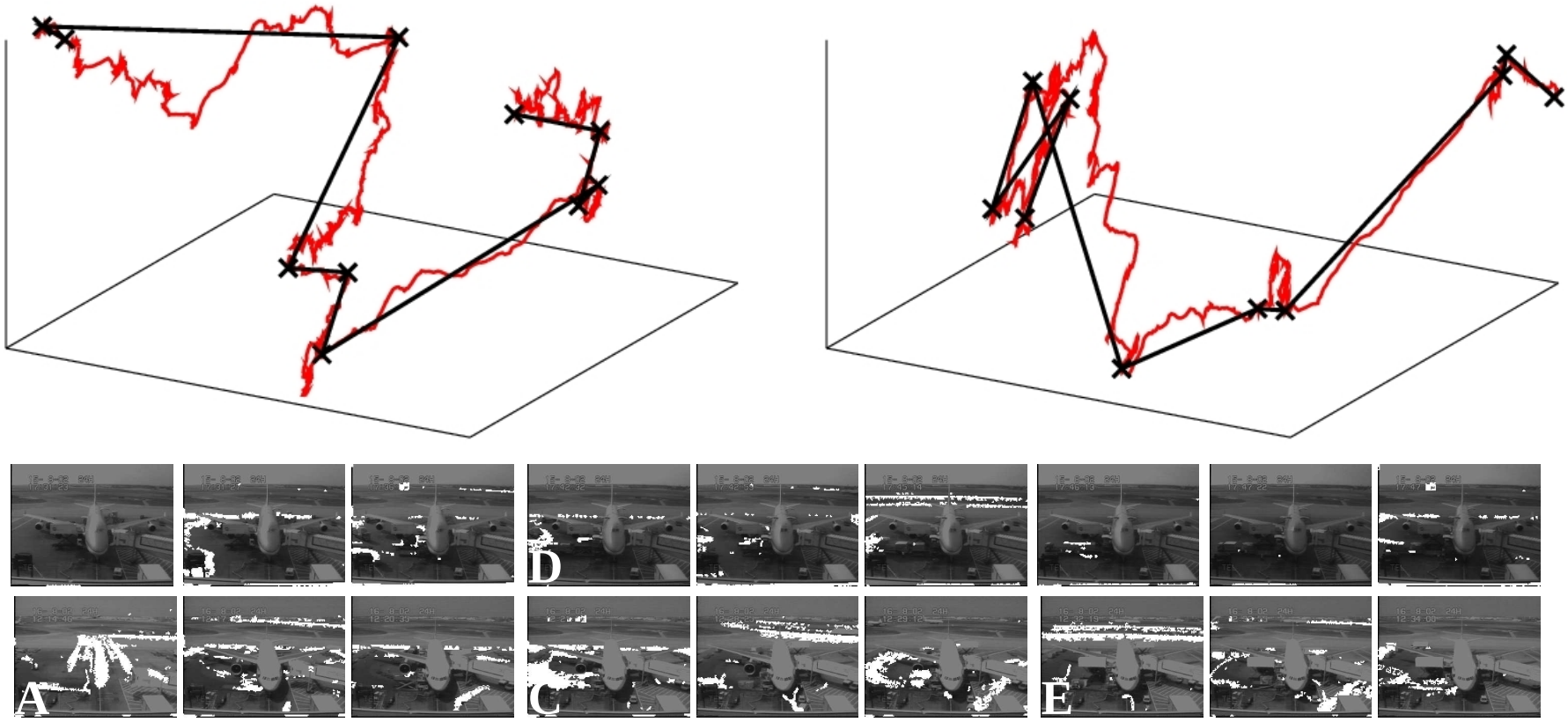


Figure 4.15: The trajectories and summaries produced for the Aircraft4-5 sequences. As Figure 4.14, the segment summary frames indicate the scene content. For a discussion of the marked summary frames, ABCDE, refer to the text.

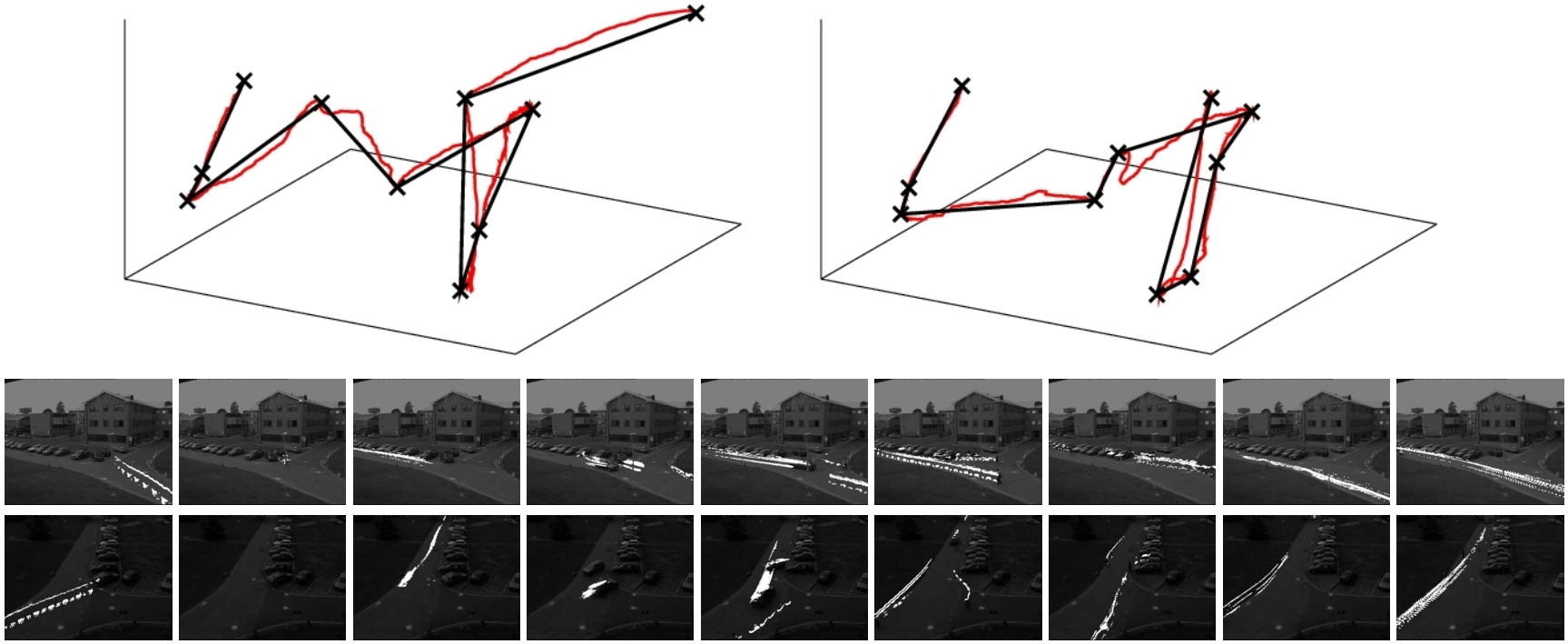


Figure 4.16: The trajectories and summaries produced for the sequences Pets1-2. The sequences are of the same scene action content as observed from two separate camera mounts that provide a different view. Clearly, the trajectories and approximations are highly similar meaning that the approach is reliable. It can also be seen that the segment summary frames produced are correlated, i.e. they show the same action.

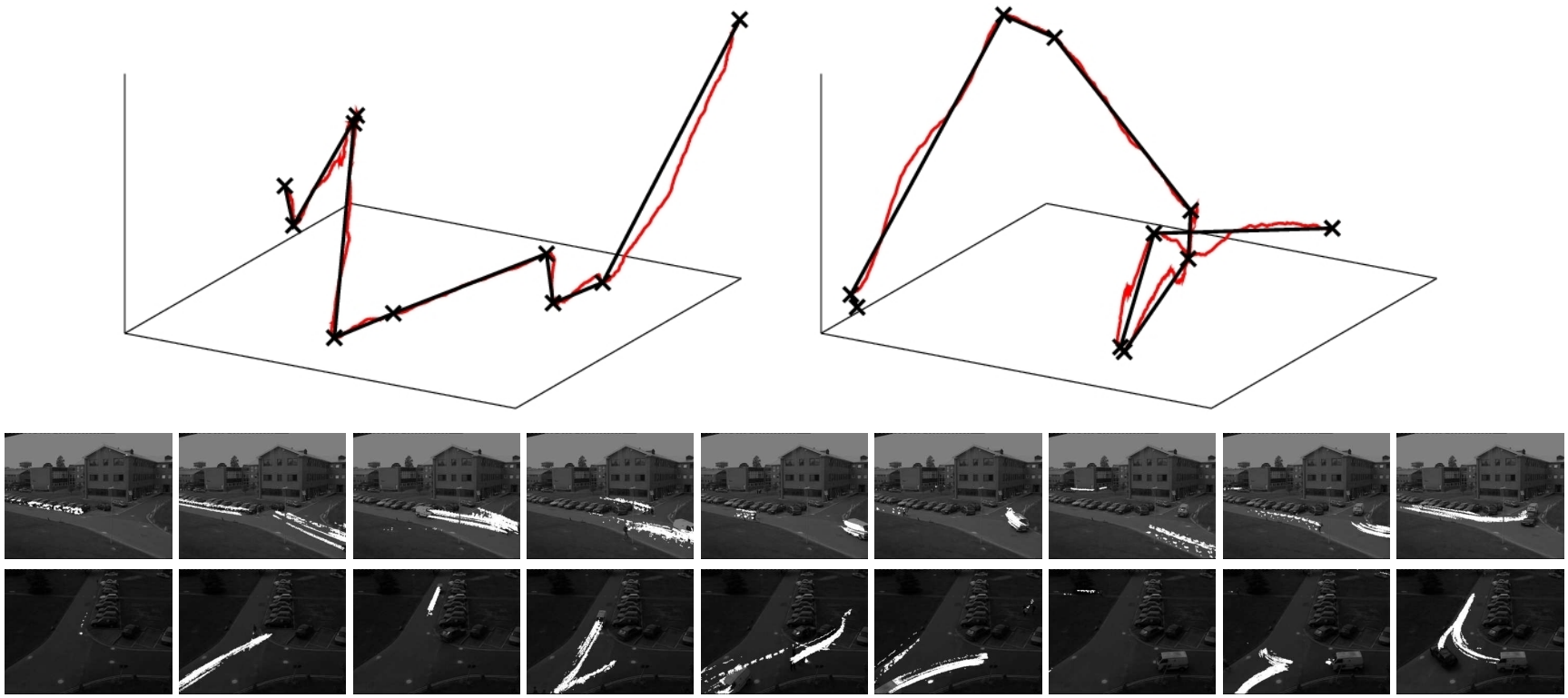


Figure 4.17: The trajectories and summaries produced for the sequences Pets3-4. The sequences are of the same scene action content as observed from two separate camera mounts that provide a different scene view. It can be seen that, similar to Figure 4.16, the trajectories and segmentations are similar, and that the summaries show the same action. This reinforces our view that the approach produces a reliable result.

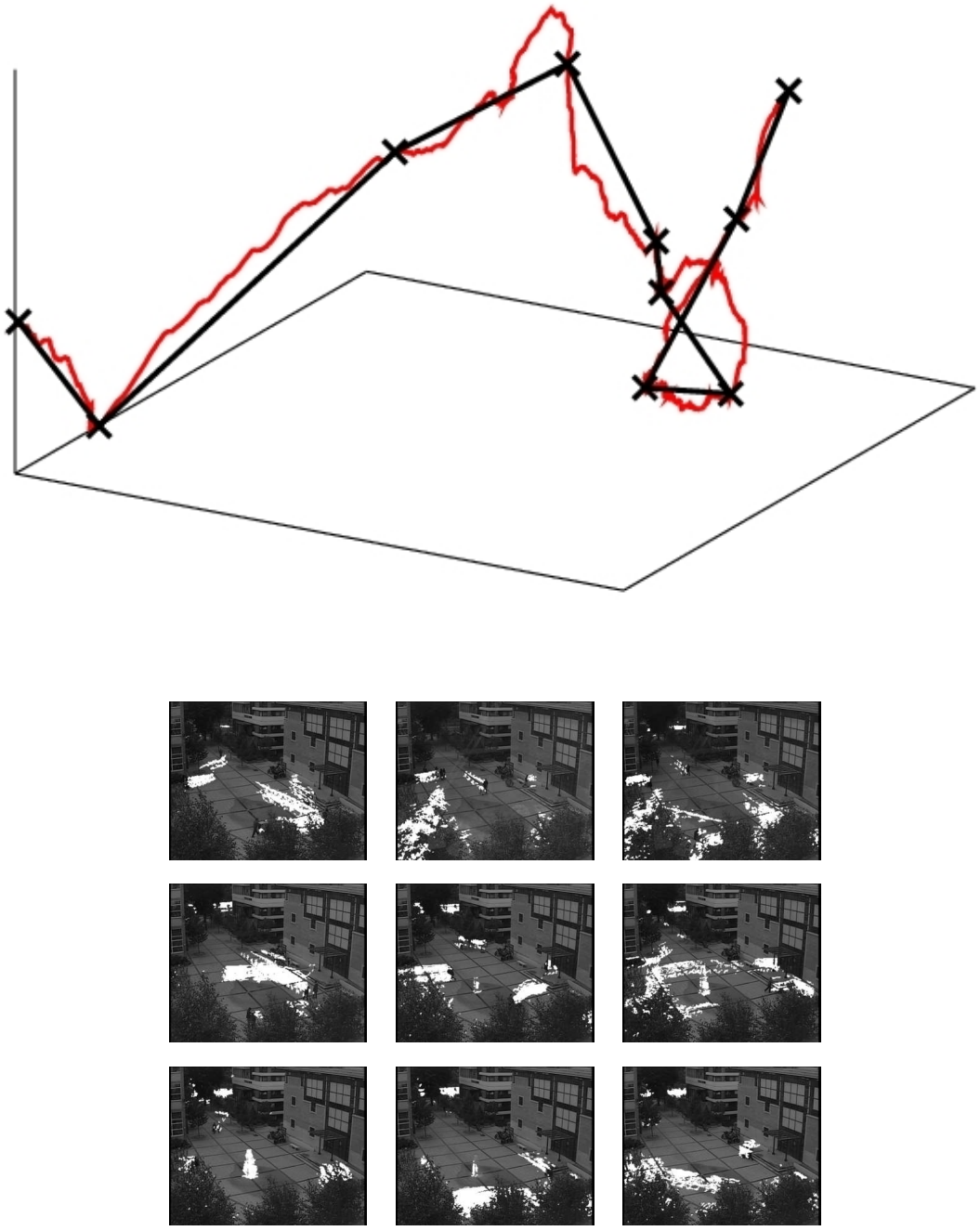


Figure 4.18: The trajectory and summary produced for the sequence Courtyard1. The scene contains no structured content and segmentation and summarisation is therefore difficult. It is generally not possible to discover structure and content where none exists.

grouped. This demonstrates a fundamental characteristic of the trajectory: it does not deviate when the scene action content remains similar. As seen in Figure 4.13, the approximation is able to discover the important points of discontinuity in the trajectory. These become the *breaks* that encapsulate segments. Finally, the combination of static and action features in the segment summary frames successfully portrays the content of the segment. The approach eases and reduces the the time required for manual video interpretation (in comparison to "watching" the sequence).

To demonstrate the consistency of the solution, similar trajectories and summaries were computed for four other aircraft docking sequences - Aircraft1,2,4,5 - with the results shown in Figures 4.14 and 4.15. It can be seen that the resultant trajectories capture content change and lead to an effective partitioning. Also, the summaries contain clear action based content - with static context - and are meaningful to a trained human observer and so facilitate a semi-automatic surveillance investigation system. In particular, a number of repeating activities are highlighted:

A The plane arrival event is easily determined from both the lack-of-plane in the static context and the active pixels.

B Many unloading activities.

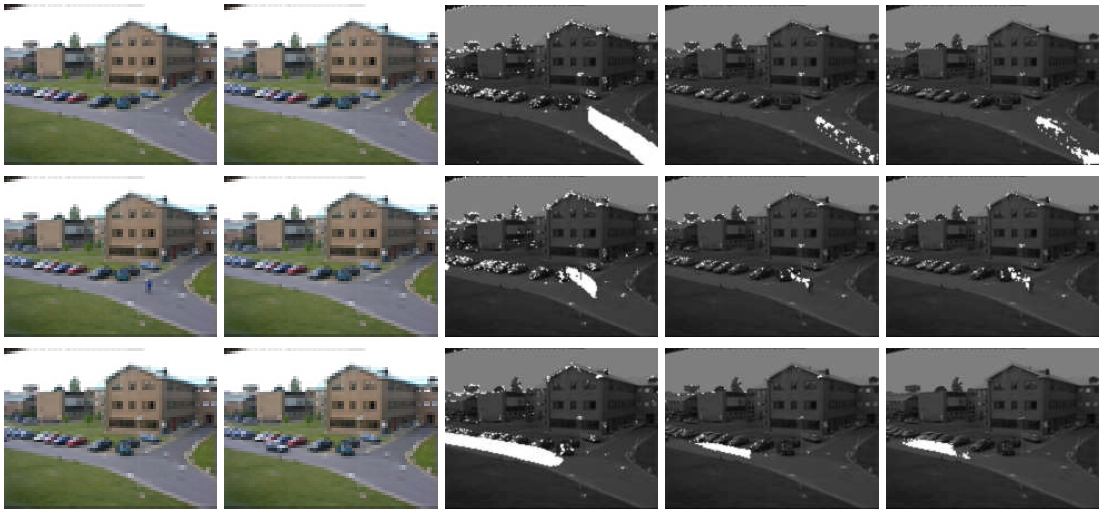
C Many loading activities.

D Examination of the plane engines.

E Examination of the Aircraft front wheel.

The repetition of similar actions is to be expected, as the aircraft docking scene contains a clear cycle of known action - arrival, unloading, loading, departure - that a trained observed can identify and use to establish the current status of the loading bay. The summaries contains a precis of the scene content and, if produced on-line, can be used for monitoring and intervention applications.

To demonstrate the approach on a more controlled environment, trajectories and summaries were produced for four sequences that show the Pets carpark scene. Figure 4.16



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

Figure 4.19: Three automatically detected segments/scenes from the Pets scene with alternative summaries: (a,b) the first and mid frame, (c) 100% of active pixels in the scene, (d) the top 25% most active pixels, (e) the top 25% discriminative pixels. The discriminative pixel approach reduces the effect of noisy pixels, for example those on the lamppost due to camera shake.

shows the results for the sequences Pets1-2, that show the same scene content as concurrently observed from two separate camera mounts. We are interested in the consistency of the result produced for the same content. It can be seen that the trajectories are very similar and that the summaries complement each other. Figure 4.17 shows the result produced for the sequences Pets3-4. These are also similar, but not quite to the same extent. We conclude from this experiment that - as the trajectory and summary produced is similar for two sequences that observe the same action - the approach produces a consistent and reliable result.

Finally, the approach is demonstrated on a complex, outdoor, natural courtyard scene Courtyard1 that contains many unrelated, unstructured activities. The use of discriminant pixels during the summary construction is able to reduce the influence of the noise produced by the shimmering trees in the immediate foreground. These pixels are so noisy that they, correctly, are not highlighted in any of the segment summary frames. The trajectory produced contains clear phases of content and can be used for investigating the sequence. However, it may be concluded from this experiment that to compute and use a temporal segmentation for an *unstructured sequence* is not generally possible.

Effectiveness of the summarisation

Figure 4.19 shows a frame based summary, an approach using the most active pixels, and our discriminative pixel approach. We find that static frames do not provide information about the action content and that the most active pixels approach contains noise, in this case due to camera shake. The use of discriminant pixels reduces the effect of noise and produces a clearer action based segment summary.

4.7 Discussion

We have presented a video representation used to perform video indexing of unstructured surveillance video. A key goal of the approach is the assumption of very little knowledge about the scene. This is necessary owing to the retrospective nature of search tasks and also the scalability issues in the surveillance domain. To this end, we proposed:

- An uncommitted frame representation based upon the extraction of invariants that capture the appearance of local action. This is achieved by firstly computing the reliable action for a scene, then dividing the image space into a grid of equally sized cells, and finally, computing the Haar wavelet transform for each cell. This provides coefficients that capture the directionality and visual structure of action.

- The construction of an Iconic visual vocabulary for scene description. The coefficient space for a sequence clustered and each frame described using the number of Iconic terms (the centroids) that occur within it. The result is a very compact and generic description of scene content.

- The model order selection, i.e. which terms to employ, was achieved by examining the entropy of an Iconic terms occurrence in a sequence. The entropy provides information on the distribution of a term, and therefore its discrimination ability.

The advantages of the representation are that it does not require explicit object and activity detection and tracking, that can be problematic in outdoor surveillance scenes. Tracking systems often require a rich feature landscape, including colour, that is not available. In addition, the representation does not assume any specific content when it is

computed. This is critical if large number of videos of diverse content are to be indexed and retrospectively searched. We call this characteristic an *uncommitted index*.

As discussed in Chapter 2, two important functions that are required in an indexing system are the ability to perform temporal segmentation and summarisation. We therefore proposed:

- The content changes in the scene are captured by forming a Video scene trajectory in which a stable, or continuous, directionality is indicative of similar content. This is achieved by computing a cumulative analysis of the frame iconic occurrence histograms to find the temporal variations in term occurrence, then the computation of the Principal Components Analysis, and finally the projection of the frames into a low-dimensional subspace that captures the variations. The result is a trajectory that captures the scene-level content change.

- The break points in a sequence are found by the discretisation of the trajectory. The trajectory is approximated using a small number of vertices and each retained vertex becomes a break position.

- A video summarisation is formed by computing the most discriminative active pixels in each detected segment and the formation of a segment summary frame. The set of segment summary frames form a video summary that conveys the scene action content.

The advantage of the trajectory approach is that it is able to provide information on how the sequence is *changing over time*, as opposed to providing information on the static or dynamic content. The result is that the approach is content independent. The summarisation approach was shown to provide pertinent action information for a sequence.

In the previous two chapters, two approaches were presented for video representation and temporal segmentation:

- In Chapter 3, activities were segmented as spatio-temporally connected active cells, and a temporal segmentation computed using frame-based or activity-based sliding window.

- In Chapter 4, a sequence is described using a vocabulary of invariants and a trajectory formed to captures the scene content changes. A temporal segmentation was computed using a trajectory discretisation.

Both chapters present an uncommitted approach, in that no content expectations are made on the scene. However, it may be argued that during a manual process of video investigation - for example, retrospective search - semantics, or knowledge, about the scene content becomes available as the manual search progresses. For example, certain periods may be tagged as important-or-not or containing-a-semantic-or-not. This is a similar concept to that of relevance feedback in text-based information retrieval systems. During pre-attentive indexing, this information cannot be assumed. However, the next Chapter addresses the problem of how semantic information can be integrated to achieve a semi-semantic search process.

Chapter 5

Semi-semantic analysis

In recent times, the *semantic gap* problem has motivated the desire for automatic semantic video analysis to detect and label content. Action history based methods have been reported for the recognition of indoor activities (Bobick and Davis, 2001) and complex outdoor events (Gong and Xiang, 2003). However, they are dependent upon successful training, the adequate provision of training data, and large computational resources. A *temporal constraint* is used to improve action recognition by enforcing strict temporal order. For example, by using a hidden Markov model where states correspond to different stages of appearance transformation. Unfortunately, the temporal order assumption is not valid in more natural scenes, and also the sensory and semantic ambiguity is high.

In this Chapter, video analysis is performed without either explicit model building or a temporal constraint. Firstly, in Section 5.2, a competing Gaussian mixture models is described that can perform unconstrained activity recognition. However, the approach is vulnerable to training issues. Second, in Section 5.3, a novel *rank-voting framework* is proposed for performing fast, uncommitted semantic analysis and browsing of video. The positions of labelled items in a content-based retrieval ranking are used to vote for their respective semantic labels. Third, in Section 5.4, a Bayesian framework is proposed for performing a combination of evidence to achieve fusion. A constructive inference algorithm is also described.

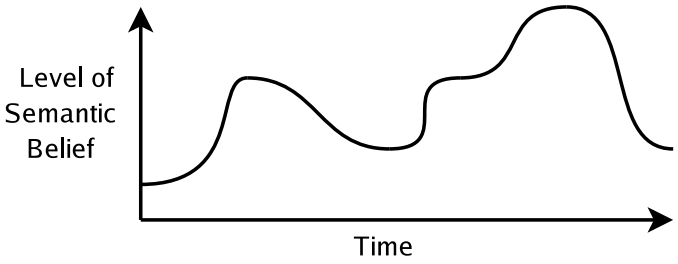


Figure 5.1: Illustration of a semantic graph. The plot shows the confidence in a particular *semantic* as it varies over time. In this example, the semantic occurs in two phases but is most prevalent towards the end of the sequence. Such a graph, when used by a browsing system, permits the user to navigate towards the desired goal (i.e. the occurrence of the semantic in the sequence).

5.1 Semantic belief based browsing

A video sequence is typically browsed by navigating its hierarchical structure, or by viewing a sub-sampled frame-set. However, if a recognition system is able to recognise a semantic label, ω, an alternative is to present a *semantic graph* to the user with the level of semantic belief presented over time. See Figure 5.1. Such a graph highlights the occurrence of a semantic in the sequence and permits a user to navigate towards a desired result. It is the role of the indexing system is to discover the confidence levels - or *belief* - for a set of semantics:

$$\mathcal{S} = \{s(\omega_i, t) \; : \; \forall i \in [1, m]; \forall t \in [1, n]\} \tag{5.1}$$

where m is the number of semantics, n the number of frames, and s the level of belief in semantic ω at time instant t.

For surveillance scenes, example semantics could be “car parking”, “plane arriving”, or “tea making”, that occur at a time instant. It is, therefore, a similar but distinct problem to that of temporal segmentation discussed in Chapters 3 and 4 in which the discontinuity in scene semantics is required. Clearly, a classic approach for building a semantic graph is to perform *training and recognition*. However, such approaches are often dependent upon clean data capture, unambiguous feature spaces, and a lack of noise (Bishop, 1995) and consequently are of limited use for outdoor scenes. In addition, it is desirable for the semantic graph to be adaptive because the search task is not known when the se-

quence is indexed. This is similar to the concept of *relevance feedback* in information retrieval systems: as a user marks documents as relevant the system is expected to alter its perception (Baeza-Yates and Ribeiro-Neto, 1999).

5.2 Competing models approach

It is widely understood that recognition can be achieved by training several competing, stochastic models to each recognise something of interest. We propose using the flexible Mixture of Gaussians to model the Haar coefficients developed in Chapter 4, for generating the semantic graphs.

5.2.1 Expectation maximisation training

A Gaussian Mixture Model (GMM) is a semi-parametric model used to estimate the probability density function of a feature space $\mathcal{X}$. The model Θ consists of several independent Gaussian distributions θ that when combined using mixing parameters α form a single result, so $\Theta = \{(\alpha_i, \theta_i)\}; i \in [1, \kappa]$ where κ is the number of components. Each component is a Gaussian, $\theta_i = (\mu_i, \Sigma_i)$, comprised of a mean μ and covariance matrix Σ. The model captures the expressiveness of the feature space using minimal parameters. The posterior probability of a data-point, x, is computed as:

$$p(x|\Theta) = \sum_{i=1}^{i \leq \kappa} p(x|\theta_i)\alpha_i \tag{5.2}$$

where $p(x|\theta_i)$ is the posterior probability of x given component i:

$$p(x|\theta_i) = \frac{1}{(2\pi)^{N/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}[x-\mu_i]^T\Sigma_i^{-1}[x-\mu_i]\right) \tag{5.3}$$

where N is the dimensionality and $[x-\mu]^T\Sigma^{-1}[x-\mu]$ the Mahalanobis distance. Note the sum of all mixing parameters, $(\sum_i^\kappa \alpha_i) = 1$.

The Gaussian Mixture Model is a semi-parametric model for which a number of parameters are needed to be estimated during a model fitting process (Gong *et al.*, 2000). This can be achieved using the Expectation Maximisation (EM) algorithm (Dempster *et*

al., 1977) that computes a Maximum Likelihood estimate of the parameters:

$$L = p(\mathcal{X}|\Theta) = \prod_{i=1}^{n} p(x_i|\Theta) \tag{5.4}$$

where n is the number of training samples. For practical purposes, the error function is minimised:

$$E = -\ln L(\Theta) = -\sum_{i=1}^{n} \ln p(x_i|\Theta) \tag{5.5}$$

as the negative log is a monotonically decreasing function. The algorithm consists of the following two steps that are iteratively performed until the error converges:

- The *Expectation* step. The current expectations are computed for all training samples and all component Gaussians

$$P(\theta_i|x) = \frac{p(x|\theta_i)P(\theta_i)}{p(x)} \tag{5.6}$$

- The *Maximisation* step. The new parameters are estimated:

$$\mu_i^{new} = \frac{\sum^{j<n} P^{old}(\theta_i|x_j)x_j}{\sum^{j<n} P^{old}(\theta_i|x_j)} \tag{5.7}$$

$$\Sigma_i^{new} = \frac{\sum^{j<n} P^{old}(\theta_i|x_j)\left[x_j - \mu_i^{new}\right]\left[x_j - \mu_i^{new}\right]^T}{\sum^{j<n} P^{old}(\theta_i|x_j)} \tag{5.8}$$

$$P^{new}(\theta_i) = \frac{\sum^{j<n} P^{old}(\theta_i|x_j)}{n} \tag{5.9}$$

Unfortunately, the accuracy of EM is known to be vulnerable to variation in the initialisation procedure and to local maxima. To reduce this risk, the model is initialised using the following method adopted from (McKenna *et al.*, 1999): component means are initialised by k-means clustering; covariance matrices are initialised to $\Sigma = I\sigma$ where I is the identity matrix and σ is the Euclidean distance between the mean and the closest other mean. The initialisation approach is illustrated in Figure 5.2. The correct initialisation approach can minimise the risk of local minima in the likelihood function and can speed-up convergence.

The choice of the number of components κ is critical as too few components will not adequately capture the density, and too many components will over-fit the data and render the model unable to generalise. Many schemes exist for automatically choosing κ. Schemes based upon the Minimum Description Length (MDL) principle attempt to balance the quality of the model fit (the likelihood) against the complexity of the model. A number of candidate models are trained with a κ in a desirable range, then the MDL is computed for each model, and finally, the model with the lowest MDL is considering that achieving the best balance and is selected for use. Although common in literature, we feel that the approach is unsuitable for large scale datasets because it requires multiple and repeated model fitting at great computational cost. Rather, an on-line pruning approach is adopted:

- The model is initialised with a large number of components
- During the EM iterations, components are removed according to a *pruning* criteria. Components are removed if they are responsible for too few data-items or if their mixing parameter α is zero.

The model fitting and pruning process is illustrated in Figure 5.3. The models are shown at various stages of fitting. Gaussian components being removed is clearly observed.

5.2.2 Haar-based models for sequence investigation

After a mixture model of Haar coefficient distribution for a given scene is estimated, using a subset of sampled video frames, a semantic graph can be constructed for any video sequences of the same scene from novel observations. This is achieved by, for each frame, computing the posterior probability that it was generated by the model.

Figure 5.4 shows the approach computed using a short sequence of a square pattern being drawn in the air: (top row) shows illustrative frames; (second row) shows the total activity for the sequence along with an indication of the frames that were used for training; (third row) shows the model output from two training mixture models; (bottom row) shows a comparison of the model output $\omega_A - \omega_B$ that is useful for analysis. It is clearly seen that the models are able to recognise the action. Figure 5.5 shows a short

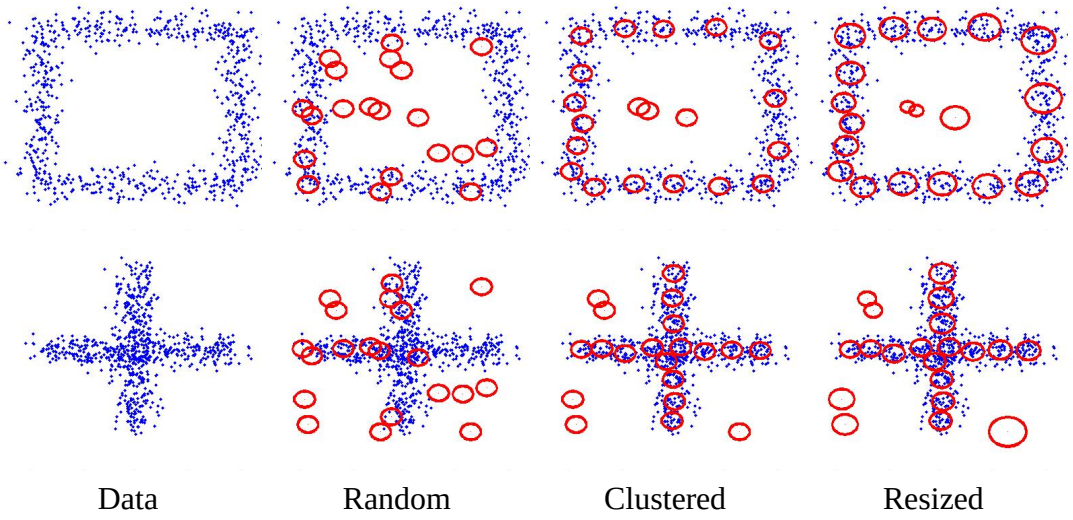


Figure 5.2: An illustration of the model initialisation for two datasets. (Random) 20 Gaussians are generated with random means and unary covariance matrices. (Clustered) K-means clustering is used to initialised the means. The covariance matrices remain unary. (Resized) Each covariance matrix is initialised to $I\sigma$ using the distance to the nearest other centroid σ and the identity matrix I. It can be seen that the resized strategy produces a good initialisation.

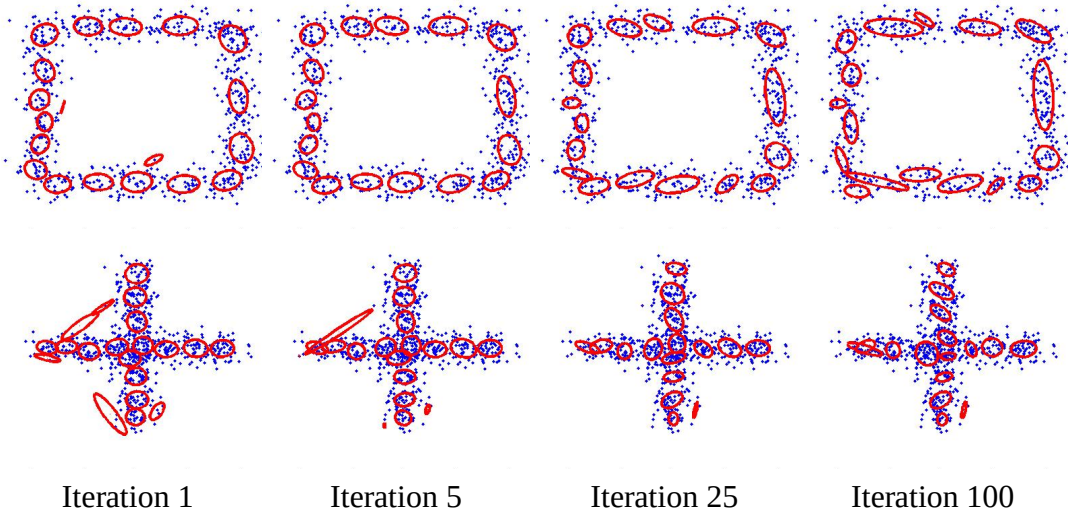


Figure 5.3: An illustration of the Expectation Maximisation fitting process for two datasets. The resized initialisation of Figure 5.2 was used. The model is shown at 1, 5, 25 and 100 iterations. It can be seen that components are removed if they are not responsible for enough data, and that the result after 100 iterations provides a good fit.

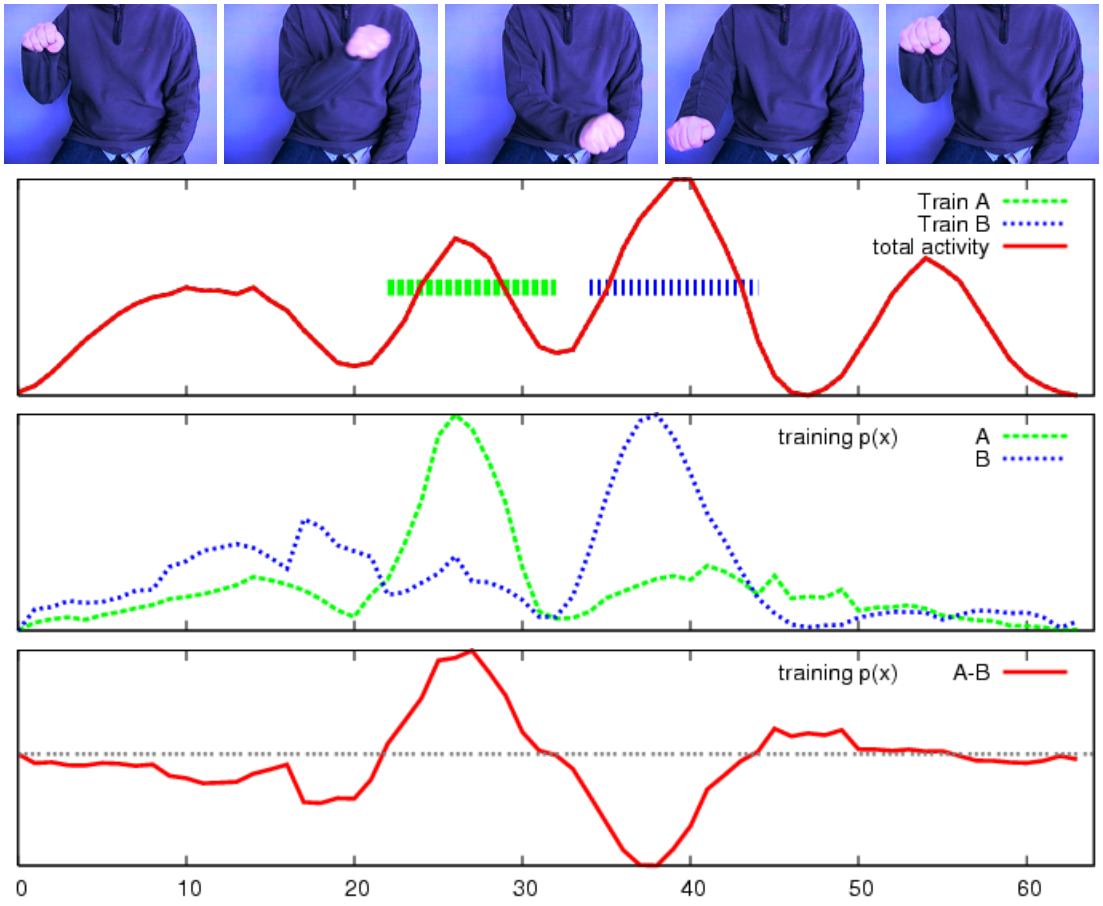


Figure 5.4: The competing models approach. (top) Frames from a short training sequence showing a square shape being drawn in the air. (second) The total level of activity in the scene. The peaks correspond to the left-down-right-up movements in the sequence. Also, we highlight the 10 frames used to train each model. (third) The posterior probability output from each model for the sequence. (bottom) A comparison of model outputs using A-B. Each model is clearly stimulated in its training period.

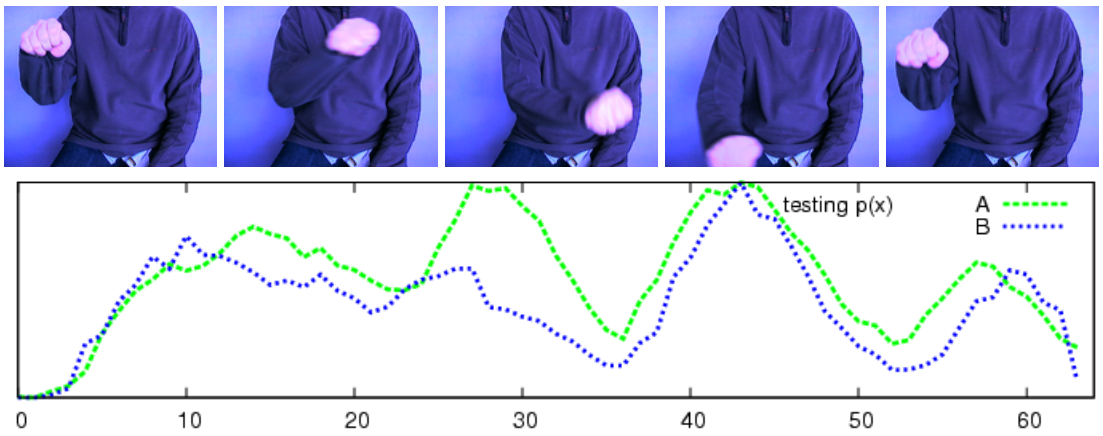
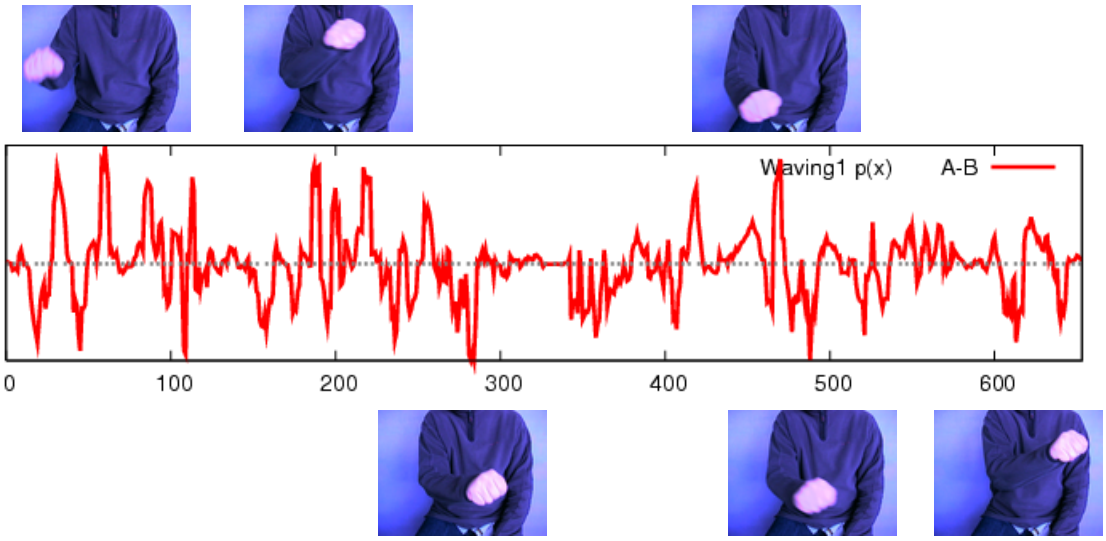
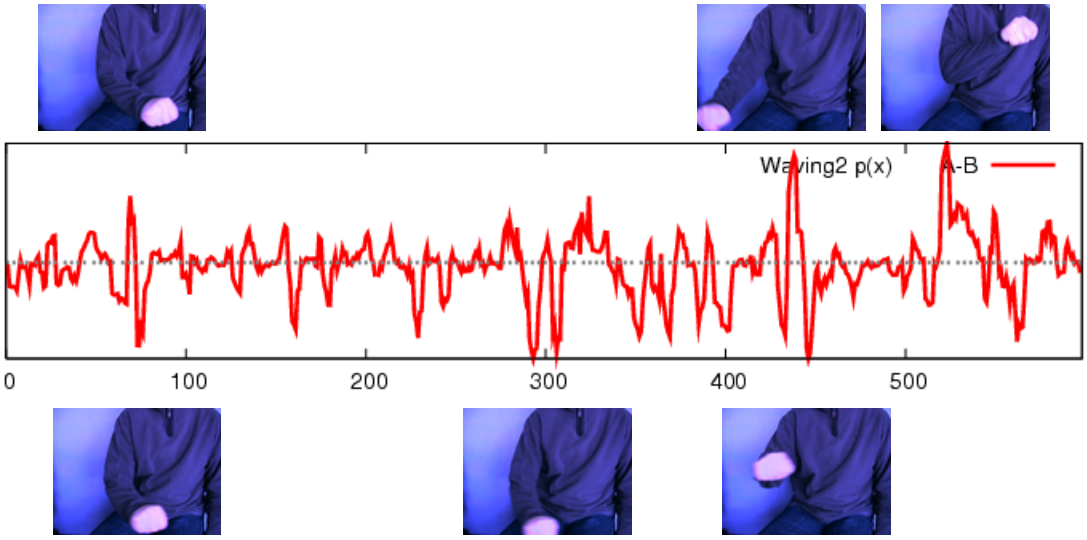


Figure 5.5: The competing models from Figure 5.4 as applied to a novel sequence containing similar action. It can be seen that the models are stimulated both correctly and incorrectly to an extent.



(a) Waving 1: semantic graph produced by competing models.



(b) Waving 2: semantic graph produced by competing models.

Figure 5.6: The semantic graphs produced by the competing models method for two novel sequences of constrained hand signal video data. The graph shows $A - B$ meaning that occurrence of semantic A results in peaks and B in troughs. We highlight frames at notable peaks and troughs. We find that the peaks and troughs, although meaningful with reference to the training data, are noisy and hence performing interpretation is difficult.

novel sequence that contains a similar square pattern and the model output. It can be seen in Figure 5.5 that the model cannot adequately generalise to novel data. Furthermore, Figure 5.6 shows the model output for two longer sequences in the same scene and it can be seen that, although the models are stimulated by different sets of coefficients produced by the different action, the result remains unclear.

It is concluded that the semantic graph obtained from a competing models approach does not justify the computational cost of model training outlined in Section 5.2.1. Also, as model training is required *off-line* the approach is not able to adapt to new semantics as they arise. To overcome these problems, in Section 5.3, an alternative approach is presented that uses a voting algorithm.

5.3 The rank voting method

5.3.1 Frame-based ranking

In Chapter 4 a compact video representation was presented (using the Iconic visual vocabulary) that could be searched using a histogram intersection metric. The process of *ranking* is well understood and widely used in text, image and video retrieval systems (Rijsbergen, 1979; Baeza-Yates and Ribeiro-Neto, 1999; Lew, 2001). The ranking process produces a ranked list of items - the ranking - with the most similar items at the head. A ranking is defined as:

$$\mathcal{R} = \{r_i \ : \ \forall i \in [1,n]\} \tag{5.10}$$

where each item in the ranking, r, is a *rank item* consisting of:

$$r_{pos} = [pos, sim_{pos}, F*_{pos}, \omega_{pos}] \tag{5.11}$$

where $pos \in [1,n]$ is the rank position, sim is the normalised similarity in the range $[0,1]$, $F* \in [1,n]$ is a content identifier (in our case, a pointer to a frame), and $\omega \in [1,m]$ is a frame label if one exists.

5.3.2 Rank positions as votes

If a ranking is generated containing a number of labelled items, i.e. $\omega_{i\in[1,n]} = [1,m]$, then, intuitively, the positions of the labelled items, *pos*, will provide an indication of the content of the query item. A high rank position for a particular label is evidence of that semantic occurring in the query frame. This idea is used to estimate a semantic graph by the following process using each frame as the query in turn:

1. A small number of frames are manually analysed and stored in the index with a label $\omega_i \in [1,m]$. The label indicates the semantic content of the frame. If more than one content type exists, then no label is assigned. It is important to note that all the semantics under consideration must have an equal number of labelled items.

2. A video frame with unknown semantics is used as the query. A ranking is generated with the frame excluded.

3. The top ρ rank items each *cast a vote* for their semantic, if one exists. Each semantic is scored as follows:

$$score(\omega,\mathcal{R}) = \sum_{pos=1}^{pos<\rho} \begin{cases} \log(\rho - pos) & \text{if}\,(\omega_{pos} = \omega) \\ 0 & \text{otherwise} \end{cases} \tag{5.12}$$

   where ρ is the number of rank items at the head to consider, $pos \in [1,\rho]$ is the rank position, ω_{pos} is the label that is checked against the class being scored ω. The vote, $\log(\rho - pos)$, is only cast by labelled items and is larger for items nearer the head.

4. The top ρ similarity scores are used to compute a measure of reliability of the interpretation. The median similarity is used:

$$rel(\mathcal{R}) = \text{median}\,(\{sim_i \,:\, \forall i \in [1,\rho]\}) \tag{5.13}$$

The *score* for a semantic will be high if items with that label appear highly placed in the ranking. See Figure 5.7. The scores for the different semantics are used to populate

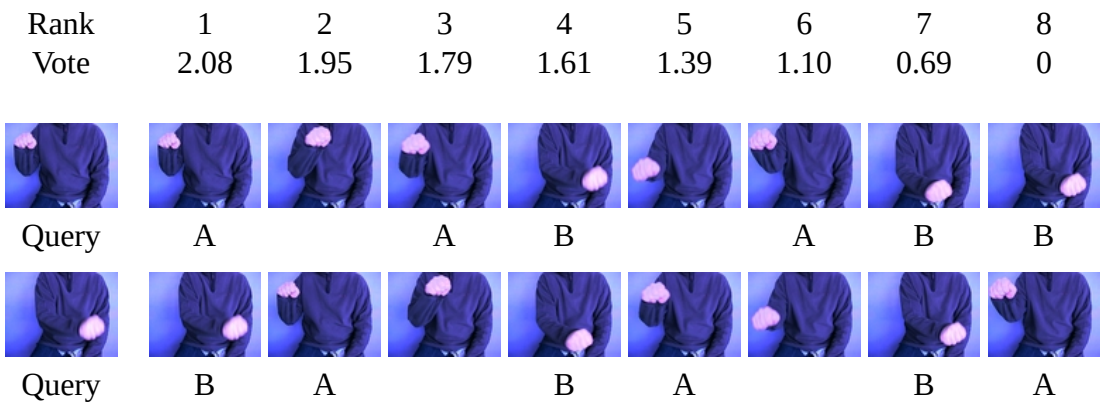


Figure 5.7: An illustration of the rank voting method using two queries. At the top we show the rank positions and the corresponding vote. For each query, a ranking is generated and then the labelled items are used to generate a vote for that label. The results for the first query are $[4.97, 2.30]$ identifying A, and for the sequence query $[3.34, 4.38]$ identifying B.

$\mathcal{S}$. The approach permits visual ambiguity because the labelled items in the index can vary in order to capture all of the visual variations of the semantic; it is quick to setup as the user only has to select example frames; and it can be extended at any time without difficulty. In particular, each labelled item is considered to be independent of other labelled items in the same class.

The approach is illustrated in Figure 5.8 using the same short hand signal sequence used for the competing models of Section 5.2. For the computation of the Iconic index, the parameters outlined in the previous chapters were used. For comparison with the competing models approach, 10 frames were labelled in the index with the two semantics. It is clear from the Figure 5.8 that the rank voting method is able to identify semantics at the training positions, but also able to generalise to the novel sequence in Figure 5.9. Note that to perform the algorithm on a novel sequence: (a) it must be indexed using the same Iconic visual vocabulary as the training sequence; and (b) the labelled items from the training sequence are added to the novel index before rank voting begins. It must also be emphasised that, in contrast to the competing models approach in which a training process must be completed in advance of the search task, the method does not require a prior model fitting. This renders it more useful for an *interactive* search task in that labelled items may be added, removed, altered, during the search.

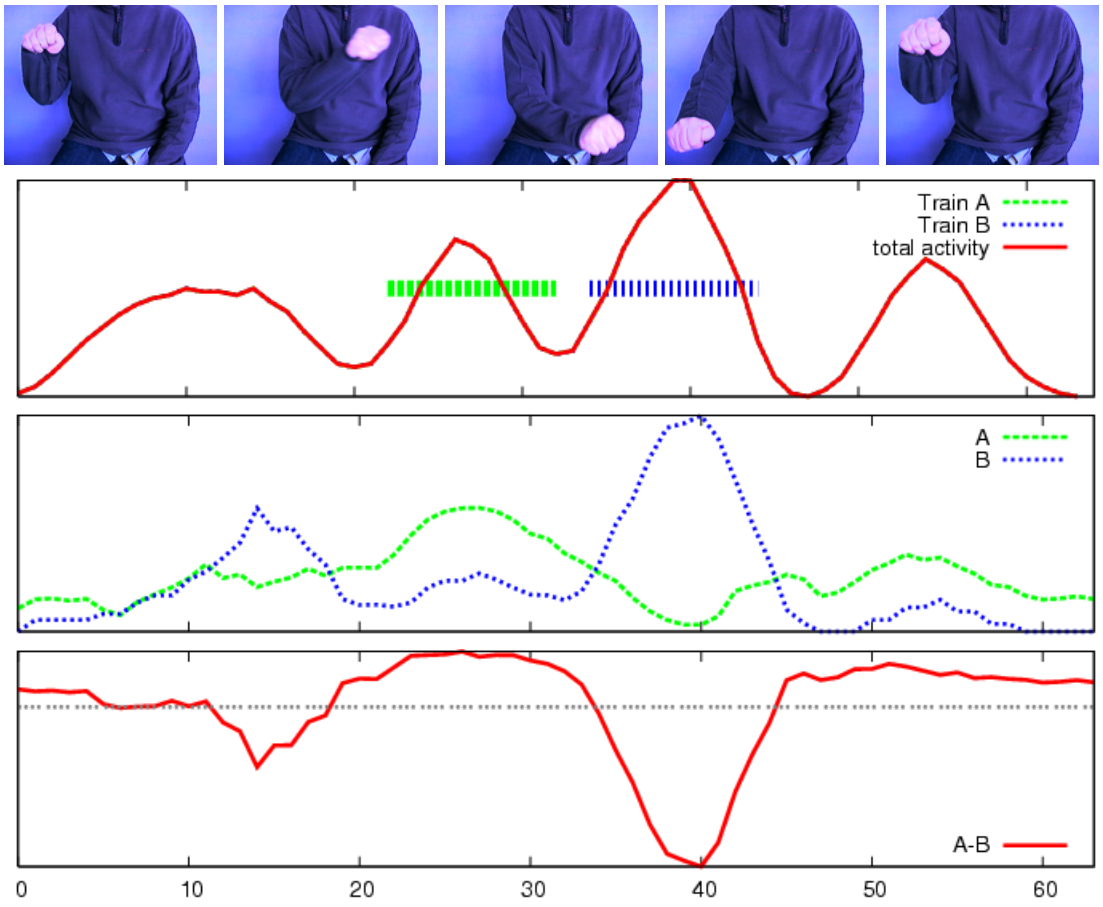


Figure 5.8: The rank voting approach. As in Figure 5.4, we show frames, the total level of activity along with the frames labelled in the index, the output of the rank voting algorithm, and also a comparison of the outputs for two semantics. It is clear that the method is able to identify frames from which it has been trained. Also, although the system was labelled with the "down" semantic (B), it has potentially identified the similar "up" semantic at $t \approx 17$.

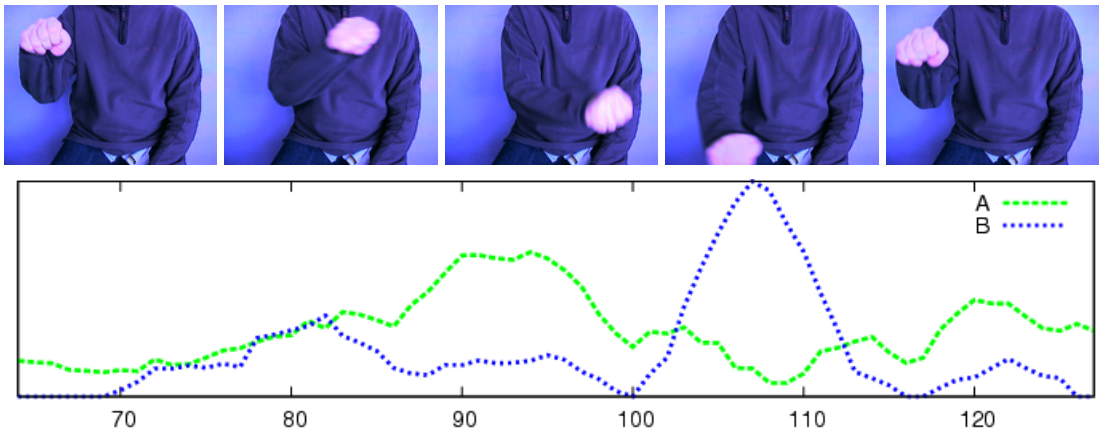


Figure 5.9: The rank voting algorithm result for a novel sequence. The labelled items from the sequence in Figure 5.8 are added to the index and the algorithm used to generate the semantic graph. Unlike in the competing models case, see Figure 5.5, the algorithm is able to generalise to novel data.

5.4 Modality fusion

In the previous Sections two approaches were described (Competing models, Rank voting) for estimating the semantic graphs $\mathcal{S}$. However, two important pragmatics have not been addressed:

- It may be computationally too expensive to interpret every frame. Rather, it is more tractable to only interpret sections that are considered interesting in some sense, e.g. they contain a level of activity above a threshold.
- The selection of the best interpretor configuration. Many potential configurations exist - Competing models vs Rank voting, with varying cell-size or similarity metric - each of which may be optimal for some situation. A one-shoe-fits-all approach is necessarily suboptimal. Each alternative is called a *modality*.

In this Section, a Bayesian network is used to obtain an optimal fused result using a number of modalities. To reduce the computational cost, a *constructive* algorithm is used for selective node population and inference.

5.4.1 Combination of evidence

It is widely understood in recognition and ranking research that an optimal solution is achieved by a statistical combination of evidence. For example, in the Inference Network retrieval model (Turtle, 1990; Graves, 2001), each word in a text query is probabilistically scored against a document, and all the word probabilities then combined using statistical approximations of Boolean operators. In (Sherrah and Gong, 2001; Toyama and Horvitz, 2000) a Bayesian fusion is performed to discover the location of faces in video, fusing the input from several separate feature detectors.

In our situation, we potentially possess a number of alternative semantic graphs, $\{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_*\}$, and we desire $\mathcal{S}_{optimal}$ that considers the computational cost $cost(\mathcal{S})$ and reliability $rel(\mathcal{S})$. We therefore propose using the Bayesian fusion framework shown in Figure 5.10. The main node to infer is the current level of belief in a semantic, S_t, with a prior provided by the the previous time-step, S_{t-1}.

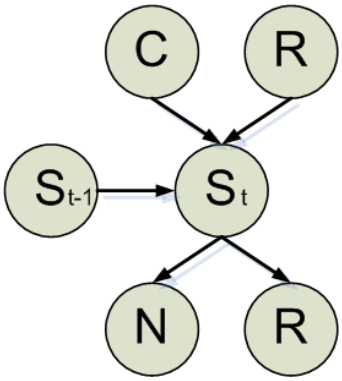


Figure 5.10: The Bayesian modality network used to perform fusion. The main node to infer is the level of semantic belief at the current time, S_t. This is "caused" by the previous time step S_{t-1} and contingent modalities C. The node is the "cause" of necessary modalities N. The R nodes represent a modality reliability. The network is applied to each time step to form a semantic graph using a principled modality fusion.

The modalities modeled in the network are either:

Contingent modalities are only indicative of potential semantic presence. For example the total scene activity is indicative of something happening but cannot provide information about what.

Necessary modalities must occur for the semantic to occur. For example, a Rank voting algorithm must have produced a good score.

In the fusion network contingent modalities are modeled with C nodes, parents of S_t, along with reliability node R. Necessary modalities are modeled with the N nodes, children of S_t, along with reliability R. In the Bayesian sense, the "semantic S_t causes the necessary evidence to occur". Figure 5.10 shows one C and one N node, however in general the number of modalities in the network is determined by the number of interpretors in the system.

We illustrate the fusion approach in Figure 5.11 compared to an additive approach (the modalities are summed). The total scene activity is used as a contingent modality and four separate rank-voting interpretors are used as the necessary modalities. The four rank-voting interpretors are configured with: $1:(\kappa=10,\lambda=32)$, $2:(\kappa=20,\lambda=32)$, $2:(\kappa=10,\lambda=16)$ and $4:(\kappa=20,\lambda=16)$, in order of increased computational cost. Figure 5.11 shows the result as more modalities are added to the computation. It is clear

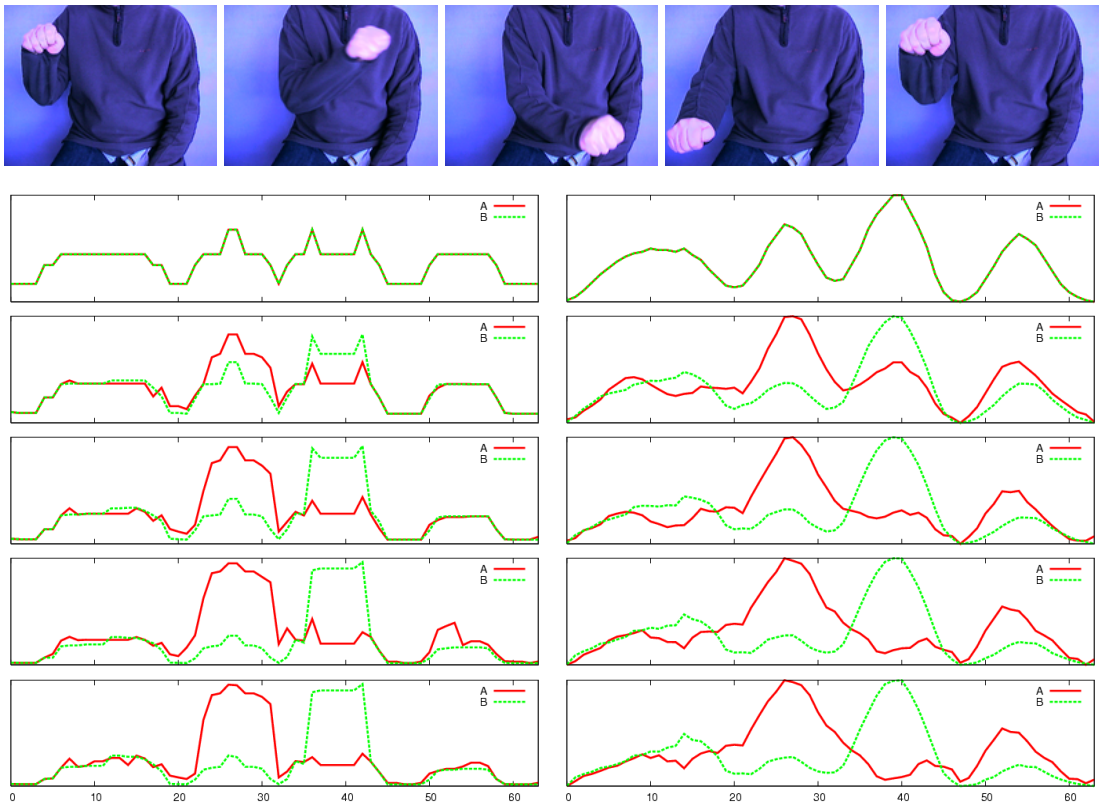


Figure 5.11: The semantic graph produced as more evidence is added. We compare the Bayesian fusion result with that produced by a straightforward additive approach using 1 contingent and 4 necessary modalities as described in the text. The fusion result can be seen to converge as more evidence is computed and added. This demonstrates that the fusion approach is performing a *combination of evidence*. Although the additive framework appears to give a good result, it essentially stabilised after a single modality meaning that the extra information was ignored.

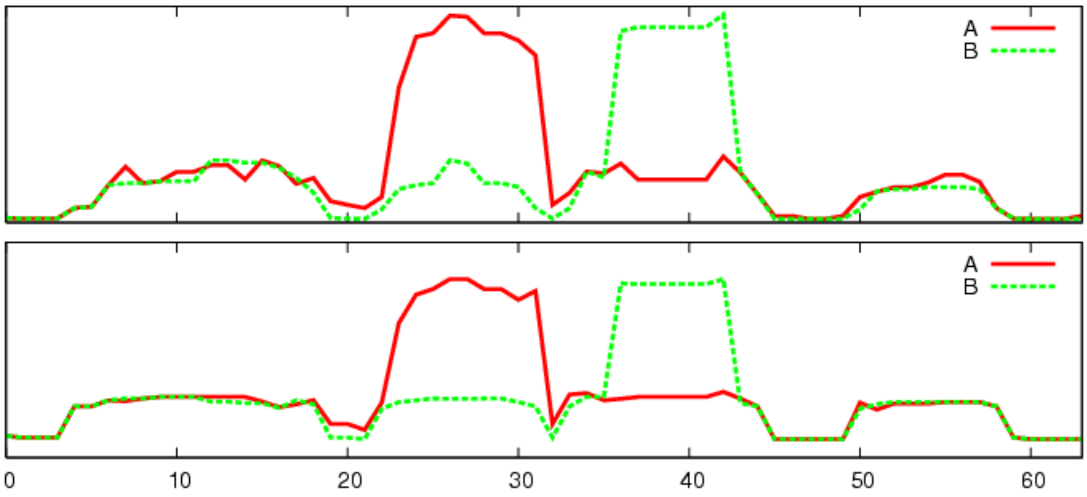


Figure 5.12: The constructive inference: (top) The semantic graph using all five modalities computed at every timestep, and (bottom) computed using the constructive algorithm. A similar result is achieved at greatly reduced computational cost.

that fusion approach converges to a result in comparison to the additive approach that does not consider the extra evidence.

5.4.2 An algorithm for constructive inference

An attractive property of Bayesian networks is that not all nodes need to be instantiated at a given time. This allows us to selectively add evidence to the network, in computational cost order, until a conclusive result is achieved. In the fusion network, the inexpensive modalities are computed and the respective nodes instantiated in the network. If the current level of semantic belief, $p(S_t = true)$, is not clear, a more expensive modality is computed and added to the network. This occurs, iteratively, until either all of the evidence is computed and nodes instantiated, or the level of belief is clear and stable. The approach reduces overall computational cost by only requiring the computation of the modalities needed to make the inference.

We use a method of adding a new modality only if the current belief is in the range $(0.2, 0.8)$, i.e. it is not clear. This range was chosen through experimentation and provides a good balance between stopping the computation (if sufficiently clear result) and continuing (if not clear). To illustrate the approach, constructive inference was performed with the five modalities of Section 5.4.1 in order to compute a semantic graph. The result, shown in Figure 5.12, is comparable to the non-constructive result, but was achieved at a much reduced computational cost:

Semantic A	[64 64 64 64 64]	[64 64 36 7 3]
Semantic B	[64 64 64 64 64]	[64 64 35 8 0]
	Normal	Constructive

The table shows, for each semantic and inference combination, the total number of modalities computed to form the semantic graphs. In other words, the constructive inference algorithm has reduced the number of feature computations and rank-voting graphs computed to achieve the result shown in Figure 5.12. For the normal case, all five modalities are computed for every frame ($t = 64$ is the length of the waving sequence). For the constructive case, the most expensive modality, $4 : (\kappa = 20, \lambda = 16)$, is only computed 3 times in total corresponding to significant savings in computational cost.

(a) The Tearoom scene showing “tea-making” activity.

(b) The Tearoom scene showing “jigsaw” activity.

Figure 5.13: Two semantic activities in an indoor tearoom scene. The tea-making activity shows a subject obtaining and filling the kettle, and generally hovering around the kitchen area of the tearoom. The jigsaw activity shows a subject participating in a group jigsaw on one of the foreground tables. Although the scene was staged indoors using a number of actors, the activities themselves are non-scripted and resemble natural action.

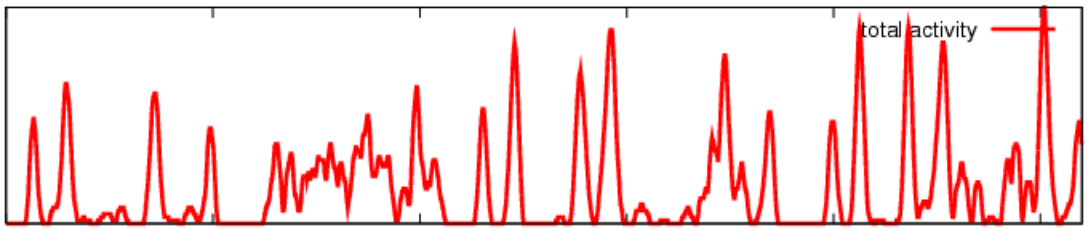


Figure 5.14: The total scene activity for the 2600 frames of the Tearoom1 sequence. It provides a useful pre-attentive indicator but no indication of semantic content.

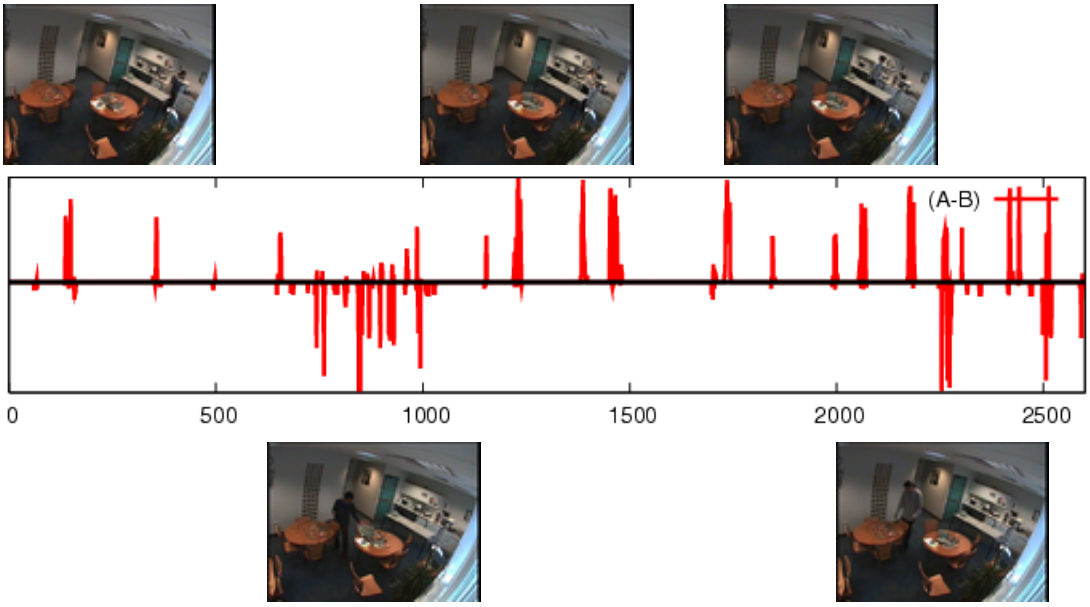


Figure 5.15: The fusion result computed using the constructive inference algorithm for sequence Tearoom1 showing an indoor tearoom scene with two semantics: A="making tea" and B="jigsaw". It can be seen that peaks and troughs are stimulated for similar content: we show the frames for positions $[150, 1230, 2060]$ and $[850, 2273]$. However, we find that the result for B is more distinctive than A as "tea-making" occurs in a spatially small area leading to few action features.

5.5 Experiments

We demonstrate Rank voting and Modality fusion using the tearoom scene and semantics shown in Figure 5.13. The scene is comprised of an indoor room, with seating and kitchen space, in which several actors participate in different semantic activities including: making tea, a communal jigsaw, sitting and talking, and also simulated theft. Although the scene content is simulated, it contains the variability of surveillance scenes and provides a controlled environment in which action can be controlled. We captured a sequence named Tearoom1 of length $t = 2600$ with each frame of spatial size (320×240) and performed indexing as follows:

- The Sustained temporal change of Chapter 3 was computed using the parameters developed previously, namely: $\alpha = \beta = 50$ and $T_{diff} = 5$.

- The Iconic index of Chapter 4 was computed using a variation of parameters $\lambda \in [16, 32]$ and $\kappa \in [10, 20]$ in order to permit alternative rank-voting modalities. Each frame is represented in each index using a histogram of iconic occurrence, as Equation (4.3). Note that although all the features were computed in advance during the experiment, a system would be optimised to compute as few as required.

- In order to perform Rank voting and Fusion, 3 frames were labelled for two semantics as follows:

Semantic A	$t = [140, 145, 150]$	The "tea-making" action semantic.
Semantic B	$t = [705, 742, 870]$	The "jigsaw" action semantic.

As previously described and illustrated in Section 5.4.1, a semantic graph was generated using the total scene activity as a contingent modality and four separate rank-voting interpretors as the necessary modalities. The total scene activity is computed as Equation (3.10) and the result shown in Figure 5.14. The four rank-voting interpretors, configured as described above, are constructively added to the fusion network in order of increased computational cost.

We show the fusion result using the constructive inference algorithm in Figure 5.15. The semantic graph shows the subtractive result, i.e. (Semantic A-Semantic B), as it

provides a visual indication of the occurrence of both semantics: peaks indicate A and troughs indicate B. A level graph indicates that the fusion result was similar for both and thus insufficient discriminant context is available to achieve a result. It can be seen in the Figure that occurrences of Semantic A and Semantic B are found using the graph. However, in the graph a number of extra peaks can be seen. It is surmised that this was because the "tea-making" activity occurs in a small spatial area, leading to few action features, and is not therefore distinctive enough from other activities to be found. In contrast, the result for Semantic B is clear and can be used for quick semi-semantic based browsing.

5.6 Discussion

An important problem for video search is to provide a browsing mechanism for sequence investigation. It is desirable to present a semantic approach, in that the user can interactively search for known contents. To this end, a *competing models* approach was examined with the supervised training of a number of Gaussian Mixture Models. It was found that, although the approach was able to recognise content in simple scenes, the extraction of feature sets and the vulnerability of the training process limit the potential scope. It is not desirable to train explicit content-recognition models before sequence investigation can begin. This requires knowledge of the information need in advance.

We presented our alternative *rank-voting* approach that exploited the presence of labelled items in a ranking. Using the current scene as the query input, labelled items vote for their semantic according to their rank position. The approach is intuitive, completely avoids the model fitting process, makes few assumptions about the scene content, uses limited manual labelling and training, and is able to be enhanced quickly and easily without complex model upheaval. However, it does require a number of items to be manually labelled and it requires the ranking process to be performed for each frame.

We recognise that that the configuration of semantic identification approach, for example by the selection of tuning parameters, is problematic. Therefore, a Bayesian fusion network was proposed that was able to reason about the results produced by several

interpretors. The use of such network provides a principled mathematical framework for performing a *combination of evidence* using the results of pre-attentive and semi-semantic cues as input. Also, it was proposed that the fusion was computed in a constructive manner - using estimates of cost and reliability - in order to generate the fused result but at reduced cost. It was found that the approach was able to provide a semantic graph for browsing a complex indoor scene.

Finally, we call our approach *Semi-semantic* because (a) the system is able to generate belief graphs for use during sequence investigation, but (b) these belief graphs do not correspond to system understanding of the content. The approach is dependent upon the user providing manual labels and having some understanding of the sequence content.

Chapter 6

Conclusions and Future Work

6.1 Motivation

It is apparent that as digital information is collected, in textual and visual domains, systems are required that are able to index and perform search. In the visual domain, this is particularly apparent for surveillance data as (a) it exists in huge quantities, and (b) real-time and retrospective scene investigation are the fundamental purpose for generation. Unfortunately, the semantic gap is also prevalent: in that the system representation has little or no correspondence with the semantic understanding of potential users.

In this thesis a framework has been presented for performing action-based scene indexing, as it is considered that "what is happening" in a surveillance scene is more important than "what is present". In particular, efficient pre-attentive cues - such as the level of scene action - have been used to reduce the scope of the search task. More specifically, the following problems have been addressed:

- Extraction of action features.

  Surveillance video is often long, is captured and stored using poor quality recording devices and storage media, and also contains content that is not visual distinctive. These are called the sensory gap and ambiguity issues (described in detail in Section 2.2). The extraction of useful pre-attentive action features from a scene is

a challenging and important issue. By pre-attentive, an approach must have little understanding of scene content during extraction.

- Forming an uncommitted representation.

  An index is formed off-line in advance of any potential retrospective search tasks. It is therefore critical that the video representation - i.e. the format of the index - does not make assumptions about the scene content. We call this desirable representation quality “uncommitted”. As sequences are typically long with little or no interesting content, the representation must also be compact and able to be searched using efficient tools.

- Temporal segmentation of surveillance video.

  For search tasks, the video frame does not provide a temporal context that corresponds to Human understanding of a scene. The automatic partitioning of video into larger contextual units - called “segments” - is considered a highly desirable feature of an indexing system. A challenging problem is to perform temporal segmentation using natural scene behaviours (rather than artificial behaviours often used in non-surveillance domains).

- Action-based conceptual visualisation.

  If a system is to provide quick access to video content, it should provide a conceptual visualisation that summarises the content. Video sequence visualisation is particularly important in surveillance video, as it reduces the the time required for manual video interpretation (in comparison to watching the sequence).

- Low-level integration of semantics.

  It is clear that a semantic-based search system is highly desirable. However, standard supervised training is not viable as it breaks the uncommitted requirement (described above). Alternatively, similar to relevance feedback in text-based search systems, semantics may become available during search. A desirable approach is able to integrate manual assessments to facilitate semi-semantic search.

6.2 Conclusions

6.2.1 Pre-attentive processing

In order to capture the required action features, the inexpensive thresholded temporal change measure of Equation (3.8) was adopted. In surveillance video, this approach is vulnerable to sensory problems that produce rogue active pixels. Therefore, a measure of Sustained temporal change was computed using a spatio-temporal smoothing process. Furthermore, the image space was divided into a grid of equally sized cells to capture a larger and more useful unit of action. Our experiments showed that the approach was able to extract reliable action information from the scene (see Figures 3.3 and 3.5) and also that a generic set of parameters could be used effective for a variety of scenes.

A sequence was indexed by detecting spatio-temporally connected regions of cells as meaningful independent activities using an adapted connected components algorithm. Activities were profiled to capture their spatial and temporal characteristics and similarity metrics defined. Our experiments showed that the approach was able to extract useful activity events from different outdoor scenes (see Figures 3.9 and 3.21).

A temporal segmentation for a sequence was computed using a measure of activity coherence. At each position in the index (frame or activity based), the past was compared to the future and those points of low coherence were detected as the breaks. Our experiments showed that the frame-based approach was vulnerable to window sizing issues, however the activity-based approach produced a result similar to manual segmentations (see Figures 3.16 and 3.17 on page 76).

6.2.2 Iconic indexing

In order to capture the the localised structure content of a cell, the Haar wavelet transform was computed and used to form a cell feature vector with the means of the coefficients in the different bands. When computed on the Sustained temporal change the features provide information on the directionality and visual appearance of action (see Figure 4.2). To provide an invariant mechanism for scene description, the coefficients were clustered and the centroids used to form an Iconic visual vocabulary. Each frame is then

described using the histogram of iconic occurrence as Equation (4.2).

It was found that the frame histograms remained stable for periods of similar action in the scene. Therefore, to perform temporal segmentation, a cumulative analysis was performed and used to generate a Video scene trajectory. The trajectory directionality remained constant if the scene content was not changing. It was found that a discretisation of the trajectory - i.e. finding the vertices at which the trajectory directionality changes - was able to partition a sequence into segments. Furthermore, in order to visualise the content of the segments, a measure of pixel discriminance was computed and the most discriminant active pixels superimposed onto a spatial context as a Segment summary frame. Experiments showed that the segmentation and visualisation approach was able to generate useful, repeating summaries of scene content in a variety of outdoor scenes (see Figures 4.13-4.18).

6.2.3 Semi-semantic analysis

In order to generate a semantic graph that shows the location of the occurrence of particular semantics in a sequence, competing probabilistic models were trained using Haar coefficients. It was found that such an approach was not able to correctly identify similar data in a novel scene, due to training and feature selection issues. In particular, we conclude that such an approach is not viable because it breaks the uncommitted index requirements - in other words, supervised training is required off-line with knowledge in advance of what is being searched. We conclude that a competing models paradigm is not sufficient for searching surveillance scenes.

To integrate manual semantic assessments, a novel Rank voting approach was proposed. The positions of the manually labelled items in a ranking are used to determine the content of the query item. It was found that, for constrained data, the approach was able to work effectively and generalise to novel data (see Figure 5.8). Furthermore, a Bayesian fusion framework was proposed to effectively perform a combination of pre-attentive evidence, and a constructive inference algorithm proposed in order to reduce computational load. Experiments showed that such a fusion approach was able to produce a useful semantic graph for a simulated indoor scene (see Figure 5.15).

6.3 Future Work

In this thesis pre-attentive, iconic and semi-semantic approaches have been presented for performing search of surveillance video. In particular, we have focused on the problems of temporal segmentation and content visualisation. However, the following problems remain to be solved.

The need for common evaluation frameworks

To perform evaluation of the temporal segmentation in Chapter 3, a small number of manual segmentations were obtained and used to illustrate the effectiveness of the approach. However, it is more desirable for a repeating large-scale data-oriented evaluation process that can be used for an objective assessment of approach performance. For example, in the text information retrieval community the establishment of the Text REtrieval Conference (TREC)[1] was highly influential in improving performance of retrieval algorithms, of unifying the research community, and demonstrating the importance of the work to external bodies. For the object detection and tracking paradigm in surveillance, the Performance Evaluation of Tracking and Surveillance workshop (PETS) provides a similar evaluation focus. A similar large-scale evaluation framework is desirable for non-tracking oriented video indexing and search tasks.

Adaptive iconic vocabularies

In the approach presented in Chapter 4 in this thesis, a scene is described using an Iconic visual vocabulary generated using Haar wavelet coefficients. To achieve improved performance for 24-hours-a-day surveillance video data, it would be beneficial to use a number of different vocabularies. For example, the current vocabulary could be optimised according to the current appearance of the scene (e.g. day or night). The result from different indices may be combined during the temporal segmentation and semantic integration tasks.

[1] See `http://trec.nist.gov`. "Its purpose was to support research within the information retrieval community by providing the infrastructure necessary for large-scale evaluation of text retrieval methodologies." TREC is co-sponsored by the National Institute of Standards and Technology (NIST) and U.S. Department of Defense.

Self tuning system

In this work, a number of subsystems were presented that require the selection of low-level parameters, for example the choice of α and β during the computation of sustained change in Chapter 3. To illustrate the ability of the approach in a variety of scenes, a single set of parameters were chosen that are considered to perform well. However, it may be preferable to learn these parameters over time for each scene.

Appendix A

Glossary

Bayesian network A mathematical framework for modelling uncertainty. A network consists of nodes that represent variables and arcs between them representing the dependence relationships.

Break point A point in a sequence at which it is considered that there is a change in the underlying semantic content. For example, in structured video, a break point would exist at positions at which one camera shot ended and another camera shot begins. The video section between two break points is known as a video segment. Discovered in the process of temporal segmentation.

Curse of dimensionality The exponential growth of hyper-volume as a function of dimensionality. Attributed to (Bellman, 1961). As a consequence, numerical methods perform poorly in high-dimensional feature spaces.

Dimensionality reduction An approach for reducing the number of dimensions in a feature space, while retaining is main character. One method is Principal components analysis.

Entropy A measure of information quality.

Expectation maximisation An iterative algorithm that maximises a likelihood function in order to fit a Gaussian mixture model to a feature space.

Feature extraction The computation of a numerical vector that represents content for some item being evaluated.

Gaussian mixture model A semi-parametric model that is able to approximate the probability density function of a feature space using a number of combined Gaussian distributions. Also known as a Mixture of Gaussians.

Ground truth A manual estimation of a result considered to be the most desirable output from an algorithm.

Iconic indexing As proposed in this thesis, an iconic index describes a video using a set of prototypical features obtained by clustering. The "icons" are the important elements in the scene.

Modality fusion A modality represents the belief in a semantic using a particular configuration. Modality fusion is a process concerned with performing a combination of evidence from several independent modalities.

Pre-attentive features The result of a feature extraction process that has no understanding of underlying content semantics and requires few computational resources

Ranking A list of items produced by a retrieval system in response to a query request. The query is compared to each item in an index using a similarity metric. The ranking is a list of item pointers ordered by similarity.

Relevance feedback The process, during the search, by which a user provides new information to the system on the quality of the current ranking. Thus permitting the system to learn online from examples in order to improve the result.

Semantic A meaning (that occurs in video). Corresponds to Human understanding and therefore subjective.

Semantic gap The common lack of coincidence between a digital video representation and human understanding of video content. A main problem in video indexing and retrieval systems. Whereas Human information need is semantic in nature, a system video representation is numerical (i.e. feature space).

Similarity metric A function that estimates the similarity between two items (usually, two items in the same video index). The fundamental building block of search systems.

Temporal segmentation The process in which the break points for a video are discovered. Also known as video partitioning or video structure discovery.

Video index A description of the video content that facilitates search applications. Constructed by the process of video parsing.

Video segment A number of consecutive frames that contain similar content. A semantic frame grouping.

Video summary A content abstraction using static frames. For example, a video summary could be constructed using the first frame of all the detected segments.

Video indexing The process of sequentially analysing a video to produce a video index. Also known as video parsing.

Wavelet transform A computation that localises a function in both space and scale. For example, in this thesis we use the Haar basis function for analysing regions of scene action.

Uncommitted (index) The requirement that a search index must not be constructed with advance knowledge about the searches that are to be performed. The index must be capable of accepting all search requests.

Appendix B

Normalisation of a series

Given a series of real valued data, $X = \{x_1, x_2, \ldots, x_N\}$, it is useful perform normalisation so that it has a mean $\mu = 0$ and is in the range $[-1, 1]$. This is achieved by computing the mean and variance, also called the first and second order moments, as:

$$\mu = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{B.1}$$

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^{N} (x_i - \mu)^2 \tag{B.2}$$

and then transforming each data value using:

$$x' = \frac{x - \mu}{3\sigma} \tag{B.3}$$

producing $X' = \{x'_1, x'_2, \ldots, x'_N\}$,

Bibliography

(Aigraine *et al.*, 1997) P. Aigraine, P. Joly, and V. Longueville. Medium knowledge-based macro-segmentation of video into sequences. In M.T. Maybury, editor, *Intelligent Multimedia Information Retrieval*, chapter 8, pages 159–173. AAAI/MIT Press, 1997.

(Arman *et al.*, 1993) F. Arman, A. Hsu, and M-Y Cheiu. Image processing on compressed data for large video databases. In *International conference on multimedia*, pages 267–272, 1993.

(Arman *et al.*, 1994) F. Arman, R. Depommier, A. Hsu, and M-Y. Chiu. Content-based browsing of video sequences. *ACM Multimedia*, pages 97–103, October 1994.

(Baeza-Yates and Ribeiro-Neto, 1999) R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.

(Bellman, 1961) R. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

(Beucher and Meyer, 1993) S. Beucher and S.F. Meyer. The morphological approach to segmentation: The watershed transformation. In E. Dougherty, editor, *Mathematical Morphology in Image Processing*, pages 433–481. Marcel Decker Inc., New York, 1993.

(Bimbo *et al.*, 2000) A. Del Bimbo, P. Pala, and L. Tanganelli. Video retrieval based on dynamics of color flows. In *International Conference on Pattern Recognition*, pages 851–854, Barcelona, September 2000.

(Bimbo, 1999) A. Del Bimbo. *Visual Information Retrieval*. Morgan Kaufmann Ed., San Francisco, USA, 1999.

(Bimbo, 2000) A Bimbo. Semantics based retrieval by content. In *IEEE International Conference on Image Processing*, volume 3, pages 516–519, Vancouver, September 2000.

(Bishop, 1995) C.M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, Oxford, England, 1995.

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

(Boreczky and Rowe, 1996) J.S. Boreczky and L.A. Rowe. Comparison of video shot boundary detection techniques. In *torage and Retrieval for Still Image and Video Databases*, pages 170–179, 1996.

(Brand *et al.*, 1997) M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 994–999, 1997.

(Buijs and Lew, 1999) J.M. Buijs and M.S. Lew. Visual learning of simple semantics in imagescape. In *Visual Information and Information Systems*, pages 131–138, 1999.

(Carson *et al.*, 1999) C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. In *Visual Information and Information Systems*, pages 509–516, Amsterdam, The Netherlands, 1999.

(Carson *et al.*, 2002) C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1026–1028, August 2002.

(Castelli and Bergman, 2002) V. Castelli and L. Bergman. *Image Databases: Search and Retrieval of Digital Imagery*. John Wiley and Sons, 2002.

(Chang *et al.*, 1998a) S.F. Chang, W. Chen, H.J. Meng, H. Sundaram, and D. Zhong. A fully automated content-based video search engine supporting spatiotemporal queries. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):602–615, September 1998.

(Chang *et al.*, 1998b) S.F. Chang, W. Chen, and H. Sundaram. Videoq: A fully automated video retrieval system using motion sketches. In *IEEE Workshop on Applications of Computer Vision*, pages 270–271, Princeton, October 1998.

(Chomat *et al.*, 2000) O. Chomat, J. Martin, and J.L. Crowley. A probabilistic sensor for the perception and recognition of activities. In *European Conference on Computer Vision*, volume 1, pages 487–503, Dublin, Ireland, June 2000.

(Colombo *et al.*, 1999) C. Colombo, A. Bimbo, and P. Pala. Semantics in visual information retrieval. *IEEE Multimedia*, 6(3):38–53, 1999.

(Combs and Bederson, 1999) T.T.A. Combs and B.B Bederson. Does zooming improve image browsing? In *ACM International Conference on Digital Libraries*, pages 130–137, 1999.

(Dee and Hogg, 2004) H.M. Dee and D.C. Hogg. Detecting inexplicable behaviour. In *British Machine Vision Conference*, pages 477–486, Kingston, 2004.

(DeMenthon *et al.*, 2000) D. DeMenthon, L.J. Latecki, A. Rosenfeld, and M.V. Stückelberg. Relevance ranking of video data using hidden markov model distances and polygon simplification. In *Advances in Visual Information Systems*, pages 49–61, Lyon, November 2000.

(Dempster *et al.*, 1977) A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, B-39(1):1–38, 1977.

(Donald, 1999) C.H.M. Donald. Assessing the human vigilance capacity of control room operators. In *International Conference on Human Interfaces in Control Rooms, Cockpits and Command Centres*, pages 7–11, 1999.

(Eakins *et al.*, 2004) J.P Eakins, P. Briggs, and B. Burford. Image retrieval interfaces: a user perspective. In *International Conference on Image and Video Retrieval*, pages 628–637, Dublin, Ireland, 2004.

(Eakins, 1996) J.P Eakins. Automatic image content retrieval - are we getting anywhere? In *Proceedings of Third International Conference on Electronic Library and Visual Information Research*, pages 123–135, 1996.

(Enser and Sandom, 1995) P.G.B. Enser and C. Sandom. Pictorial information retrieval. In *Journal of Documentation*, volume 51, pages 126–170, 1995.

(Enser and Sandom, 2003) P.G.B. Enser and C. Sandom. Towards a comprehensive survey of the semantic gap in visual image retrieval. In *International Conference on Image and Video Retrieval*, pages 291–299, Urbana-Champaign, USA, 2003.

(Feng *et al.*, 2002) X. Feng, C.K.I. Williams, and S.N. Felderhof. Combining belief networks and neural networks for scene segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):467–483, April 2002.

(Ferryman, 2003) J.M. Ferryman, editor. *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*. Reading, October 2003.

(Flickner *et al.*, 1995) M. Flickner, H. Sawhney, and W. Niblack. Query by image and video content: The QBIC system. *IEEE Computer*, 28:23–32, September 1995.

(Gombrich, 1995) E. Gombrich. *The Story of Art*. Phaidon, New York, 1995.

(Gong and Xiang, 2003) S. Gong and T. Xiang. Recognition of group activities using dynamic probabilistic networks. In *IEEE International Conference on Computer Vision*, pages 742–749, Nice, France, October 2003.

(Gong *et al.*, 2000) S. Gong, S. McKenna, and A. Psarrou. *Dynamic Vision: From Images to Face Recognition*. Imperial College Press, London, England, 2000.

(Gonzalez and Woods, 1992) R. Gonzalez and R. Woods. *Digital Image Processing*, chapter 2. Addison-Wesley, 1992.

(Graps, 1995) A. Graps. An introduction to wavelets. *IEEE Computational Science and Engineering*, 2(2):50–61, 1995.

(Graves and Lalmas, 2002) A.P. Graves and M. Lalmas. Video retrieval using an MPEG-7 based inference network. In *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 339–346, Tampere, Finland, August 2002.

(Graves, 2001) Andrew P. Graves. Video indexing and retrieval using an MPEG-7 based inference network. Master's thesis, Queen Mary, University of London, 2001.

(Greenhill *et al.*, 2004) D. Greenhill, G. Rendall, J. Orwell, and G.A. Jones. Occlusion analysis: Learning and utilising depth maps in object tracking. In *British Machine Vision Conference*, pages 467–476, Kingston, 2004.

(Gunsel *et al.*, 1997) B. Gunsel, Y. Fu, and A.M. Tekalp. Hierarchical temporal video segmentation and content characterization. *Multimedia Storage and Archiving Systems II, SPIE*, 3229:46–55, 1997.

(Hampapur *et al.*, 1995) A. Hampapur, R.C. Jain, and T. Weymouth. Production model based digital video segmentation. *International journal of multimedia tools and applications*, 1(1):9–46, 1995.

(Hanjalic *et al.*, 1999) A. Hanjalic, R. L. Lagendijk, and J. Biemond. Automated high-level movie segmentation for advanced video-retrieval systems. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(4):580–588, June 1999.

(Hauptmann and Smith, 1995) A. Hauptmann and M. Smith. Text, speech, and vision for video segmentation: The informedia project. In *AAAI Fall 1995 Symposium on Computational Models for Integrating Language and Vision*, pages 90–95, Boston, 1995.

(Heesch and Rüger, 2004) D. Heesch and S. Rüger. NN^k networks for content-based image retrieval. In *European Conference on Information Retrieval*, pages 253–266, Sunderland, April 2004.

(Heesch and Rüger, 2005) D. Heesch and S. Rüger. Image browsing: Semantic analysis of NN^k networks. In *International Conference on Image and Video Retrieval*, pages 609–618, Singapore, July 2005.

(Isard and Blake, 1998) M. Isard and A. Blake. CONDENSATION – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

(Iyengar and Lipman, 2000) G. Iyengar and A.B. Lipman. Content-based browsing and editing of unstructured video. In *IEEE International Conference on Multimedia & Expo*, pages 159–162, 2000.

(Jain and Vailaya, 1995) A.K. Jain and A. Vailaya. Image retrieval using color and shape. In *Asian Conference on Computer Vision*, pages 529–533, Singapore, December 1995.

(Kender and Yeo, 1998) J.R. Kender and B.L. Yeo. Video scene segmentation via continuous video coherence. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 367–373, Santa Barbara, June 1998.

(Keogh *et al.*, 2001) E. Keogh, S. Chu, D. Hart, and M. Pazzani. An online algorithm for segmenting time series. In *International Conference on Data Mining*, pages 289–297, San Jose, California, 2001.

(Kikukawa and Kawafuchi, 1992) T. Kikukawa and S. Kawafuchi. Development of an automatic summary editing system for the audio-visual resources. *Transactions on Electronics and Information*, 75(A):204–212, 1992.

(Lai and Tait, 1999) T.S. Lai and J. Tait. CHROMA: A content-based image retrieval system. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 324, Berkely, California, January 1999.

(Lefèvre *et al.*, 2003) S. Lefèvre, J Holler, and N. Vincent. A review of real-time segmentation of uncompressed video sequences for content-based search and retrieval. *Real-Time Imaging*, 9(1):73–98, 2003.

(Lew and Sebe, 2000) M.S. Lew and N. Sebe. Visual websearching using iconic queries. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 788–789, Hilton Head Island, USA, 2000.

(Lew, 2000) M.S. Lew. Next generation web searches for visual content. In *IEEE Computer*, pages 46–53, November 2000.

(Lew, 2001) M.S. Lew. *Principles of Visual Information Retrieval*. Springer Verlag, 2001.

(Li *et al.*, 2001) Y. Li, T. Zhang, and D. Tretter. An overview of video abstraction techniques. Technical report, HP Laboratory, July 2001.

(Lienhart, 1996) R. Lienhart. Automatic text recognition for video indexing. *IEEE Multimedia*, pages 11–20, 1996.

(Lin and Zhang, 2000) T. Lin and H.J. Zhang. Automatic video scene extraction by shot grouping. In *International Conference on Pattern Recognition*, volume 4, pages 39–42, Barcelona, Spain, September 2000.

(Lin *et al.*, 2001) T. Lin, H.J. Zhang, and Q.Y. Shi. Video scene extraction by force competition. In *IEEE International Conference on Multimedia & Expo*, Tokyo, Japan, August 2001.

(Longuet-Higgins, 1984) H.C. Longuet-Higgins. The visual ambiguity of a moving plane. In *Proceedings of the Royal Society of London (B, Biological Sciences)*, volume 223, pages 165–175, London, 1984.

(Lui, 2002) T.Y. Lui. An object oriented image retrieval system. Master's thesis, Queen Mary, University of London, 2002.

(Ma and Zhang, 2000) W.Y. Ma and H.J. Zhang. An indexing and browsing system for home video. In *European Conference on Signal Processing.*, pages 131–134, Patras, Greece, September 2000.

(Makris and Ellis, 2002) D. Makris and T. Ellis. Path detection in video surveillance. *Image and Vision Computing*, 20(12):895–903, October 2002.

(Maybury, 1997) M. Maybury. *Intelligent Multimedia Information Retrieval*. MIT Press, Cambridge, US, 1997.

(McCahill and Norris, 2002) M. McCahill and C. Norris. CCTV in London. Technical report, Urban Eye Project, Working Paper No.6, June 2002.

(McKenna *et al.*, 1999) S. McKenna, Y. Raja, and S. Gong. Tracking colour objects using adaptive mixture models. *Image and Vision Computing*, 1(17):225–231, 1999.

(Meng *et al.*, 1995) J. Meng, Y. Juan, and S.F. Chang. Scene change detection in a MPEG compressed video sequence. *IS&T/SPIE Symposium Proceedings on Electronic Imaging: Science & Technology*, 2419:14–25, February 1995.

(Mills *et al.*, 1992) M. Mills, J. Cohen, and Y-Y. Wong. A magnifier tool for video dat. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 93–98, Monterey, CA, May 1992.

(Nagasaka and Tanaka, 1991) A. Nagasaka and Y. Tanaka. Automatic video indexing and full-motion search for object appearances. In *IFIP Working conference on visual database systems*, pages 113–127, Budapest, Hungary, September 1991.

(Nait-Charif and McKenna, 2004) H. Nait-Charif and S.J. McKenna. Activity summarisation and fall detection in a supportive home environment. In *International Conference on Pattern Recognition*, pages 323–326, Cambridge, 2004.

(Ng, 2002) J. Ng. *Learning Temporal Models for Interpreting Dynamic Scenes*. PhD thesis, Queen Mary, University of London, 2002.

(Ngo *et al.*, 2002) C.W. Ngo, T.C. Pong, and H.J. Zhang. Motion-based video representation for scene change detection. *International Journal of Computer Vision*, 50(2):127–142, November 2002.

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

(Oren *et al.*, 1997) M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 193–199, Puerto Rico, June 1997.

(Porter *et al.*, 2003a) S. V. Porter, M. Mirmehdi, and B. T. Thomas. Temporal video segmentation and classification of edit effects. *Image and Vision Computing*, 21(13-14):1097–1106, December 2003.

(Porter *et al.*, 2003b) S.V. Porter, M. Mirmehdi, and B.T. Thomas. Video indexing using motion estimation. In *British Machine Vision Conference*, pages 659–668, Norwich, England, September 2003.

(Press *et al.*, 1992) W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 2nd edition, 1992.

(Psarrou *et al.*, 2002) A. Psarrou, S. Gong, and M. Walter. Recognition of human gestures and behaviour. *Image and Vision Computing*, 20(5):349–358, 2002.

(QMUL, 2002) QMUL. Incident recognition for surveillance and security. Technical report, QMUL, June 2002.

(Queries, 2002) ImageScape Visual Queries. http://skynet.liacs.nl/, 2002.

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

(Rijsbergen, 1979) C. J. Van Rijsbergen. *Information Retrieval*. Butterworths, 1979.

(Rodden *et al.*, 2001) K. Rodden, W. Basalaj, D. Sinclair, and K.R. Wood. Does organisation by similarity assist image browsing? In *ACM Conference on Human Factors in Computing Systems*, pages 190–197, Seattle, April 2001.

(Rubner and Tomasi, 1999) Y. Rubner and C. Tomasi. Texture-based image retrieval without segmentation. In *IEEE International Conference on Computer Vision*, volume 2, pages 1018–1024, Corfu, Greece, September 1999.

(Rubner *et al.*, 2000) Y. Rubner, C. Tomasi, and L.J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

(Rui and Huang, 2000) Y. Rui and T. Huang. A unified framework for video browsing and retrieval. In A. Bovik, editor, *Image and Video Processing Handbook*, pages 705–715. New York: Academic, 2000.

(Rui *et al.*, 1998) Y. Rui, S. Huang, and S. Mehrota. Exploring video structures beyond the shots. In *IEEE International Conference on Multimedia Computing and Systems*, pages 237–240, Austin, Texas, July 1998.

(Salton, 1989) Gerard Salton. *Automatic Text Processing – The Transformation, Analysis, and Retrieval of Information by Computer*. Addison–Wesley, 1989.

(Sawhney and Ayer, 1996) H.S. Sawhney and S. Ayer. Compact representations of videos through dominant and multiple motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):814–830, 1996.

(Sebestyen, 1962) G.S. Sebestyen. *Decision-Making Processes in Pattern Recognition*. Macmillan, New Yotk, 1962.

(Shahraray, 1995) B. Shahraray. Scene change detection and content-based sampling of video sequences. *SPIE Digital Video Compression, Algorithm and Technologies*, 2419:2–13, 1995.

(Sherrah and Gong, 2001) J. Sherrah and S. Gong. Continuous global evidence-based bayesian modality fusion for simultaneous tracking of multiple objects. In *IEEE International Conference on Computer Vision*, pages 42–49, Vancouver, 2001.

(Shi and Malik, 2000) J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.

(Sivic and Zisserman, 2003) J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision*, pages 470–1477, Nice, October 2003.

(Smeaton, 2001) A.F. Smeaton. Content-based access to digital video: The físchlár system and the TREC video track. In *Joint National Science Foundation/INRIA/IBM/University of California Berkeley Workshop on Multimedia Content Based Indexing and Retrieval (MMCBIR2001)*, INRIA, Rocquencourt, 2001.

(Smeulders *et al.*, 2000) A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

(Smith and Chang, 1997) J.R. Smith and S.F. Chang. VisualSEEK: A fully automated content-based image query system. In *ACM Multimedia*, pages 87–98, Boston, 1997.

(Srinivasan *et al.*, 1999) S. Srinivasan, D.B. Ponceleon, A. Amir, and D. Petkovic. "what is in that video anyway?" in search of better browsing. In *IEEE International Conference on Multimedia Computing and Systems*, volume 1, pages 388–393, Florence, Italy, June 1999.

(Stauffer and Grimson, 2000) C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–757, 2000.

(Sundaram and Chang, 2000) H. Sundaram and S.F. Chang. Video scene segmentation

using video and audio features. In *IEEE International Conference on Multimedia and Expo*, pages 1547–1550, New York, August 2000.

(Swain and Ballard, 1991) M.J. Swain and D.H. Ballard. Color indexing. *International Journal of Computer Vision*, 7:11–32, 1991.

(Tonomura and Abe, 1990) Y. Tonomura and S. Abe. Content-oriented visual interface using video icons for visual database systems. *Journal of Visual Languages and Computing*, 1:183–198, 1990.

(Toyama and Horvitz, 2000) E. Toyama and E. Horvitz. Bayesian modality fusion: Probabilistic integration of multiple vision algorithms for head tracking. In *Asian Conference on Computer Vision*, Tapei, Taiwan, January 2000.

(Turtle, 1990) H.R. Turtle. *Inference Networks for Document Retrieval*. PhD thesis, University of Massachusetts, 1990.

(Uchihashi, 1999) S. Uchihashi. Video Manga: Generating semantically meaningful video summaries. In *Proc. ACM Multimedia 99, Orlando, FL, Nov.*, pages 383–292, 1999.

(Ullman, 2000) S. Ullman. *High-level Vision: Object Recognition and Visual Cognition*. MIT Press, 2000.

(Unser, 1995) M. Unser. Texture classification and segmentation using wavelet frames. *IEEE Transactions on Image Processing*, 4(11):1549–1560, November 1995.

(Vailaya *et al.*, 2001) A. Vailaya, M. Figueiredo, A.K. Jain, and H.J. Zhang. Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10(1):117–130, January 2001.

(Vinod, 1998) V.V. Vinod. Activity based video shot retrieval and ranking. In *International Conference on Pattern Recognition*, pages 682–684, Brisbane, August 1998.

(Wang *et al.*, 2001) J.Z. Wang, J. Li, and G. Wiederhold. Simplicity: Semantics sensitive integrated matching for picture libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):947–963, 2001.

(Weisstein, 2006a) E.W. Weisstein. Kurtosis. From MathWorld–A Wolfram Web Resource, http://mathworld.wolfram.com/Kurtosis.html, 2006.

(Weisstein, 2006b) E.W. Weisstein. Statistical median. From MathWorld–A Wolfram Web Resource, http://mathworld.wolfram.com/StatisticalMedian.html, 2006.

(Welsh and Farrington, 2002) B.C. Welsh and D.P. Farrington. Crime prevention effects of closed circuit television: a systematic review. Technical report, Home Office Research, Study 252, August 2002.

(Wolf, 1996) W. Wolf. Key frame selection by motion analysis. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 1228–1231, 1996.

(Xiang and Gong, 2006) T. Xiang and S. Gong. Beyond tracking: Modelling activity and understanding behaviour. *International Journal of Computer Vision*, 1(67):21–51, 2006.

(Xie *et al.*, 2003) L. Xie, S.-F. Chang, A. Divakaran, and H. Sun. Unsupervised mining of statistical temporal structures in video. In A. Rosenfeld, D. Doremann, and D. Dementhon, editors, *Video Mining*, chapter 10, pages 279–308. Kluwer Academic Publishers, June 2003.

(Yeung *et al.*, 1996) M. Yeung, B.L. Yeo, and B. Liu. Extracting story units from long programs for video browsing and navigation. In *IEEE International Conference on Multimedia Computing and Systems*, pages 296 – 305, June 1996.

(Zeng *et al.*, 2002) W. Zeng, W. Gao, and D. Zhao. Video indexing by motion activity maps. In *IEEE International Conference on Image Processing*, Rochester, September 2002.

(Zhang *et al.*, 1993) H. Zhang, A. Kankamhalli, and S. Smoliar. Automatic partitioning of full-motion video. *Multimedia Systems*, 1:10–28, 1993.

(Zhang *et al.*, 1994) H.J. Zhang, C.Y. Low, Y.H. Gong, and S.W. Smoliar. Video parsing using compressed data. In *SPIE Conference on Image and Video Processing*, pages 142–149, 1994.

(Zhao *et al.*, 2000) L. Zhao, W. Qi, S.Z. Li, S.Q. Yang, and H.J. Zhang. Key-frame extraction and shot retrieval using Nearest Feature Line. In *IW-MIR, ACM MM*, pages 217–220, 2000.

(Zhong *et al.*, 2004) H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 819–826, Washington, 2004.

(Zhuang *et al.*, 1998) Y. Zhuang, Y. Rui, T.S. Huang, and S. Mehrotra. Adaptive key frame extraction using unsupervised clustering. In *IEEE International Conference on Image Processing*, volume 1, pages 866–870, Chicago, October 1998.