Highly Efficient Low-Level Feature Extraction For Video Representation And Retrieval

Janko Ćalić

Submitted for the Degree of

Doctor of Philosophy

Department of Electronic Engineering,

Queen Mary, University of London

2004

to Jelena

ABSTRACT

Witnessing the omnipresence of digital video media, the research community has raised the question of its meaningful use and management. Stored in immense multimedia databases, digital videos need to be retrieved and structured in an intelligent way, relying on the content and the rich semantics involved. Current Content Based Video Indexing and Retrieval systems face the problem of the semantic gap between the simplicity of the available visual features and the richness of user semantics.

This work focuses on the issues of efficiency and scalability in video indexing and retrieval to facilitate a video representation model capable of semantic annotation. A highly efficient algorithm for temporal analysis and key-frame extraction is developed. It is based on the prediction information extracted directly from the compressed-domain features and the robust scalable analysis in the temporal domain. Furthermore, a hierarchical quantisation of the colour features in the descriptor space is presented. Derived from the extracted set of low-level features, a video representation model that enables semantic annotation and contextual genre classification is designed.

Results demonstrate the efficiency and robustness of the temporal analysis algorithm that runs in real time maintaining the high precision and recall of the detection task. Adaptive key-frame extraction and summarisation achieve a good overview of the visual content, while the colour quantisation algorithm efficiently creates hierarchical set of descriptors. Finally, the video representation model, supported by the genre classification algorithm, achieves excellent results in an automatic annotation system by linking the video clips with a limited lexicon of related keywords.

ACKNOWLEDGEMENTS

The completion of my thesis would not have been possible without the strong support of a number of people.

Firstly, I would like to thank my supervisor Dr Ebroul Izquierdo for giving me irreplaceable guidelines both professionally and personally, and for his exceptional determination and unique personality.

Furthermore, I would like to thank the whole Electronic Engineering Department at Queen Mary, University of London, especially Prof. Laurie Cuthbert, Mrs Lynda Rolfe, Mr Phil Willson and Mr Kok Ho Huen for generous help and support throughout my PhD, as well as the whole crew from the Multimedia and Vision Research Lab, particularly: Dr Alan Pearmain, Mr Ivan Damnjanović, Mr Andres Dorado Alban, Mr Miguel Tavares Coimbra, Miss Maria Trijillo Uribe; Mr Ivan Đorđević from the Intelligent Communications Lab and specially Dr Veselin Rakočević for his continuous help and presence.

Above all, I would like to thank my family, my wife Jelena, my parents Jovo and Eržebet and my brother Čaba for love and support throughout my studies.

TABLE OF CONTENT

Highly Efficient Low-Level Feature Extraction For Video Representation And Retrieval.......1
Abstract 3
Acknowledgements..........4
Table of Content..........5
List of Figures..........9
List of Abbreviations..........12
I. Introduction..........13
I.1. Prologue 13
I.2. Problem 13
I.3. Research Scope..........16
I.4. Project Specification and Objectives 17
I.5. Research Contributions..........18
I.6. Structure 19
II. Content Based Video Indexing and Retrieval..........21
II.1. Overview..........21
II.2. Video Indexing and Retrieval..........22
II.2.1. Visual Media: Our scope..........22
II.2.2. Three generations of CBVIR 23
II.3. Semantic Gap..........25
II.3.1. The Question of Meaning: From Query by Example to Semantic Retrieval..........26
II.4. Knowledge Representation for Video 27
II.5. Computational Media Aesthetics..........29
II.5.1. Background 29
II.5.2. Overview of CMA 30
II.5.3. Novelty and Contribution of CMA..........31
II.6. Context in CBVIR..........31
II.6.1. Human subject : Importance of its presence..........32
II.6.2. User Relevance Feedback 33
II.6.3. Genre Theory..........34
II.7. Summary..........35
III. Current Tendencies in CBVIR 36
III.1. Overview..........36
III.2. MPEG Video Compression Standards 36
III.2.1. Fundamentals of MPEG Video Compression 36
III.2.2. The MPEG Video Coder Source Model 37
III.2.3. Subsampling and Interpolation 38

III.2.4. Motion Compensated Prediction 38
III.2.5. Transform Domain Coding 40
III.2.6. MPEG-1 41
III.2.7. The Basic MPEG-1 Inter-Frame Coding Scheme 42
III.2.8. Specific Functionalities 45
III.2.9. MPEG-2 Standard Overview 47
III.3. Temporal Video Feature Extraction 48
III.3.1. Temporal Video Segmentation in Uncompressed Domain 49
III.3.2. Clustering-Based Temporal Video Segmentation 57
III.3.3. Feature Based Temporal Video Segmentation 58
III.3.4. Model Driven Temporal Video Segmentation 59
III.3.5. Temporal Video Segmentation in MPEG Compressed Domain 61
III.4. Key-frame extraction Techniques 74
III.5. Representing Colour in CBVIR 78
III.5.1. Colour Histograms 78
III.5.2. Colour Histogram Comparison 79
III.5.3. Selection of Colour Space 80
III.5.4. Colour Space Quantisation 82
III.5.5. Histogram Quantisation 83
III.6. Semantic efforts in CBVIR 85
III.6.1. Semiotic Theories 86
III.6.2. Semiotics in CBVIR 88
III.6.3. Relevance feedback 89
III.7. Representations of Videos in CBVIR 90
III.7.1. Expressive Elements used in Video Representations 91
III.8. Genre classification 96
IV. Efficient Low-level Feature Extraction 99
IV.1. Overview 99
IV.2. Metric extraction 99
IV.2.1. Prediction Information in the MPEG stream 99
IV.2.2. MPEG Sequence structure 101
IV.2.3. Frame Difference Metric within one SGOP 101
IV.2.4. Metric Extension for Gradual Change Detection 104
IV.2.5. Low-pass filtering of the Metric 107
IV.2.6. Generalized Frame Difference Metrics 108
IV.2.7. Curve Smoothing 110
IV.3. Metric Simplification 111
IV.3.1. Relevance Order 112

IV.3.2. Relevance Measure ... 113

IV.4. Motion flow extraction ... 114

IV.4.1. Motion Vectors as Optical Flow Approximation ... 114

IV.4.2. Flow Estimation ... 114

IV.4.3. Camera Motion Analysis ... 116

IV.5. Key-frame extraction ... 117

IV.6. Colour histogram quantisation ... 118

V. Video Representation Model ... 122

V.1. Overview ... 122

V.2. Introduction ... 122

V.3. Video Representation Model ... 123

V.3.1. Representation Objectives ... 123

V.3.2. Proposed Representation Model ... 125

V.4. Genre Classification ... 132

V.4.1. Classification method ... 132

V.4.2. K-means clustering ... 133

V.5. Summary ... 136

VI. Experimental Testbed ... 137

VI.1. Overview ... 137

VI.2. Experimental Dataset ... 137

VI.3. Automatic Video Annotation system ... 138

VI.3.1. Annotation System Overview ... 138

VI.3.2. The Learning Unit ... 141

VI.3.3. Automatic Video Annotation ... 143

VI.4. Summary ... 144

VII. Results ... 145

VII.1. Overview ... 145

VII.2. Metric Extraction ... 145

VII.2.1. Frame-to-frame Difference Metrics ... 145

VII.2.2. Shot Detection Evaluation Procedure ... 146

VII.2.3. Random Distance Difference Metrics ... 147

VII.2.4. Results for the Generalized Difference Metric ... 149

VII.3. Motion Flow and Camera Work Analysis ... 154

VII.4. Key-frame Extraction ... 155

VII.5. Hierarchical colour histogram quantisation ... 157

VII.6. Genre Classification using k-means algorithm ... 158

VII.7. Representation Model for Automatic Annotation ... 159

VII.8. Automatic Annotation Evaluation ... 161

VII.9. System Demonstrators 162
VII.10. Summary 164
VIII. Discussion and Conclusions 165
VIII.1. Discussion 165
VIII.2. Future Work 167
VIII.3. Conclusions 168
IX. References 169
X. Annexes 178
X.1. List of Author's Publications 178
X.2. GANTT Charts 179
X.3. Summary of Used video sequences 180

LIST OF FIGURES

Figure I.1 Development Timeline of Multimedia Technologies..14
Figure II.1 Scheme of a common CBVIR System 24
Figure II.2 Bridging the gap between user's query concept and metadata representation..........26
Figure II.3 Representation as a part of the processing chain 28
Figure II.4 Representation looking glass...28
Figure III.1 Motion Compensation in MPEG stream...39
Figure III.2 MPEG sequence decomposition..43
Figure III.3 Block diagram of a basic MPEG encoder structure..43
Figure III.4 a) Frame at time instance *N* to be coded. 45
Figure III.5 I-pictures (I), P-pictures (P) and B-pictures (B) 46
Figure III.6 Non-overlapping square blocks in net comparison algorithm 52
Figure III.7 Twin comparison: a. consecutive, b. accumulated histogram differences. 55
Figure III.8 A full resolution image and its DC image...64
Figure III.9 Gradual transition g_n and pixel difference model $D_{gn}(l,l+k)$ in dissolve det. 65
Figure III.10 DC term estimation in the method of Shen and Delp 66
Figure III.11 Histogram difference diagram (*:cut, ---:dissolve)..67
Figure III.12 Video shot detection scheme of Patel and Sethi...69
Figure III.13 Shot detection algorithm of Meng, Juan and Chang..69
Figure III.14 Cuts: a) video structure, b) number of intra coded MBs..71
Figure III.15 Fade out, fade in, dissolve: a) video structure, b) No. of intra coded MBs...........72
Figure III.16 Three frames showing an example of a zoom-in sequence 76
Figure III.17 Three frames depicting a trolley-bus entering the scene. 77
Figure III.18 Sample Colour Image suzie.jpg...78
Figure III.19 Colour histogram of the image above 78
Figure III.20 Colour circle..81
Figure III.21 Colour cone...81
Figure III.22 A) Basic unit of Hadamard transform, B) A schematic diagram............................84
Figure III.23 Semiotic framework for multimedia and features extracted from multimedia.....88
Figure III.24 Camera Motion Types...93
Figure III.25 Camera zoom operation and corresponding motion vectors 94
Figure III.26 Pan operation and corresponding motion vectors 95
Figure IV.1 Possible positions of the cut in a frame triplet.. 102
Figure IV.2 Difference metric example.. 103
Figure IV.4 Structure of two frame triples 106
Figure IV.5 Gaussian Smoothing of the Metric Curve.. 107
Figure IV.6 Content change in a frame triple.. 109

Figure IV.7 Belll shaped pulse response of the applied Gaussian filter 110
Figure IV.8 Noise suppression 110
Figure IV.9 Flowchart of the DCE algorithm 111
Figure IV.10 Relevance order examples 112
Figure IV.11 DCE linearization of two adjacent line segments 113
Figure IV.12 Flowchart of the flow vectors extrapolation procedure 115
Figure IV.13 Flow Extraction: Analysis within one SGOP 115
Figure IV.14 Flow Extraction: Analysis of two SGOP 115
Figure IV.15 Flowchart of the Camera Classification Algorithm 117
Figure IV.16 Detection Process using DCE 118
Figure IV.17 Histogram simplification algorithm 119
Figure IV.18 Colour Histogram Simplification Process 120
Figure V.1 Shot duration extraction for sample MPEG file ulosci.mpg 126
Figure V.2 Scalable shot length distribution generation applying uniform division 127
Figure V.3 Exponential division 127
Figure V.4 Distribution generation based on the production rules: non-linear division 127
Figure V.5 Normalised division: overall database SLD histogram is uniform 128
Figure V.6 Typical SLD representation as applied in the genre classification algorithm 129
Figure V.7 An example of the shot activity measure from frame difference metric Δ 130
Figure V.8 SA representation example 131
Figure VI.1. Mapping between low-level features and high-level concepts. 138
Figure VI.2 Flowchart of the learning unit 139
Figure VI.3 Flowchart of the annotation unit 140
Figure VI.4 Knowledge representation for changes on dominant colour in key-frames 142
Figure VI.5 A set of five key-frames from the sequence news136.mpg. 144
Figure VII.1 Detection of the cuts on the 6th, 16th and 23rd frame in the test sequence 146
Figure VII.2 Obvious lack of defined motion vectors during transitions 148
Figure VII.3 Difference metrics Δ for three types of gradual changes: cut, wipe and dissolve149
Figure VII.4 Stages in the temporal analysis for the sample mpeg file ulosci.mpg 150
Figure VII.5 Robustness to the MPEG resolution/bitrate change 151
Figure VII.6 DCE algorithm results 152
Figure VII.7 DCE linearization of two adjacent line segments 153
Figure VII.8 Evident increase in the cost function 154
Figure VII.9 Motion Flow extraction 155
Figure VII.10 Dotted lines show the key-frame positions 156
Figure VII.11 Summary of the commercial video clip 156
Figure VII.12 Scalable Quantisation of the colour historgam descriptor 157
Figure VII.13 Video Filtering into three sub-classes: news, commercials and others. 158

Figure VII.14 Example of the 6-dimensional temporal descriptor 159

Figure VII.15 Definition of the fuzzy variables for the knowledge representation. 160

Figure VII.16 Temporal descriptors of the training dataset. 160

Figure VII.17 IPOT 2001 Demo 163

Figure VII.18 DTI 2002 Demo... 163

Figure VII.19 MPhil transfer Demo 164

LIST OF ABBREVIATIONS

ATM	Asynchronous Transfer Mode
CBIR	Content Based Image Retrieval
CBVIR	Content Based Video Indexing and Retrieval
CIFF	Digital Video Format with resolution 352x288 pixels
CMA	Computational Media Aesthetics
CPU	Central Processing Unit
DC	Direct Current
DCE	Discrete Contour Evolution
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DPCM	Delta Pulse Code Modulation
DVD	Digital Versatile Disc
EMD	Earth Movers Distance
GOP	Group of Pictures
H.26X	Family of video coding standards published by the International Telecom Union
HDTV	High-definition Television
HMM	Hidden Markov Model
HSV	Hue Saturation Value – colour space
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
IST	Information Society Technologies
JPEG	Joint Photographic Experts Group
MB	MacroBlock
MPEG	Motion Picture Experts Group
MV	Motion Vector
NOKF	Number of Key Frames
NOKP	Number of Key Points
NTSC	National Television System Committee, USA/Japan TV broadcast standard
PAL	Phase Alternating Line, major European TV broadcast standard
QCIF	Digital Video Format with resolution 176x144 pixels
RGB	Red Green Blue – colour space
SAD	Shot Activity Distribution
SGOP	Sub Group of Pictures
SLD	Shot Lenght Distribution
SQL	ANSI standard computer language for accessing and manipulating databases
YUV	Colour space - Y is luminance component, and U and V are chrominance components

I. INTRODUCTION

I.1. PROLOGUE

In the year 1936 and continuing into the Second World War years, German engineer Konrad Zuse had been building a computer in the living room of his parents' apartment in Berlin. It was the first working digital computer to have its programs driven by a punched tape. The tape Zuse used was actually discarded 35mm film tape. Like every other mass-produced object of the industrial age, film copies would often end up as waste, pilling up in dustbins outside film studios. It was probably there that a young Berliner had found the material essential for his new invention. Zuse carved the first dots and bars of digital information over smiles and tears cut as the unsuccessful takes of the pre war German filmmaking.

Things remain same, more or less, for a modern computer. The content of film scenes and TV shows are far from being comprehensible to current digital machinery. Having the latest computer vision armoury, we are raising the question of content analysis in visual media and its application to visual information retrieval.

I.2. PROBLEM

It was only in the 20th century that media became so deeply ingrained in our lives. Yet a new era of the digital media world is emerging in our global village. With the advent of the World Wide Web we are experiencing a new form of world perception through all-pervasive digital media.

Development of the new forms of media have arisen hand-in-hand with the development of the computing machinery. Media has defined its identity through mainly technological terms like its discrete and numerical representation, scalability, automation, variability, etc.

All these features enable enormous possibilities of digital media and, by that, our perception of the world. In order to make use of it we need to develop better ways of interaction with the new forms of media. The most prevalent form of the new media is digital multimedia, and in this thesis, we will try to find ways to improve our interaction with it.

The term *multimedia* refers broadly to information presented in different formats such as text, graphics, video, animation and sound in an integrated way. Long touted as the future revolution in computing, multimedia applications were, until the mid-1990s, rare due to the expensive hardware required. With the increase in performance and decrease in price, however, multimedia applications are now commonplace. Nearly all PCs are capable of displaying video (though the resolution available depends on the power of the computer's video adapter and CPU). Nowadays, MP3 music, DVD, Video-on-demand, interactive TV, IP telephony and digital image collections are everyday phrases.

A timeline of the digital media development is depicted in Figure I.1. It shows that fundamental technologies, such as the laser disk and video games, were invented in the 1970s and 1980s..The multimedia industry has grown significantly in the last decade with the total world market estimated to be about $50 billion.

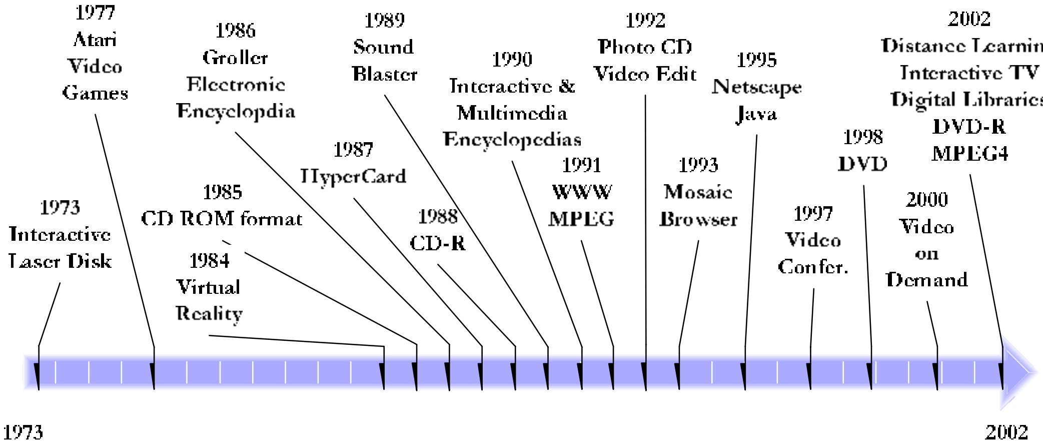


Figure I.1 Development Timeline of Multimedia Technologies

Another aspect of the digital multimedia revolution is the establishment of a new media industry comprising the computer, entertainment, communication and consumer electronics companies. Information technology and media industries, like telephone, cable, and satellite TV companies, TV and radio broadcasters, Internet Service Providers, multimedia and gaming software designers are currently involved in creating new products and services to attract customers and create new markets.

The main driving force of multimedia nowadays is its omnipresence through the biggest network in the world – the Internet. In order to adapt to networked modalities, major research efforts in the multimedia field include media streaming, media retrieval from large and remote repositories, media compression and resource management. Conversely, multimedia forms one of the main driving force of the Internet nowadays.

Moved by the recent collapse of the IT market, internet businesses are trying to revive their customer's needs by offering them rich multimedia content with music, video streaming and on-line gaming experiences.

Either via the Internet or locally, creators of digital media (as well as end-users) approach multimedia through various interfaces, but the primary form of the digital media is the multimedia database. This structured collection of digital data has not only enabled the storage of large multimedia records, but has boosted the choices of experiencing the stored media.

> *"… Following analysis of linear perspective as a "symbolic form" of the modern age, we may even call database a new symbolic form of a post-modern computer age, a new way to structure our experience of ourselves and of the world . . . that appears to us as an endless and unstructured collection of images, texts, and other data records, it is only appropriate that we will be moved to model it as a database …"[Lev Manovich, Database as a Symbolic Form, 1998.]*

In order to facilitate user's utilisation of these "endless and unstructured" collections, a system for the intuitive exploration of multimedia databases has to be developed. This system should enable; easy access to multimedia data; intuitive browsing of the records; the placing of meaningful queries; the retrieval of desirable media; the suggestion of appropriate content, etc.

While getting deeply involved with multimedia database technology, one shouldn't forget that, on the other side of the interface, human subjects make queries and expect meaningful results. The user could be a journalist, a policeman or a student. Using the fact that there are as many modalities of interaction with a multimedia database as there are users, is one of our final objectives.

Moreover, the contemporary creative work of media production, like editing a documentary or making a multimedia web portal, can be understood as the construction of an interface to a multimedia database. The database becomes the centre of the creative process in the computer age [MANOV].

Tendencies, driven by the market, are trying to please consumers by offering less user interference, followed by more autonomous machine processing. These tendencies are inherited from the industrialisation era, when the need for a diminishing of human effort was crucial.

For that reason, variety and creativity of new media and its non-consumer existence in our lives is facing a dead end. Lacking human input, omnipresent media will end up oversimplified and schematised. Thus, unlike its innate richness and diversity, multimedia produces mainly poor, insipid and inhuman content. In order to preserve creativity and richness of new forms of media, and with it our perception of the world, it is crucial to get the human subject deeply involved in the processes of media creation and usage.

So, to achieve this task, one needs to automatically analyse content of the media involved. The research area that has been attracting the attention of the multimedia and computer vision research community during last decade and that tackles the problem of the meaningful management of the multimedia databases is Content Based Indexing and Retrieval.

I.3. RESEARCH SCOPE

Bearing in mind the importance of multimedia databases (and our need to intuitively handle their content), meeting the user's requirements with the available content based video indexing and retrieval technology appears to be the main focus of the research in the field of multimedia and computer vision. The research presented here focuses on the problem of bridging the "semantic gap" between a users' need for meaningful retrieval and the current technology for computational analysis and description of the media content. It takes into account both the high complexity of the real-world implementation and user's need for conceptual video retrieval and browsing.

Focusing on video media as the most complex form of multimedia, in particular the visual aspect of it, is the most challenging task of CBIR. The reason for that are its diversity of expressive elements on one hand and the simplicity of our perception of the visual information.

The initial work focuses on the temporal analysis of video sequences involving shot boundary detection, visual event detection and key-frame extraction. These algorithms, essential for content based video indexing and retrieval system, enable analysis of the main dimension of video media – time. The algorithms parse video sequences into its basic structural units, i.e. shots, and by utilising that information generates a temporally structured description of the sequence. The major stress was on algorithm robustness and efficiency by utilising compressed domain features of the existing video compression standards such as H.26X and MPEG-1/2.

As the next step in content based video retrieval systems, key-frame extraction modules extract a set of the most representative images for a given sequence. This module leads to a substantial reduction of processing complexity, enabling feature analysis in the spatial domain. Thus, the choice of the most appropriate key-frame set is crucial to the quality of retrieval results.

The next requirement for an intuitive video indexing system is the ability to adapt its representations to different environments and application scenarios. Thus attributes like system scalability and multi-resolution analysis are of great importance.

This work confronts the problem of analysis in both temporal and spatial domain descriptors in order to achieve scalability of the system behaviour in indexing as well as during retrieval. Current research efforts in key-frame extraction attempt to embed scale space as part of the algorithm, accomplishing the user customised video summaries and desired preciseness in the process of representation creation. By following the hierarchical structure in the descriptor space this research deals with the problem of a perceptually driven quantisation of colour. In addition, a robust method for extracting camera motions is also presented.

In order to achieve semantic and conceptual retrieval of videos from large repositories, the representation of videos has to render the important aspects of the video sequence in a given context. Thus, in the research presented here, this problem is addressedby generating novel video representations following new paradigms such as computational media aesthetics. Furthermore, contextual information of the video's genre is generated using the representation defined earlier in the research. Using representations extracted from low-level features and contextual information from the genre classification process, a foundation has been built for the automatic semantic annotation of video sequences.

I.4. PROJECT SPECIFICATION AND OBJECTIVES

The research leading to this work has been supported by the Engineering and Physical Sciences Research Council (EPSRC), the UK Government's leading funding agency for research and training in engineering and the physical sciences, on the Hierarchical

Video Indexing Project, grant number R01699/01. A Gantt chart of the project timeline can be found in the Appendix of the thesis[1].

The project commenced by focusing on low-level feature extraction aimed at temporal parsing of videos, shot detection and key-frame extraction. However, following the new tendencies in the field towards high-level semantic issues and problems concerning the semantic gap in CBVIR, consequent research has focused more on video classification and automatic annotation tasks.

The main objectives of the project were to:

- Make advances in the area of temporal video analysis including shot detection, key-frame extraction and temporal description, by exploiting compressed domain processing techniques in order to achieve efficient, robust and scalable algorithms,
- Tackle the problem of multi-resolution, scalable and hierarchical video description to gain efficiency in the task of video indexing,
- Generate appropriate representations of videos in the processes of semantic classification and automatic annotation following new perspectives in the area using the above-mentioned descriptors and to
- Make advances towards automatic video annotation e.g. genre classification, annotation propagation, high-level semantic labelling, etc. by exploiting generated representations.

Last, but not the least, our objective was to evaluate the research results by comparing it with published work in the field in an appropriate experimental environment consisting of a representative digital video dataset.

I.5. RESEARCH CONTRIBUTIONS

Following the research guidelines given above, a system for content based video indexing and retrieval has been developed. Almost every module of the system introduces novelty: from highly efficient temporal analysis to video representation

[1] The final part of the project was developed in collaboration with Mr. Andres Dorado together with whom I designed the rule inference system essential for evaluation of the video representations and genre classification algorithm.

supporting automatic annotation and labelling. The major contributions to the general knowledge in the CBVIR field are given below:

- Efficient transformation of the complex MPEG video stream into a one dimensional metric representing the visual activity of the sequence.
- Robust real-time shot detection utilising only compressed domain features based on a generic frame difference metric.
- Scalable and hierarchical temporal description of videos with robust key-frame extraction.
- An HSV colour histogram descriptor based on a perceptual degradation criterion and hierarchical quantisation.
- Video representation based on the available set of low-level features utilising shot length distribution, shot activity, dominant colour change, etc. in order to achieve a good foundation for further processing.
- Hierarchical k-means clustering of videos into genre sub-classes.
- Lexicon based classification/annotation using rule-based annotation.

These contributions have lead not only to a set of theoretical methods for solving problems of CBVIR, but a real world implementation as well, sowing the seeds of collaboration with media industry and further commercial development.

I.6. STRUCTURE

The Chapters are structured in a way to gradually introduce the problem tackled in this work. In Chapter II a wider background theory of the CBVIR field is presented, bringing the overview of the basic approaches and paradigms throughout the development of the CBVIR technology. Consequently, Chapter III introduces a detailed description of closely related research work in the area, focusing on temporal analysis, feature representation and methods to tackle the problem of the "semantic gap". Chapter IV thoroughly describes the methods developed for temporal analysis and extraction of low-level video descriptors, including frame difference extraction, shot detection and key-frame extraction and colour analysis. Based on the set of extracted low-level descriptors, a video representation model and its application in genre classification is presented in Chapter V. An experimental environment is described in Chapter VI, while Chapter VII presents the results achieved. After the discussion and the conclusion of Chapter VII, the thesis ends with the bibliography and appendices of the project timeline.

II. CONTENT BASED VIDEO INDEXING AND RETRIEVAL

II.1. OVERVIEW

The contemporary development of various multimedia compression standards combined with a significant increase in desktop computer performance, and a decrease in the cost of storage media, has led to the widespread exchange of multimedia information. The availability of cost effective means for obtaining digital video has led to the easy storage of digital video data, which can be widely distributed over networks or storage media such as CDROM or DVD. Unfortunately, these collections are often not catalogued and are accessible only by the sequential scanning of the sequences. To make the use of large video databases more feasible, we need to be able to automatically index, search and retrieve relevant material.

Content-Based Video Indexing and Retrieval (CBVIR) has been the focus of the research community during last 15 years. The main idea behind this concept is to access information and interact with large collections of videos referring to and interacting with its content, rather than its form. Although there has been a lot of effort put in this research area the outcomes were relatively disappointing. The discontinuity between the available content descriptions like colour layout or motion activity and the user's need for rich semantics in user queries makes user approval of automated content retrieval systems very difficult. Thus, in order to develop a meaningful CBVIR system one has to involve multidisciplinary knowledge ranging from image and video signal processing to semiotic theories and video production techniques. Signal processing and computer vision methodologies achieved astonishing results in extracting structural and perceptual features from the video data. Algorithms from database system theory and other computer science disciplines enabled efficient, adaptive and intelligent indexing and retrieval of data with various structure and content. Furthermore, fields like computational linguistics and even semiotics have engaged with problems of natural language and even visual media semantics. However, this knowledge is scattered and needs a way to fuse into one system that will enable content-based retrieval of videos in a way natural for users.

This Chapter gives an insight into foundations and chronological development of CBVIR as well as introducing the biggest challenges to the research community in the filed. It presents the research context in which this work aims to develop groundwork for semantic video indexing and retrieval. Section 2 introduces the scope of the research with the short description of CBVIR development during the last decade. Further on, the problem of the "semantic gap" is defined and discussed in Section 3. As vital issues of this work, the problem of knowledge representation in video databases and the computational media aesthetics as the first fruitful theory trying to solve it are described in Sections 4 and 5. In order to achieve high-level retrieval CBVIR system has to involve the context information profoundly. Thus, the issues raised by the context analysis in CBVIR systems are given in Section 6. Chapter concludes outlining the proposed system for efficient low-level feature extraction and video representation for semantic video retrieval.

II.2. VIDEO INDEXING AND RETRIEVAL

II.2.1. VISUAL MEDIA: OUR SCOPE

Focus of this research is the most complex form of multimedia – video media. Uniting both visual and aural elements, this heterogeneous digital media has very demanding and complex form. Especially when there is a need to process vast amount of digital data in a video database and enable the user to communicate and interact with it. Furthermore, the visual aspect of video media is the one of the most challenging areas of CBVIR. The reason for that is in its diversity of expressions and forms on one hand and the simplicity of our visual perception. In the book Visual Intelligence Hoffman describes the importance and the challenge of understanding the way we see:

"…Vision is normally so swift and sure, so dependable and informative, and apparently so effortless that we naturally assume that it is, indeed, effortless. But the swift ease of vision, like the graceful ease of an Olympic ice skater, is deceptive. Behind the graceful ease of the skater are years of rigorous training, and behind the swift ease of vision is an intelligence so great that it occupies nearly half of the brain's cortex. Our visual intelligence richly interacts with, and in many cases precedes and drives, our rational and emotional intelligence. To understand visual intelligence is to understand, in large part, who we are…" [HOFFM].

Our perception of the world is in the first place visual. Therefore, exploring the visual part of the video media should be the essential milestone in the development process of a CBVIR system. Demands for high computational complexity of data processing and still unexplored semantic issues are unique challenges that keep the research in visual information retrieval a hot topic within the research community.

II.2.2. THREE GENERATIONS OF CBVIR

In the first generation of visual retrieval systems attributes of visual data are extracted manually. Such attribute-based representations entail a high level of image abstraction and model visual content at a conceptual level. They identify significant entities contained in the image or video (an object, a person, etc.), object parts (eyes in the face, boat in the lake, etc.) or the scene represented and concepts associated to it (a landscape, a storm, etc.). Representation schemes like relational models and object-oriented models are used. Search engines work in the textual domain and use either traditional query languages like SQL or full text retrieval. Cost of annotation is typically very high and the whole process suffers from subjectivity of descriptions, in that the annotator is a different person from the one who issues the query.

Different from the first generation, second-generation systems address perceptual features like colour, textures, shape, spatial relationships, etc. They concentrate on obtaining fully automated numeric descriptors from objective measurements of the visual content and support retrieval by content based on combinations of these features. These systems take advantage of the research in pattern recognition and computer vision, which has provided solutions to model and extract visual primitives from image frames. Therefore, in these systems image processing, pattern recognition and computer vision subsystems are an integral part of the architecture and operation. Retrieval is based on similarity models that somehow replicate the way in which humans assess similarity between different objects. Unlike still images, video conveys informative messages through multiple planes of communication. These include the way in which the frames are linked together by using editing effects (cut, fades, dissolves, mattes, etc.), and high level information embedded in the frame sequence (the characters, the story content, the story message). Text embedded in the video frames and the other sensory data like speech and sound can be employed to extract useful data. Research on second-generation video retrieval has mostly been concerned with automatic extraction of the video structure [BOREC] – by detecting the edit effects that permit video composition, the extraction of the key-frames from the shots, and modelling perceptual content of these key-frames. In this way the problem of video retrieval by content has been reduced to the problem of retrieval by content of structured still images.

A CBVIR system depicted in the Figure II.1 has the typical structure of a second generation retrieval system with additional user relevance feedback functionality. Initially, it segments video into its temporal units like shots or scenes and afterwards extracts a set of representative key-frames. Exploiting various image processing and computer vision techniques system generates a low-level feature descriptor and stores it in a metadata database for later retrieval. When user makes a query, query is transformed into the structurally same low-level feature descriptor and the search engine finds the most similar record from a metadata base. The relevance feedback unit monitors feedback given by user during the retrieval process and adapts the feature descriptor in order to achieve more consistent results in terms of perceptual similarity.

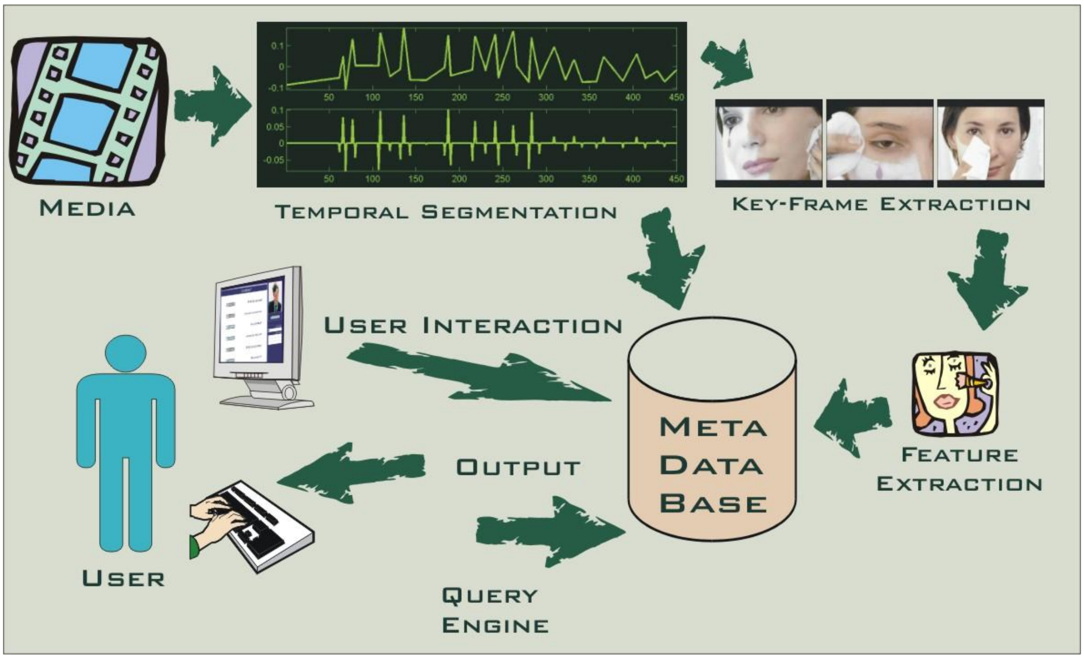


Figure II.1 Scheme of a common CBVIR System

Despite of some effective results that have been reported in the literature, a key problem with second-generation retrieval systems remains bridging the semantic gap between the system and users. Virtually all the systems proposed so far use only low-level perceptively meaningful representations of pictorial data. Similarity of perceptual properties is generally of little use in most practical cases of retrieval by content, if not combined with similarity of high-level information.

We are now on the way to third generation retrieval systems, looking for more information from images, audio and video content. Who are the characters, their roles, the actions and their logical relations, as well as the feeling the user perceives, are information that we aim to extract automatically, with no or minimal manual intervention, so as to support objective semantic-based retrieval. Third generation retrieval systems are particularly important for video media. Much more that single images, retrieval of video is generally meaningful only if performed at high levels of representation and has to do with image sequence classification based on semantically meaningful categories of information. In fact, human memory is much more concerned with the narrative and discourse structure of the video content than merely with perceptual elements of the video. Individual frames are not perceived as such, and the spectator doesn't realise the segmentation into shots and the editing performed by director. Instead he perceives the rhythm of the sequence (which is induced by the editing), the scenes (which are obtained from shots), the story (including the characters, their roles, actions and their logical relations), as well as the feeling (which depends on the combination of perceptual facts like colours, objects, music, sounds, etc. and from the meaning of the story).

II.3. SEMANTIC GAP

One of the major failings of current media annotation systems is the semantic gap that refers to the discontinuity between the simplicity of features or content descriptions that can be currently computed automatically and the richness of semantics in user queries posed for media search and retrieval.

The failing of current systems is that while "the user seeks semantic similarity, the database can only provide similarity on data processing". The authors define the semantic gap as the "lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data has for a user in a given situation" [SMOUL]. Bridging this semantic gap between the simplicity of available visual features and the richness of user semantics is the key issue in building effective content management systems.

II.3.1. THE QUESTION OF MEANING: FROM QUERY BY EXAMPLE TO SEMANTIC RETRIEVAL

Visual information retrieval has emerged in the last 10 years as a natural extension of certain database ideas to multimedia data— in particular, for images and video. The idea seemed natural in its simplicity: retrieve media from a large repository based on its content or, more precisely, certain standard interpretations of its contents.

Such a program's feasibility assumes that there is something we can reasonably call a media's *meaning*. Traditional computer vision hypothesized that we could, in principle, extract meaning from the visual data and represent it in a symbolic or numerical way. In other words, it is possible to extract features from images or videos and cast them into an appropriate metric space in such a way that similar images or videos have similar meaning. The *query by example* model of multimedia databases is based on this idea.

Although it makes a smaller ontological commitment, query by example still presupposes the existence of a media's meaning. Impossible as it might be to characterize this meaning using syntactic features, it is nevertheless still a function of the visual data and, although absolute meaning can't be revealed, similarity of meaning between images can.

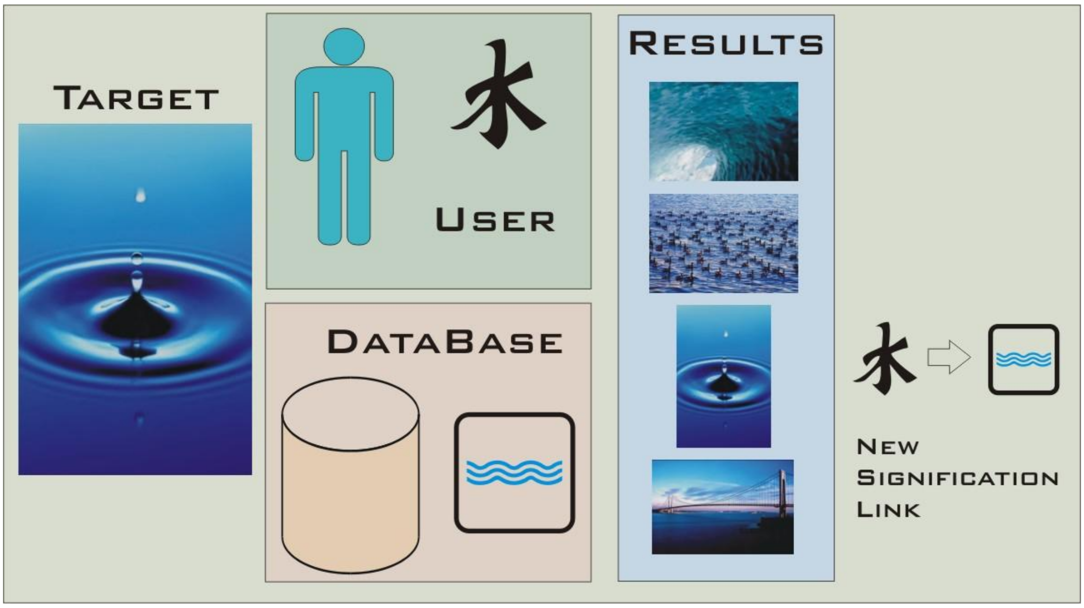


Figure II.2 Bridging the gap between user's query concept and metadata representation of the multimedia database record

A fair share of the problems that plague multimedia databases comes from this semantic presupposition, and we'll only solve these problems by redefining the concept and role of meaning in an information system. Meaning is not a datum that is presented in the image or video clip (however implicitly) and that can be computed and encoded prior to the query process, but it is the result of the user activity during the query process. This is the case of emergent semantics: the meaning of an image or clip is given by its relations with all the other records in the database as it emerges during the query interaction. It is no longer a property of a single media element but of the whole system, which consists of that element, all the others, the user, and his/her cultural presuppositions. The concept of emergent semantics started as a computational linguistic method, but has been present in multimedia retrieval for some time now and has lit the area with a brand new light. It owes its paradigm of dynamic construction of meaning to the semiotics, discipline responsible for creation of meaning from signs.

So as to digest the vast amount of information involved in the construction of video semantics it is substantial to define appropriate video representation in a CBVIR system. The next section raises the issue of video knowledge representation as the breaking point of the signification chain between the digital video media on one side and the user on the other.

II.4. KNOWLEDGE REPRESENTATION FOR VIDEO

The problem of bridging the semantic gap in content based video indexing and retrieval requires very complex analysis of the low-level video features. Therefore the development of techniques involved in CBVIR area set course towards intelligent and knowledge based approaches.

Regarding the fact that the research community is agreed on the need for artificial intelligence and knowledge methods to achieve meaningful content-based multimedia retrieval, essential question would be how to adequately represent videos to and within intelligent and/or knowledge based systems. The core of intelligent video indexing is the knowledge representation model, analogous to the data-model in a generic database system. Explicit knowledge representation has been recognized as the key to dealing with domain specific problems in the artificial intelligence community [RICH]. Thus, the problem of video representation model is at the core of the CBVIR development process.

A typical processing chain that links user with the desired media, video clip in our case, is depicted in Figure II.3. Video clip is fed into the database system by digitalisation or in its native digital form. Represented as a data stream it is processed by the feature extraction module that produces low-level descriptor metadata. In order to achieve user-centred retrieval offering conceptual and semantic communication with the user system has to develop contextually adaptive representation of videos.

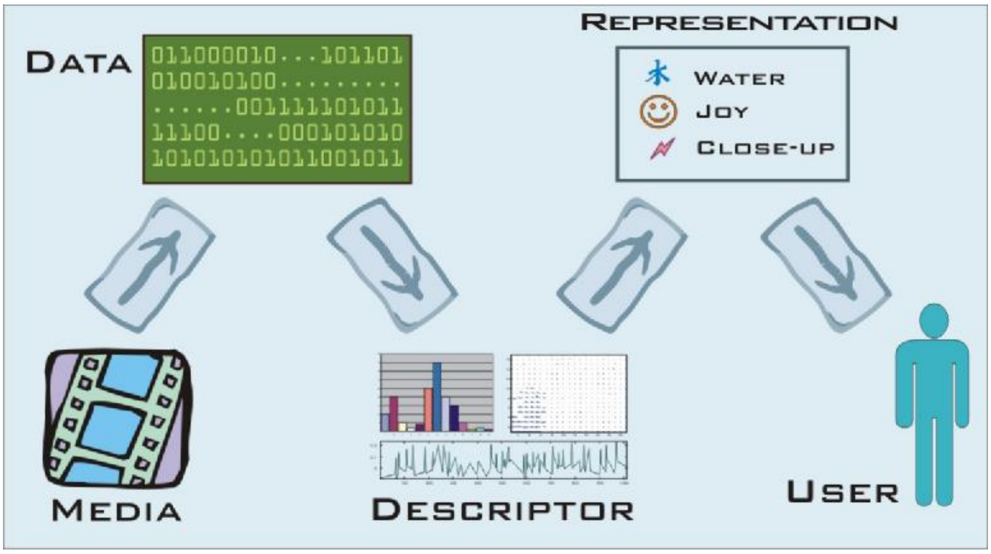


Figure II.3 Representation as a part of the processing chain between the input media and the user

In a CBVIR system videos, clips or single frames should be represented as points in an appropriate multidimensional metric space where dissimilar videos are distant from each other, similar videos are close to each other, and where the distance function captures well the user's concept of similarity [CASTELLI]. Just like in Figure II.4, representation looking glass transforms this multidimensional metric space into concepts intrinsically understandable by users in a given context.

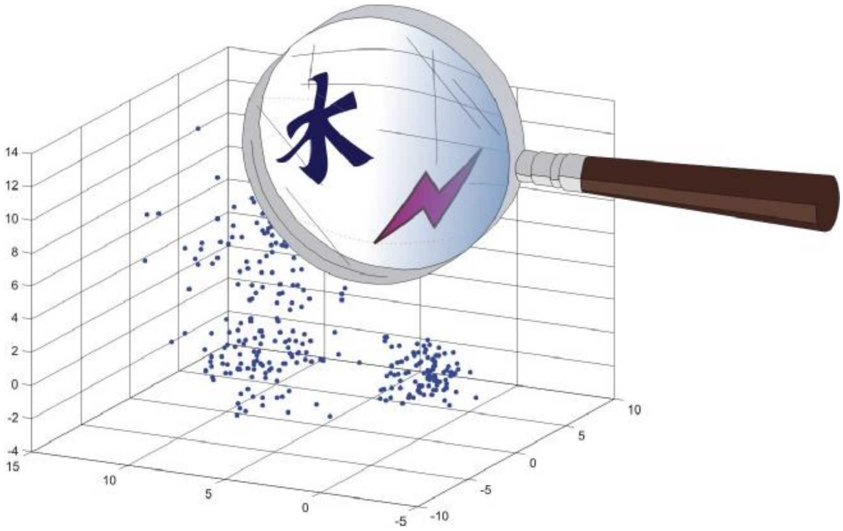


Figure II.4 Representation looking glass
– transforming low-level metric space into the user's conceptual space -

Unfortunately, in the existing CBVIR systems a metric space that encapsulates user's similarity demands hasn't been developed yet. Neither has an appropriate video representation. Although the user centred disciplines that model the user's behaviour in the given multimedia retrieval context, like relevance feedback analysis or HCI, have been a hot topics in the area for years now, there haven't been any revolutionary achievements. Besides, research activities in the video knowledge representation domain hasn't been that vigorous yet. Nevertheless, by the emergence of the inclinations in the area towards media aesthetics, semiotics and film theory a new perspective is given to the development of the video knowledge representation.

II.5. COMPUTATIONAL MEDIA AESTHETICS

Computational Media Aesthetics is an emerging approach to multimedia analysis that aims at bridging the semantic gap and by building innovative content annotation and navigation services. This approach is founded upon an understanding of media elements and their role in synthesis and manipulation of multimedia content with a systematic study of the media production. It proposes a framework for computational understanding of the dynamic nature of the narrative structure and techniques via analysis of the integration and sequencing of audio/visual elements.

II.5.1. BACKGROUND

To address the issue of semantic gap, it is essential to have an approach that goes beyond representing what is being directly shown in a video or a movie, and aims to understand the semantics of the content portrayed and to harness the emotional, visual appeal of the content seen. It should focus on deriving a computational scheme to analyze and understand the content of video and its form. Accepted rules and techniques in video production are used by directors worldwide to solve problems presented by the task of transforming a story from a written script to a captivating narration [ARIJON]. These rules, termed as film grammar in the movie domain, refer to repeated use of certain objects, visual imagery, and patterns in many films to instantly invoke a specific cinematic experience to the viewers. The rules and icons serve as shorthand for compressing story information, characters, and themes into known familiar formulae, often becoming the elements of a genre production. They constitute a style or form of artistic expression that is characteristic of content portrayed, and can be considered to be almost idiomatic in the language of any program composer or director. Production rules are found more in history of use, than in an abstract predefined set of regulations, and elucidate on ways in which basic visual and aural elements can be synthesized into larger structures.

Employment of these tacitly followed rules in any genre not only can be understood and derived automatically with a systematic study of media productions, but also be exploited in characterizing what is happening in a video for high-level video/film abstraction in an algorithmic framework.

II.5.2. OVERVIEW OF CMA

The Computational Media Aesthetics framework approaches the computational understanding of the dynamic nature of the narrative structure and techniques via analysis of the integration and sequencing of audio/visual elements. It is aimed at bridging the semantic gap and building effective content management systems at higher level of abstraction. Further, it puts video/film analysis on a sound footing resting on principles and practices from video/film production rather than on ad hoc schemes.

Zettl [ZETTL] defines Media Aesthetics as a study and analysis of media elements such as lighting, motion, colour and sound both by themselves and their roles in synthesizing effective productions. Computational Media Aesthetics [DORAI] is defined as the algorithmic study of a number of image and aural elements in media and the computational analysis of the principles that have emerged underlying their use and manipulation, individually or jointly, in the creative art of clarifying, intensifying, and interpreting some event for the audience.

What does this new framework entail? By focusing on the emotional and visual appeal of the content, it attempts to uncover the semantic and semiotic information by a study of the relations between the cinematic elements and narrative form. It enables distilling techniques and criteria to create efficient, effective and predictable messages in media communications, and to provide a handle on interpreting and evaluating relative communication effectiveness of media elements through a knowledge of film codes that mediate perception, appreciation and rejection.

This approach, undergirded by the broad rules and conventions of content creation, uses the production knowledge to elucidate the relationships between the many ways in which basic visual and aural elements are manipulated in video and their intended meaning and perceived impact on content users. Its computational scheme analyzes videos to understand the film grammar, in particular and uses the set of rules that are commonly followed during the narration of a story, to assist us in deriving the annotation or description of video contents effectively. A system built using this principled approach where videos are analyzed guided by the tenets of film grammar will be effective in providing high-level concept oriented media descriptions that can function across many contexts and in enhancing the quality and richness of descriptions derived.

II.5.3. NOVELTY AND CONTRIBUTION OF CMA

Let's discuss what sets this approach apart from other schemes at the initial stage. It extracts complex constructs, or expressive elements that expose the underlying semantic information embedded in the media production. The extraction of increasingly complex features from a hierarchical integration of underlying primitives is a commonly followed approach. But the key difference is this framework of analysis based on production knowledge, that is, to both define what to extract, and how to extract these constructs seeks guidance from film grammar and theory. It is done so because directors create and manipulate expressive elements related to some aspect of visual or emotional appeal in particular ways to have maximum impact on the observer. With movies for example, this approach draws attention to the film creation process, and argue that to interpret the data one must see it through the filmmaker's eye. Film theory is the portal that gives us insight into the film creation process. It can tell us not only what expressive elements a director manipulates, but also how he/she does it, why, and what the intended impact is. Thus, complex constructs are both defined and extracted only if media production knowledge tells us that it is an element that the director crafts or manipulates intentionally. These elements by their derivation and study result in grafting human-friendly content descriptions since they directly impact viewers' engagement with the content portrayed.

II.6. CONTEXT IN CBVIR

By definition [OALD], *context* is the situation in which something happens and that helps you to understand it. In the context of CBVIR, that would be the whole environment involved in the video retrieval process. There are many aspects of the retrieval environment: human subject as the user, application of the retrieved results, cultural background, genre, etc. In order to reduce the complexity of retrieval process and facilitate the semantic analysis a CBVIR system needs contextual information extracted automatically from the involved media and its environment. This section discusses ways of utilising contextual information in a CBVIR system.

II.6.1. HUMAN SUBJECT : IMPORTANCE OF ITS PRESENCE

While getting deeply involved with multimedia database technology, one shouldn't forget that on the other side of the interface human subject makes queries and expects meaningful results. That could be a journalist, a policeman or a student. Overlooking the fact that there are as many modalities of interaction with a multimedia database as there are users, would lead us far from our final objective – Content Based Indexing and Retrieval (CBIR).

According to the Information Society Technologies (IST) Programme, launched in 1999 by European Commission to help create a user-friendly information society by building a global knowledge, media and computer space in EU [BADIQUÉ], the development of new forms of media should focus on more interactive interfaces using adaptable multi-sensory interfaces to achieve creativity and authoring of this new form and content. This programme promotes more creativity and better design in key multimedia applications: knowledge, business, publishing, etc. through the development of advanced content technologies. One of the objectives is to develop and demonstrate integrated multi-sensor subsystem using advanced sensor, actuator and display technologies including image and auditory scene processing, 3D and virtual reality, etc. in order to develop new interaction paradigms and inter-mediation technologies supporting intelligent multi-modal, multi-sensorial user interfaces for portable and/or wearable information appliances and systems.

On the other hand, tendencies driven by the market are trying to please a current user by offering less user interference followed by more autonomous machine processing. These tendencies are inherited from the industrialisation era, when the need for diminishing of human labour was crucial, and just exploited later to boost consumer's need for technological revolution.

For that reason, variety and creativity of new media and its non-consumer existence in our lives is facing dead end. Lacking human subject, omnipresent media will end up oversimplified and schematised. Thus unlike its innate richness and diversity, new media produces mainly poor, insipid and inhuman content. In order to preserve creativity and richness of new media, and by that our perception of the world, it is crucial to get the human subject deeply involved in the processes of media creation and usage.

II.6.2. USER RELEVANCE FEEDBACK

In the light of content-based retrieval, human interaction is crucial. A picture is worth a thousand words, and thus a profound challenge comes from the dynamic interpretation of videos under various circumstances. In other words, the perceptual similarity depends upon the application, the person and the context of usage. Therefore as well as the machine needs to learn the associations between high-level concepts and low-level descriptors, it has to learn the users preferences online by getting the user inside the retrieval loop.

A natural way of getting user in the loop is to ask the user to give feedbacks regarding the relevance of the current outputs of the system. Though this is an idea borrowed from the text retrieval field, it seems to work better in visual domain: it is easier to tell the relevance of an image or video than of a document – image reveals its content instantly.

Different methods have been developed under different assumptions or problem settings. The main questions in the user relevance-feedback setup are:

- What is the user looking for? User target can be a particular video clip or a group of similar videos.
- What to feedback? How much information to get from the user: binary yes/no or value estimation.
- What is the distribution? Linear like Gaussian or non-linear kernel based.
- What to learn and how? To improve the current query result or to modify the representation/similarity of records.

Various techniques are proposed to utilise the information given as a feedback from user, starting from heuristic formulations with empirical parameter adjustment [RUI] to the contemporary optimal schemes like support vector machines [TONG], EM algorithms [WU] and Bayesian learning [COX]. This work doesn't utilise user's relevance feedback, but the representations presented support self-adaptive behaviour and revaluation. Furthermore, the rule-based platform developed as the experimental testbed leaves big space for development of a user feedback exploitation.

II.6.3. GENRE THEORY

The genre approach within television and film theory is a way of media classification and it includes a consideration of the codes and conventions applied to the analysed media. Although the term 'genre' translates easily from the French as 'type' or 'kind', its meaning within media studies is both more complex and more far-reaching than this simple explanation suggests. It is possible to study genres in a range of ways: as socio-historical actualities, as thematic and ideological constructions deriving from history, and in terms of their conventions in iconography, visual imagery, narrative patterns and archetypal characters. However the scope of this research limits our comprehension of genres only through production codes applied in genre-oriented media production.

Genre classification in film studies was originally used in relation to Hollywood films made within the specific historical and economic conditions of the studio system. The economic conditions of production - that is, the 'factory-like' arrangements whereby films were a product churned out as quickly and cheaply as possible - meant that once a studio hit upon a commercially successful idea, it would be repeated, with minor variations, for some time. Thus, in the beginning of the filmmaking, films and shows sharing a similar theme or targeted audiences were having similar production conditions, e.g. lightning, actors, make-up and costumes as well as the editing rules and thus were classified as a genre.

In addition to common production rules applied on film, the television industry utilises genre classification as a shorthand mean of scheduling, targeting and maintaining popularity. Television relies on regularity of programming to provide continuity, predictability and reassurance to its audiences. Therefore the information about the broadcast timing added to its genre label squeezes the analysed show into a narrow contextual space facilitating computational analysis in a more meaningful way.

The most common genres are commercials, news programs, situation comedies, soap operas, documentaries, sports shows, talk shows, action adventure programs, detective shows, science-fiction shows, hospital dramas, and westerns. In principle, there may be a finite number of genres and each television show should fit into only one of them, if the classification system works perfectly.

Given the proliferation of television forms and channels, classification into recognisable genres is becoming increasingly difficult, even on a common-sense level. Although the genre approach may be losing its relevance in TV and film studies, it is a quintessential way of getting contextual information in the CBVIR area. Production of the genre-based shows tend to be formulaic - that is, they observe certain familiar conventions which make it relatively easy for audiences to follow them. By analysing those conventions this work investigates ways of representing video sequences targeting automatic annotation and indexing of videos and their further semantic analysis in CBVIR systems.

II.7. SUMMARY

This Chapter brought an overview of content-based video indexing and retrieval, its chronological development and current problems that occupy the research community in the field. After introducing the major focus of CBVIR and presenting three generations of CBVIR systems from its cradle to the current developments, Section 3 presented the first and foremost problem of CBVIR and that is the semantic gap between the user conceptual needs and low-level perceptual descriptors we can extract automatically today. After questioning the matter of meaning in a visual retrieval system, in Section 4 the diminished problem of knowledge representation for video is highlighted and some aspects of its development were presented. As one of the main paradigms that try to deal with the problem of semantic gap, computational media aesthetics was outlined in the Section 5. Contextual issues in the CBVIR systems were discussed in Section 6, introducing the need for classification more intrinsic to video media – genre classification.

Concluding this overview of the main research topic involved in this work, one thing should be stressed. In order to achieve further development of the CBVIR filed one has to focus on the creation of the groundwork for semantic analysis of videos. In other words, relying upon the vast amount of low-level information extracted from videos appropriate knowledge representations have to be generated and put in the appropriate context just to start approaching the signification space needed for semantics' creation in visual media. Moreover, the technical characteristics of the CBVIR system like efficiency, robustness and scalability should be maintained in order to implement real world applications.

III. CURRENT TENDENCIES IN CBVIR

III.1. OVERVIEW

This chapter brings an overview of the state-of-the-art research specifically related to the problems in the CBVIR area addressed in this work. Being key video formats used in the development process, MPEG video compression standards are described in the following Section 2. Temporal analysis methods e.g. shot detection and key-frame extraction are presented in the Sections 3 and 4. In addition to methods for temporal description and analysis, an overview of the spatial feature extraction, specifically colour based, is given in Section 5. Finally, a short survey of research focused on the representational and contextual aspects of CBVIR in Sections 6 and 7 concludes the chapter.

III.2. MPEG VIDEO COMPRESSION STANDARDS

The purpose of this section is to provide an overview of the MPEG-1 [ISO1] and MPEG-2 [ISO2] video coding algorithms and standards and their role in video communications. The Moving Picture Experts Group (MPEG) is a working group of ISO/IEC in charge of the development of international standards for compression, decompression, processing, and coded representation of moving pictures, audio and their combination. MPEG developed video coding standards MPEG-1/2 that are used as the main video format in this work. Thus the next section describes MPEG-1 and MPEG-2 in more detail presenting their basic concepts in the section opening, followed by the theory and techniques involved.

III.2.1. FUNDAMENTALS OF MPEG VIDEO COMPRESSION

Generally speaking, video sequences contain a significant amount of *statistical* and *subjective* redundancy within and between frames. The ultimate goal of video source coding is the bit-rate reduction for storage and transmission by exploring both statistical and subjective redundancies and to encode a "minimum set" of information using entropy-coding techniques. This usually results in a compression of the coded video data compared to the original source data. The performance of video compression techniques depends on the amount of redundancy contained in the image data as well as on the actual compression techniques used for coding. With practical coding schemes a trade-off between coding performance (high compression with sufficient quality) and implementation complexity are targeted. For the development of the MPEG compression algorithms the consideration of the capabilities of "state of the art" technology foreseen for the lifecycle of the standards was most important.

Dependent on the applications requirements we may envisage "lossless" and "lossy" coding of the video data. The aim of "lossless" coding is to reduce image or video data for storage and transmission while retaining the quality of the original images - the *de*coded image quality is required to be identical to the image quality prior to encoding. In contrast the aim of "lossy" coding techniques - and this is relevant to the applications envisioned by MPEG-1 and MPEG-2 video standards - is to meet a given target bit-rate for storage and transmission. Important applications comprise transmission of video over communications channels with constrained or low bandwidth and the efficient storage of video. In these applications high video compression is achieved by degrading the video quality - the decoded image "objective" quality is reduced compared to the quality of the original images prior to encoding (i.e. taking the mean-squared-error between both the original and reconstructed images as an objective image quality criteria). The smaller the target bit-rate of the channel the higher the necessary compression of the video data and usually the more coding artefacts become visible. The ultimate aim of lossy coding techniques is to optimise image quality for a given target bit rate subject to "objective" or "subjective" optimisation criteria. It should be noted that the degree of image degradation (both the objective degradation as well as the amount of visible artefacts) depends on the complexity of the image or video scene as much as on the sophistication of the compression technique - for simple textures in images and low video activity a good image reconstruction with no visible artefacts may be achieved even with simple compression techniques.

III.2.2. THE MPEG VIDEO CODER SOURCE MODEL

The MPEG digital video coding techniques are statistical in nature. Video sequences usually contain statistical redundancies in both temporal and spatial directions. The basic statistical property upon which MPEG compression techniques rely is inter-pel correlation, including the assumption of simple correlated translatory motion between consecutive frames. Thus, it is assumed that the magnitude of a particular image pel can be predicted from nearby pels within the same frame (using Intra-frame coding techniques) or from pels of a nearby frame (using Inter-frame techniques). Intuitively it is clear that in some circumstances, i.e. during scene changes of a video sequence, the temporal correlation between pels in nearby frames is small or even vanishes - the video scene then assembles a collection of uncorrelated still images. In this case Intra-frame coding techniques are appropriate to explore spatial correlation to achieve efficient data compression. The MPEG compression algorithms employ Discrete Cosine Transform (DCT) coding techniques on image blocks of 8x8 pels to efficiently explore spatial correlations between nearby pels within the same image. However, if the correlation between pels in nearby frames is high, i.e. in cases where two consecutive frames have similar or identical content, it is desirable to use Inter-frame DPCM coding techniques employing temporal prediction (motion compensated prediction between frames). In MPEG video coding schemes an adaptive combination of both temporal motion compensated prediction followed by transform coding of the remaining spatial information is used to achieve high data compression (hybrid DPCM/DCT coding of video).

III.2.3. SUBSAMPLING AND INTERPOLATION

Almost all video coding techniques make extensive use of subsampling and quantization prior to encoding. The basic concept of subsampling is to reduce the dimension of the input video (horizontal dimension and/or vertical dimension) and thus the number of pels to be coded prior to the encoding process. This technique may be considered as one of the most elementary compression techniques which also makes use of specific physiological characteristics of the human eye and thus removes subjective redundancy contained in the video data - i.e. the human eye is more sensitive to changes in brightness than to chromaticity changes. Therefore the MPEG coding schemes first divide the images into YUV components (one luminance and two chrominance components). Next the chrominance components are subsampled relative to the luminance component with a Y:U:V ratio specific to particular applications (i.e. with the MPEG-2 standard a ratio of 4:1:1 or 4:2:2 is used).

III.2.4. MOTION COMPENSATED PREDICTION

Motion compensated prediction is a powerful tool to reduce temporal redundancies between frames and is used extensively in MPEG-1 and MPEG-2 video coding standards as a prediction technique for temporal DPCM coding. The concept of motion compensation is based on the estimation of motion between video frames, i.e. if all elements in a video scene are approximately spatially displaced, the motion between frames can be described by a limited number of motion parameters (i.e. by motion vectors for translatory motion of pels).

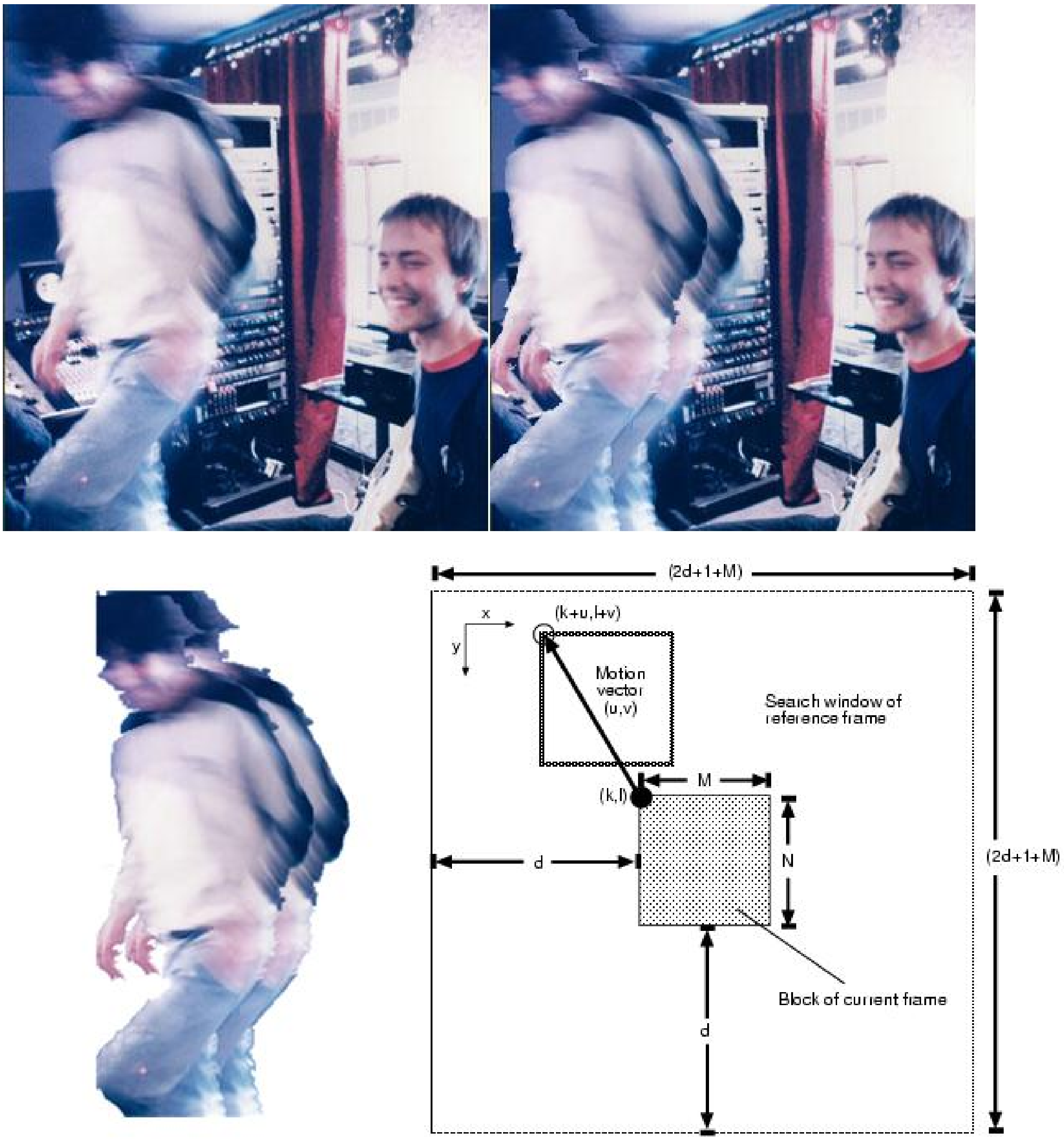


Figure III.1 Motion Compensation in MPEG stream

In this simple example the best prediction of an actual pel is given by a motion compensated prediction pel from a previously coded frame. Usually both, prediction error and motion vectors, are transmitted to the receiver. However, encoding one motion information with each coded image pel is generally neither desirable nor necessary. Since the spatial correlation between motion vectors is often high it is sometimes assumed that one motion vector is representative for the motion of a "block" of adjacent pels. To this aim images are usually separated into disjoint blocks of pels (i.e. 16x16 pels in MPEG-1 and MPEG-2 standards) and only one motion vector is estimated, coded and transmitted for each of these blocks (see Figure III.2).

In the MPEG compression algorithms the motion compensated prediction techniques are used for reducing temporal redundancies between frames and only the prediction error images - the difference between original images and motion compensated prediction images - are encoded. In general the correlation between pels in the motion compensated Inter-frame error images to be coded is reduced compared to the correlation properties of Intra-frames due to the prediction based on the previous coded frame.

Figure III.1 shows block matching approach for motion compensation: One motion vector MV(u,v) is estimated for each block in the actual frame N to be coded. The motion vector points to a reference block of same size in a previously coded frame N-1. The motion compensated prediction error is calculated by subtracting each pel in a block with its motion shifted counterpart in the reference block of the previous frame.

III.2.5. TRANSFORM DOMAIN CODING

Transform coding has been studied extensively during the last two decades and has become a very popular compression method for still image coding and video coding. The purpose of Transform coding is to de-correlate the Intra- or Inter-frame error image content and to encode Transform coefficients rather than the original pels of the images. To this aim the input images are split into disjoint blocks of pels $\boldsymbol{b}$ (i.e. of size *NxN* pels). The transformation can be represented as a matrix operation using a *NxN* Transform matrix $\boldsymbol{A}$ to obtain the *NxN* transform coefficients $\boldsymbol{c}$ based on a linear, separable and unitary *forward* transformation

$$c = A \cdot b \cdot A^{T} \tag{III.1}$$

Here, $\boldsymbol{A}^{T}$ denotes the transpose of the transformation matrix $\boldsymbol{A}$. Note, that the transformation is reversible, since the original *NxN* block of pels $\boldsymbol{b}$ can be reconstructed using a linear and separable *inverse* transformation

$$b = A^{T} \cdot c \cdot A \tag{III.2}$$

Upon many possible alternatives the Discrete Cosine Transform (DCT) applied to smaller image blocks of usually *8x8* pels has become the most successful transform for still image and video coding [AHMED]. In fact, DCT based implementations are used in most image and video coding standards due to their high decorrelation performance and the availability of fast DCT algorithms suitable for real time implementations.

A major objective of transform coding is to make as many Transform coefficients as possible small enough so that they are insignificant (in terms of statistical and subjective measures) and need not be coded for transmission. At the same time it is desirable to minimize statistical dependencies between coefficients with the aim to reduce the amount of bits needed to encode the remaining coefficients.

The DCT is closely related to Discrete Fourier Transform (DFT) and it is of some importance to realize that the DCT coefficients can be given a frequency interpretation close to the DFT. Thus low DCT coefficients relate to low spatial frequencies within image blocks and high DCT coefficients to higher frequencies. This property is used in MPEG coding schemes to remove subjective redundancies contained in the image data based on human visual systems criteria. Since the human viewer is more sensitive to reconstruction errors related to low spatial frequencies than to high frequencies, a frequency adaptive weighting (quantization) of the coefficients according to the human visual perception (perceptual quantization) is often employed to improve the visual quality of the decoded images for a given bit rate.

The combination of the two techniques described above - temporal motion compensated prediction and transform domain coding - can be seen as the key elements of the MPEG coding standards. A third characteristic element of the MPEG algorithms is that these two techniques are processed on small image blocks (of typically 16x16 pels for motion compensation and 8x8 pels for DCT coding). To this reason the MPEG coding algorithms are usually referred to as hybrid block-based DPCM/DCT algorithms.

III.2.6. MPEG-1

The video compression technique developed by MPEG-1 covers many applications from interactive systems on CD-ROM to the delivery of video over telecommunications networks. The MPEG-1 is thought to be generic. To support the wide range of applications profiles a diversity of input parameters including flexible picture size and frame rate can be specified by the user. MPEG has recommended a constraint parameter set: every MPEG-1 compatible decoder must be able to support at least video source parameters up to TV size: including a minimum number of 720 pixels per line, a minimum number of 576 lines per picture, a minimum frame rate of 30 frames per second and a minimum bit rate of 1.86 Mbits/s.

MPEG-1 was primarily targeted for multimedia CD-ROM applications, requiring additional functionality supported by both encoder and decoder. Important features provided by MPEG-1 include frame based *random access* of video, *fast forward/fast reverse (FF/FR)* searches through compressed bit streams, *reverse playback* of video and *editability* of the compressed bit stream.

III.2.7. THE BASIC MPEG-1 INTER-FRAME CODING SCHEME

The basic MPEG-1 (as well as the MPEG-2) video compression technique is based on a MacroBlock structure, motion compensation and the conditional replenishment of MacroBlocks. As outlined in Figure III.2 the MPEG-1 coding algorithm encodes the first frame in a Group of Pictures in Intra-frame coding mode (I-picture). Each subsequent frame is coded using Inter-frame prediction (P-pictures) - only data from the nearest previously coded I- or P-frame is used for prediction. The MPEG-1 algorithm processes the frames of a video sequence as a block-based structure. Each colour input frame in a video sequence is partitioned into non-overlapping MacroBlocks as depicted in Figure III.2. Each MacroBlock contains blocks of data from both luminance and co-sited chrominance bands - four luminance blocks (Y1, Y2, Y3, Y4) and two chrominance blocks (U, V), each with size *8x8* pels. Thus the sampling ratio between Y:U:V luminance and chrominance pels is 4:1:1. P-pictures are coded using motion compensated prediction based on the nearest previous frame. Each frame is divided into disjoint "MacroBlocks" (MB). With each MacroBlock (MB), information related to four luminance blocks (Y1, Y2, Y3, Y4) and two chrominance blocks (U, V) is coded. Each block contains *8x8* pels.

The block diagram of the basic hybrid DPCM/DCT MPEG-1 encoder structure is depicted in Figure III.3. The first frame in a video sequence (I-picture) is encoded in *INTRA* mode without reference to any past or future frames. At the encoder the DCT is applied to each *8x8* luminance and chrominance block and, after output of the DCT, each of the 64 DCT coefficients is uniformly quantized (Q). The quantiser coefficients used to quantize the DCT-coefficients within a MacroBlock are transmitted to the receiver.

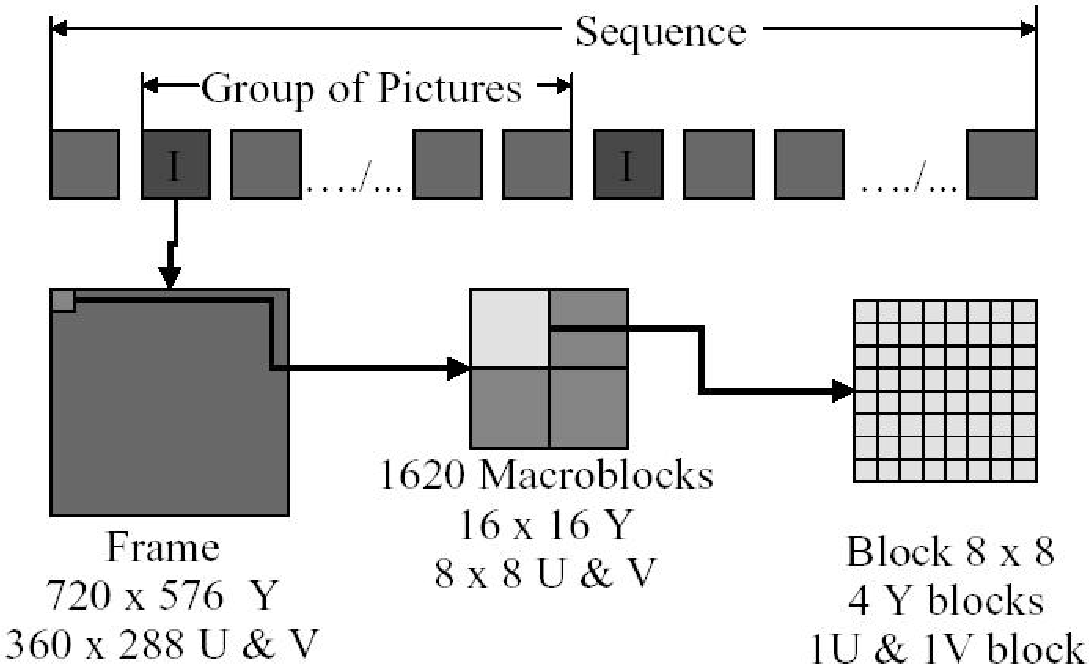


Figure III.2 MPEG sequence decomposition

After quantization, the lowest DCT coefficient (DC coefficient) is treated differently from the remaining coefficients (AC coefficients). The DC coefficient corresponds to the average intensity of the component block and is encoded using a differential DC prediction method. The non-zero quantizer values of the remaining DCT coefficients and their locations are then "zig-zag" scanned and run-length entropy coded using variable length code (VLC) tables.

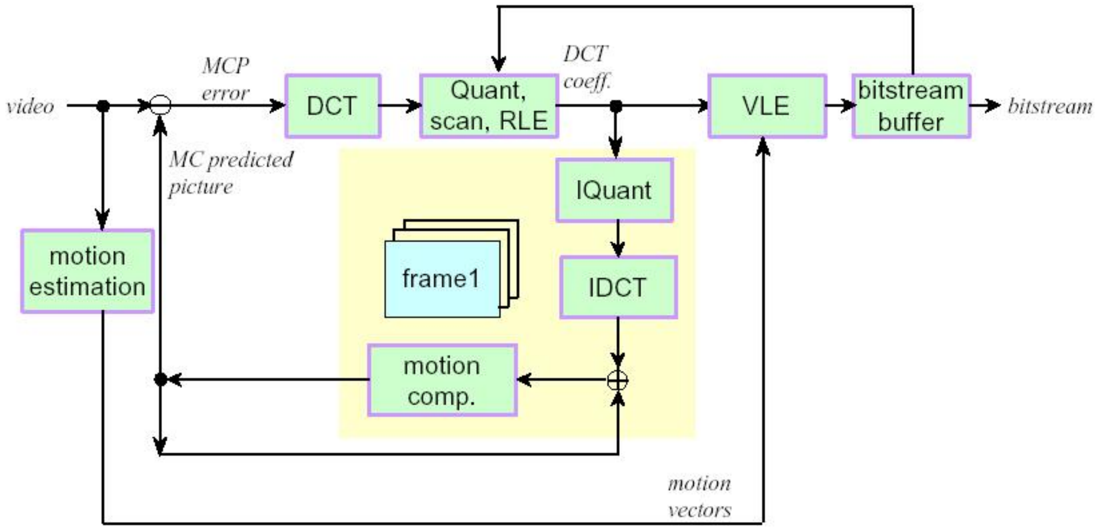


Figure III.3 Block diagram of a basic MPEG encoder structure.

The decoder performs the reverse operations, first extracting and decoding (VLD) the variable length coded words from the bit stream to obtain locations and quantiser values of the non-zero DCT coefficients for each block. With the reconstruction (Q*) of all non-zero DCT coefficients belonging to one block and subsequent inverse DCT (DCT-1) the quantized block pixel values are obtained. By processing the entire bit stream all image blocks are decoded and reconstructed.

For coding P-pictures, the previously I- or P-picture frame *N-1* is stored in a frame store (FS) in both encoder and decoder. Motion compensation (MC) is performed on a MacroBlock basis - only one motion vector is estimated between frame *N* and frame *N-1* for a particular MacroBlock to be encoded. These motion vectors are coded and transmitted to the receiver. The motion compensated prediction error is calculated by subtracting each pel in a MacroBlock with its motion shifted counterpart in the previous frame. A *8x8* DCT is then applied to each of the *8x8* blocks contained in the MacroBlock followed by quantization (Q) of the DCT coefficients with subsequent run-length coding and entropy coding (VLC). The decoder uses the reverse process to reproduce a MacroBlock of frame *N* at the receiver.

The advantage of coding video using the motion compensated prediction from the previously reconstructed frame *N-1* in an MPEG coder is illustrated in Figure III.4 for a typical test sequence. Figure III.4a depicts a frame at time instance *N* to be coded and Figure III.4b the reconstructed frame at instance *N-1* that is stored in the frame store (FS) at both encoder and decoder (note that the motion vectors depicted in the image are not part of the reconstructed image stored at the encoder and decoder). The block motion vectors (MV, see also Figure III.1) depicted in Figure III.4b were estimated by the encoder motion estimation procedure and provide a prediction of the translatory motion displacement of each MacroBlock in frame *N* with reference to frame *N-1*. Figure III.4c depicts the pure frame difference signal (frame *N* - frame *N-1*), which is obtained if no motion compensated prediction is used in the coding process - thus all motion vectors are assumed to be zero. Figure III.4d depicts the motion compensated frame difference signal when the motion vectors in Figure III.4b are used for prediction. It is apparent that the residual signal to be coded is greatly reduced using motion compensation if compared to pure frame difference coding in Figure III.4c.

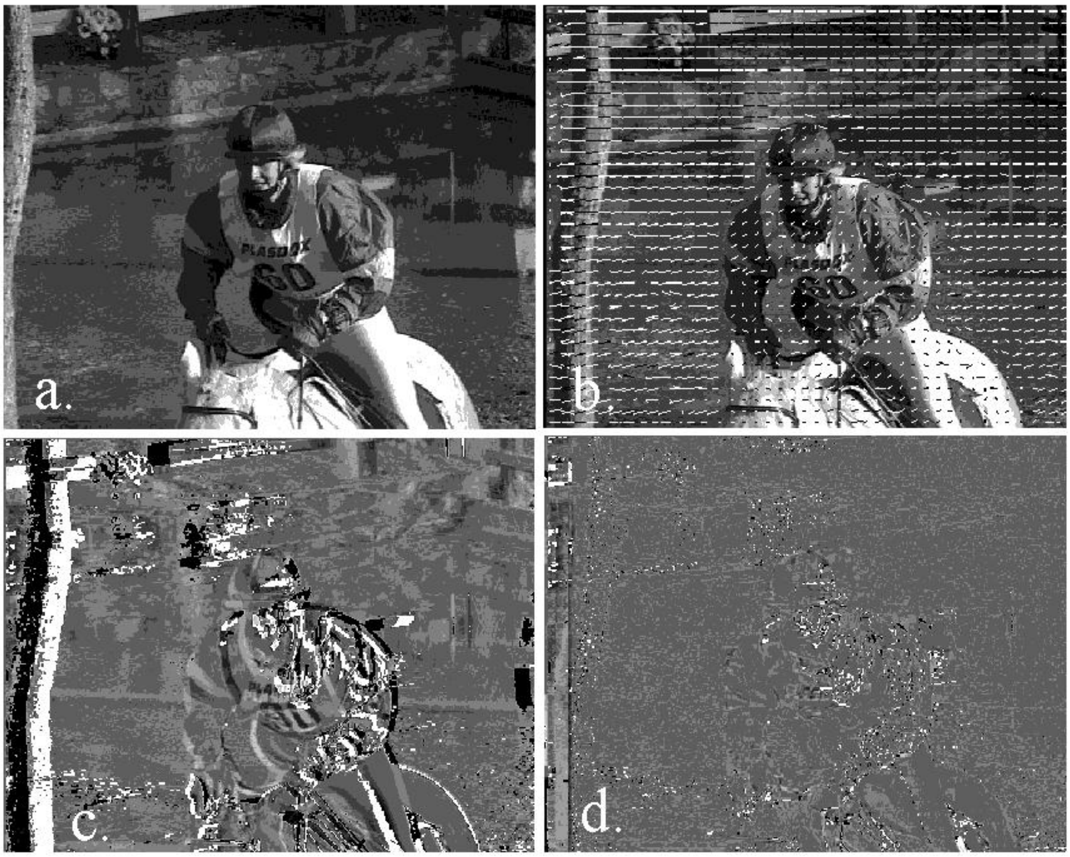


Figure III.4 a) Frame at time instance *N* to be coded b) Frame at *N-1* used for prediction of the content in frame *N* c) prediction error image obtained without using motion d) motion compensated prediction is employed.

III.2.8. SPECIFIC FUNCTIONALITIES

An essential feature supported by the MPEG-1 coding algorithm is the possibility to update MacroBlock information at the decoder only if needed - if the content of the MacroBlock has changed in comparison to the content of the same MacroBlock in the previous frame. The MPEG standard distincts mainly between three different MacroBlock coding types (MB types):

1. *Skipped MB* - prediction from previous frame with zero motion vector. No information about the MacroBlock is coded nor transmitted to the receiver.
2. *Inter MB* - motion compensated prediction from the previous frame is used. The MB type, the MB address and, if required, the motion vector, the DCT coefficients and quantization stepsize are transmitted.
3. *Intra MB* - no prediction is used from the previous frame (Intra-frame prediction only). Only the MB type, the MB address and the DCT coefficients and quantization stepsize are transmitted to the receiver.

For accessing video from storage media the MPEG-1 video compression algorithm was designed to support important functionalities such as random access and fast forward (FF) and fast reverse (FR) playback functionalities. To incorporate the requirements for storage media and to further explore the significant advantages of motion compensation and motion interpolation, the concept of B-pictures (bi-directional predicted/bi-directional interpolated pictures) was introduced by MPEG-1.

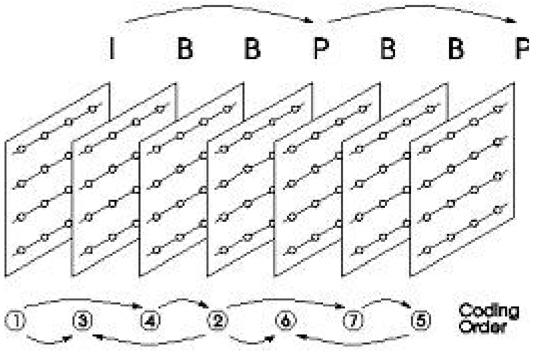


Figure III.5 I-pictures (I), P-pictures (P) and B-pictures (B) used in a MPEG-1 video sequence.

This concept is depicted in Figure III.5 for a group of consecutive pictures in a video sequence. Three types of pictures are considered: Intra-pictures (I-pictures) are coded without reference to other pictures contained in the video sequence. I-pictures allow access points for random access and FF/FR functionality in the bit stream but achieve only low compression. Inter-frame predicted pictures (P-pictures) are coded with reference to the nearest previously coded I-picture or P-picture, usually incorporating motion compensation to increase coding efficiency. Since P-pictures are usually used as reference for prediction for future or past frames they provide no suitable access points for random access functionality or editability. Bi-directional predicted/interpolated pictures (B-pictures) require both past and future frames as references.

To achieve high compression, motion compensation can be employed based on the nearest past and future P-pictures or I-pictures. B-pictures themselves are never used as references. B-pictures can be coded using motion compensated prediction based on the two nearest already coded frames (either I-picture or P-picture). The arrangement of the picture coding types within the video sequence is flexible to suit the needs of diverse applications. The direction for prediction is indicated in the figure. The user can arrange the picture types in a video sequence with a high degree of flexibility to suit diverse applications requirements. As a general rule, a video sequence coded using I-pictures only (I I I I I I .....) allows the highest degree of random access, FF/FR and editability, but achieves only minor compression. A sequence coded with a regular I-picture update and no B-pictures (i.e. I P P P P P P I P P P P ...) achieves low compression and a certain degree of random access and FF/FR functionality. Incorporation of all three pictures types, as i.e. depicted in Figure 8 (I B B P B B P B B I B B P...), may achieve high compression and reasonable random access and FF/FR functionality. It is the most common MPEG stream format used and therefore the approach to compressed domain analysis adopted in this work assumes that B-type frames are present in the stream.

III.2.9. MPEG-2 STANDARD OVERVIEW

Worldwide MPEG-1 developed into an important and successful video coding standard with an increasing number of products becoming available on the market. A key factor for this success is the generic structure of the standard supporting a broad range of applications and applications specific parameters. However, MPEG continued its standardization efforts in 1991 with a second phase (MPEG-2) to provide a video coding solution for applications not originally covered or envisaged by the MPEG-1 standard. Specifically, MPEG-2 was given the charter to provide video quality not lower than NTSC/PAL and up to CCIR 601 quality. Emerging applications, such as digital cable TV distribution, networked database services via ATM, digital VTR applications and satellite and terrestrial digital broadcasting distribution, were seen to benefit from the increased quality expected to result from the new MPEG-2 standardization phase. Work was carried out in collaboration with the ITU-T SG 15 Experts Group for ATM Video Coding and in 1994 the MPEG-2 Draft International Standard (which is identical to the ITU-T H.262 recommendation) was released [HALHED]. The specification of the standard is intended to be generic - hence the standard aims to facilitate the bit stream interchange among different applications, transmission and storage media.

Basically MPEG-2 can be seen as a superset of the MPEG-1 coding standard and was designed to be backward compatible to MPEG-1 - every MPEG-2 compatible decoder can decode a valid MPEG-1 bit stream. Many video coding algorithms were integrated into a single syntax to meet the diverse applications requirements. New coding features were added by MPEG-2 to achieve sufficient functionality and quality, thus prediction modes were developed to support efficient coding of *interlaced video*. In addition *scalable video* coding extensions were introduced to provide additional functionality, such as embedded coding of digital TV and HDTV, and graceful quality degradation in the presence of transmission errors.

However, implementation of the full syntax may not be practical for most applications. MPEG-2 has introduced the concept of "Profiles" and "Levels" to stipulate conformance between equipment not supporting the full implementation. Profiles and Levels provide means for defining subsets of the syntax and thus the decoder capabilities required to decode a particular bit stream.

As a general rule, each Profile defines a new set of algorithms added as a superset to the algorithms in the Profile below. A Level specifies the range of the parameters that are supported by the implementation (i.e. image size, frame rate and bit rates). The MPEG-2 core algorithm at MAIN Profile features non-scalable coding of both progressive and interlaced video sources. It is expected that most MPEG-2 implementations will at least conform to the MAIN Profile at MAIN Level which supports non-scalable coding of digital video with approximately digital TV parameters - a maximum sample density of 720 samples per line and 576 lines per frame, a maximum frame rate of 30 frames per second and a maximum bit rate of 15 Mbit/s.

III.3. TEMPORAL VIDEO FEATURE EXTRACTION

The temporal video segmentation research efforts have resulted in a great variety of algorithms. Early work focuses on cut detection, while more recent techniques deal with the more difficult problem of gradual-transition detection. Fades, dissolves and wipes are special video editing effects that gradually change the content and therefore are more difficult to detect. Fade-in is a editing effect which allows the progressive transition from a solid black frame to full brightness of a shot content, while a fade-out is a progressive darkening of a shot until the last frame becomes completely black. Dissolve is a superimposition of a fade-in and a fade-out: the first shot fades-out while the following fades-in to full brightness. Wipe is a content transition from one scene to another wherein the new scene is revealed by a moving line or pattern. In simplest form, simulates a window shade being drawn. More sophisticated variations include colorized wipes, quivering wipes and triangle wipes. In the following sections a number of relevant methods is described and compared with the algorithm proposed.

III.3.1. TEMPORAL VIDEO SEGMENTATION IN UNCOMPRESSED DOMAIN

The majority of algorithms for temporal video segmentation exploits uncompressed video data. Usually, a similarity measure between successive images is defined. When two images are sufficiently dissimilar, there is a high probability of a cut. Gradual transitions are detected by using cumulative difference measures and more sophisticated thresholding schemes. Based on the metrics that is used to detect the difference between successive frames, the algorithms for temporal video segmentation in uncompressed domain can be divided broadly into three categories: pixel, block-based and histogram comparisons.

III.3.1.1. Pixel Comparison

Pair-wise pixel comparison (also called template matching) evaluates the differences in intensity or colour values of corresponding pixels in two successive frames. The simplest way is to calculate the absolute sum of pixel differences and compare it against a threshold [KIKUK]:

$$D(i,i+1)=\frac{\sum_{x=1}^{X}\sum_{y=1}^{Y}\left|P_i(x,y)-P_{i+1}(x,y)\right|}{X\cdot Y} \tag{III.3}$$

for grey level images,

$$D(i,i+1)=\frac{\sum_{x=1}^{X}\sum_{y=1}^{Y}\sum_{c}\left|P_i(x,y,c)-P_{i+1}(x,y,c)\right|}{X\cdot Y} \tag{III.4}$$

for colour images,

where i and i+1 are two successive frames with dimension X*Y, $P_i(x,y)$, is the intensity value of the pixel at the coordinates (x,y), in frame i, c is index for the colour components and $P_i(x,y,c)$, is the colour component of the pixel at y,x in frame i.

A cut is detected if the difference D(i,i+1) is above a pre specified threshold *T*. The main disadvantage of this method is that it is not able to distinguish between a large change in a small area and a small change in a large area. For example, cuts are misdetected when a small part of the frame undergoes a large, rapid change. Therefore, methods based on simple pixel comparison are sensitive to object and camera movements. A possible improvement is to count the number of pixels that change in value more than some threshold and to compare the total against a second threshold [ZHANG][NAGAS]:

$$DP(i,i+1,x,y)=\begin{cases}1 & \text{if } \left|P_i(x,y)-P_{i+1}(x,y)\right|>T_1,\\ 0, & \text{otherwise,}\end{cases} \tag{III.5}$$

$$D(i,i+1)=\frac{\sum_{x=1}^{X}\sum_{y=1}^{Y}DP(i,i+1,x,y)}{X\cdot Y} \tag{III.6}$$

If the percentage of changed pixels D(i,i+1) is greater than a threshold T2, a cut is detected. Although some irrelevant frame differences are filtered out, these approaches are still sensitive to object and camera movements. For example, if camera pans, a large number of pixels can be judged as changed, even though there is actually a shift with a few pixels. It is possible to reduce this effect to a certain extend by the application of a smoothing filter: before the comparison each pixel is replaced by the mean value of its neighbours.

III.3.1.2. Block-based comparison

In contrast to template matching that is based on global image characteristic (pixel by pixel 8 differences), block-based approaches use local characteristic to increase the robustness to camera and object movement. Each frame i is divided into b blocks that are compared with their corresponding blocks in i+1. Typically, the difference between i and i+1 is measured by

$$D(i,i+1)=\sum_{k=1}^{b}c_k\cdot DP(i,i+1,k) \tag{III.7}$$

where ck is a predetermined coefficient for the block k and DP(i,i+1,k) is a partial match value between the kth blocks in i and i+1 frames.

In [KASTURI] corresponding blocks are compared using a likelihood ratio:

$$\lambda_k=\frac{\left[\frac{\sigma_{k,i}+\sigma_{k,i+1}}{2}+\left(\frac{\mu_{k,i}+\mu_{k,i+1}}{2}\right)^2\right]^2}{\sigma_{k,i}\cdot\sigma_{k,i+1}} \tag{III.8}$$

where $\sigma_{k,i},\sigma_{k,i+1}$ are the mean intensity values for the two corresponding blocks k in the consecutive frames i and i+1, and $\mu_{k,i},\mu_{k,i+1}$ are their variances, respectively. Then, the number of blocks for which the likelihood ratio is greater than a threshold T_1 is counted:

$$DP(i,i+1,x,y)=\begin{cases}1 & \text{if } \lambda_k>T_1,\\ 0, & \text{otherwise,}\end{cases} \tag{III.9}$$

A cut is declared when the number of changed blocks is large enough, i.e. D(i,i+1) is greater than a given threshold T_2 and c_k=1 for all k. Compared to template matching, this method is more tolerant to slow and small object motion from frame to frame. On the other hand, it is slower due to the complexity of the statistical formulas. Additional potential disadvantage is that no change will be detected in the case of two corresponding blocks that are different but have the same density function. Such situations, however, are very unlikely.

Another block-based technique is proposed by Shahraray [SHAHR]. The frame is divided into 12 non-overlapping blocks. For each of them the best match is found in the respective neighbourhoods in the previous image based on image intensity values. A non-linear order statistics filter is used to combine the match values. Thus, the effect of camera and object movements is further suppressed. The author claims that such similarity measure of two images is more consistent with human judgement. Both cuts and gradual transitions are detected. Cuts are found using thresholds like in the other approaches that are discussed while gradual transitions are detected by identifying sustained low-level increase in match values.

Xiong, Lee and Ip [XIONG] describe a method they call *net comparison*, which attempts to detect cuts inspecting only part of the image. It is shown that the error will be low enough if less than half of so called base windows (non-overlapping square blocks, as in Figure III.6) are checked. Under an assumption about the largest movement between two images, the size of the windows can be chosen large enough to be indifferent to a non-break change and small enough to contain the spatial information as much as possible. Base windows are compared using the difference between the mean values of their grey-level or colour values. If this difference is larger than a threshold, the region is considered changed. When the number of changed windows is greater than another threshold, a cut is declared.

The experiments demonstrated that the approach is faster and more accurate than pixel pair-wise, likelihood and local histogram methods. In their subsequent paper [XIONG1], the idea of video subsampling into space is further extended to subsampling in both space and time. The new Step variable algorithm detects both abrupt and gradual transition comparing frames i and j, where j=i+myStep. If no significant change is found between them, the move is with half step forward and the next comparison is between i+myStep/2 and j+myStep/2. Otherwise, binary search is used to locate the change. If i and j are successive and their difference is bigger than a threshold, cut is declared.

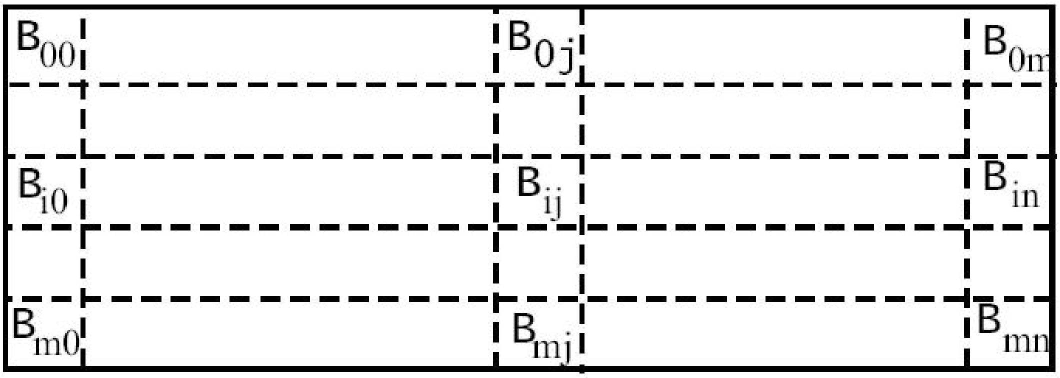


Figure III.6 Non-overlapping square blocks in net comparison algorithm

Otherwise, edge differences between the two frames are compared against another threshold to check for gradual transition. Obviously, the performance depends on the proper setting of myStep: large steps are efficient but increase the number of false alarms, too small steps may result in missing gradual transition. In addition, the approach is very sensitive to object and camera motion.

III.3.1.3. Histogram comparison

A step further towards reducing sensitivity to camera and object movements can be done by comparing the histograms of successive images. The idea behind histogram-based approaches is that two frames with unchanging background and unchanging (although moving) objects will have little difference in their histograms. In addition, histograms are invariant to image rotation and change slowly under the variations of viewing angle and scale. As a disadvantage one can note that two images with similar histograms may have completely different content. However, the probability for such events is low enough, moreover techniques for dealing with this problem have already been proposed in [PASS].

A grey level (colour) histogram of a frame i is an n-dimensional vector $H_i(j)=1,\ldots,n$ where n is the number of grey levels (colours) and H(j) is the number of pixels from the frame i with grey level (colour) j.

III.3.1.4. Global Histogram Comparison

The simplest approach uses an adaptation of the metrics from Equation (III.3): instead of intensity values, grey level histograms are compared. A cut is declared if the absolute sum of histogram differences between two successive frames D(i,i+1) is greater than a threshold *T*:

$$D(i,i+1)=\sum_{j=1}^{n}\left|H_i(j)-H_{i+1}(j)\right| \tag{III.10}$$

where $H_i(j)$ is the histogram value for the grey level j in the frame i, j is the grey value and n is the total number of grey levels.

Another simple and very effective approach is to compare colour histograms. Zhang, Kankanhalli and Smoliar [ZHANG] apply Equation (III.10) where j, instead of grey levels, denotes a code value derived from the three colour intensities of a pixel. In order to reduce the bin number (3 colours x 8 bits create histograms with 2^{24} bins), only the upper two bits of each colour intensity component are used to compose the colour code. The comparison of the resulting 64 bins has been shown to give sufficient accuracy.

To enhance the difference between two frames across a cut, several authors propose the use of the χ^2 test to compare the (colour) $H_i(j)$ histograms and $H_{i+1}(j)$ of the two successive frames i and i+1:

$$D(i,i+1)=\sum_{j=1}^{n}\frac{\left|H_i(j)-H_{i+1}(j)\right|^2}{H_{i+1}(j)} \tag{III.11}$$

When the difference is larger than a given threshold *T*, a cut is declared. However, experimental results reported in [ZHANG] show that χ^2 test not only enhances the difference between two frames across a cut but also increases the difference due to camera and object movements. Hence, the overall performance is not necessarily better than the linear histogram comparison represented in Equation (III.11) In addition, χ^2 statistics requires more computational time. Gargi *et al.* [GHARGI] evaluate the performance of three histogram based methods using six different colour coordinate systems: RGB, HSV, YIQ, L*a*b*, L*u*v* and Munsell. The RGB histogram of a frame is computed as three sets of 256 bins. The other five histograms are represented as a 2-dimensional distribution over the two non-intensity based dimensions of the colour spaces, namely: H and S for the HSV, I and Q for the YIQ, a* and b* for the L*a*b*, u* and v* for the L*u*v* and hue and chroma components for the Munsell space. The number of bins is 1600 (40x40) for the L*a*b*, L*u*v* and YIQ histograms and 1800 (60 hues x 30 saturations/chroma) for the HSV and Munsell space histograms. The difference functions used to compare histograms of two consecutive frames are defined as follows:

Bin-to-bin differences:

$$D(i,i+1)=\sum_{j=1}^{n}\left|H_i(j)-H_{i+1}(j)\right| \tag{III.12}$$

Histogram intersection:

$$D(i,i+1)=1-\frac{\sum_{j=1}^{n}\min\left(H_i(j)-H_{i+1}(j)\right)}{\sum_{j=1}^{n}\max\left(H_i(j)-H_{i+1}(j)\right)} \tag{III.13}$$

Note that for two identical histograms the intersection is 1 and the difference 0 while for two frames which do not share even a single pixel of the same colour (bin), the difference is 1.

Weighted bin differences:

$$D(i,i+1)=\sum_{j=1}^{n}\sum_{k\in N(k)}W(k)\cdot\left(H_i(j)-H_{i+1}(j)\right) \tag{III.14}$$

where N(k) is a neighbourhood of bin j and W(k) is the weight value assigned to that neighbour. A 3x3 or 3 neighbourhoods are used in the case of 2-dimensional and 1-dimensional histograms, respectively.

It is found that in terms of overall classification accuracy YIQ, L*a*b* and Munsell colour coordinate spaces perform well, followed by HSV, L*u*v* and RGB. In terms of computational cost of conversion from RGB, the HSV and YIQ are the least expensive, followed by L*a*b*, L*u*v* and the Munsell space.

So far only histogram comparison techniques for cut detection have been presented. They are based on the fact that there is a big difference between the frames across a cut that results in a high peak in the histogram comparison and can be easily detected using one threshold. However, such one threshold based approaches are not suitable to detect gradual transitions. Although during a gradual transition the frame-to-frame differences are usually higher than those within a shot, they are much smaller than the differences in the case of cut and cannot be detected with the same threshold. On the other hand, object and camera motions might entail bigger differences than the gradual transition. Hence, lowering the threshold will increase the number of false positives. Below we review a simple and effective two-thresholds technique for gradual transition recognition.

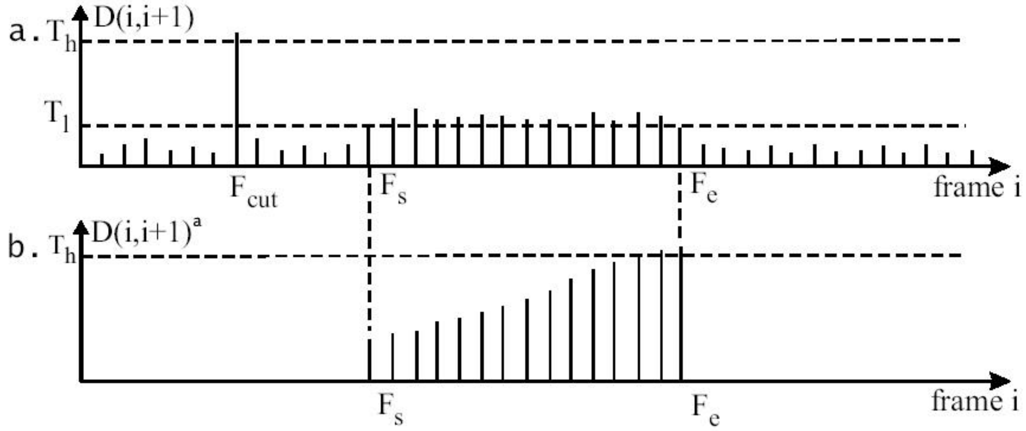


Figure III.7 Twin comparison: a. consecutive, b. accumulated histogram differences.
Figure taken from [ZHANG]

The *twin-comparison* method [ZHANG1] takes into account the cumulative differences between frames of the gradual transition. In the first pass a high threshold T_h is used to detect cuts as shown in Figure III.7a. In the second pass a lower threshold T_l is employed to detect the potential starting frame F_s of a gradual transition. F_s is than compared to subsequent frames (Figure III.7b). This is called an accumulated comparison as during a gradual transition this difference value increases. The end frame F_e of the transition is detected when the difference between consecutive frames decreases to less than T_l, while the accumulated comparison has increased to a value higher than T_h. If the consecutive difference falls below T_l before the accumulated difference exceeds T_h, then the potential start frame F_s is dropped and the search continues for other gradual transitions. It was found, however, that there are some gradual transitions during which the consecutive difference falls below the lower threshold. This problem can be easily solved by setting a tolerance value that allows a certain number of consecutive frames with low difference values before rejecting the transition candidate. As it can be seen, the twin-comparison detects both abrupt and gradual transitions at the same time. Boreczky and Rowe [BOREC] compared several temporal video segmentation techniques on real video sequences and found that twin-comparison is a simple algorithm that works very well.

III.3.1.5. Local Histogram Comparison

As it was already discussed, histogram based approaches are simple and more robust to object and camera movements but they ignore the spatial information and, therefore, fail when two different images have similar histograms. On the other hand, block based comparison methods make use of spatial information. They typically perform better than pair-wise pixel comparison but are still sensitive to camera and object motion and are also computationally expensive. By integrating the two paradigms, false alarms due to camera and object movement can be reduced while enough spatial information is retained to produce more accurate results.

The frame-to-frame difference of frame i and frame i+1 is computed as:

$$D(i,i+1)=\sum_{k=1}^{b} DP(i,i+1,k) \qquad \text{(III.15)}$$

$$DP(i,i+1,k)=\sum_{j}^{n}\left|H_i(j,k)-H_{i+1}(j,k)\right| \qquad \text{(III.16)}$$

where $H_i(j,k)$ denotes the histogram value at grey level j for the region (block) k and b is the total number of the blocks.

For example, Nagasaka and Tanaka [NAGAS] compare several statistics based on grey-level and colour pixel differences and histogram comparisons. The best results were obtained by breaking the image into 16 equal-sized regions, using χ^2 test on colour histograms for these regions and discarding the largest differences to reduce the effects of noise, object and camera movements.

Another approach based on local histogram comparison is proposed by Swanberg *et al.* [SWANB]. The partial difference DP(i,i+1,k) is measured by comparing the colour RGB histograms of the blocks using the following equation:

$$DP(i,i+1,k)=\sum_{c\in\{R,G,B\}}\sum_{l=1}^{n}\frac{\left(H_i^c(l)-H_{i+1}^c(l)\right)^2}{H_i^c(l)-H_{i+1}^c(l)} \qquad \text{(III.17)}$$

Then, Equation (III.7) is applied where c_k is 1/b for all k.

Lee and Ip [LEEIP] introduce a *selective HSV histogram* comparison algorithm. In order to reduce the frame-to-frame differences caused by change in intensity or shade, image blocks are compared in HSV (hue, saturation, value) colour space. It is the use of hue that makes the algorithm insensitive to such changes since hue is independent of saturation and intensity. However, as hue is unstable when the saturation or the value are very low, selective comparison is proposed. To further improve the algorithm by increasing the differences across a cut, local histogram comparison is performed. It is shown that the algorithm outperforms both histogram (grey level global and local) and pixel differences based approaches. However, none of the algorithms gives satisfactory performance on very dark video images.

III.3.1.6. Algorithm Comparison

Compared with the algorithm proposed in this thesis, all temporal video parsing techniques that exploit information in uncompressed domain lack efficiency. The reason for that is in the nature of the approach. In the feature extraction part the majority of uncompressed analysis techniques must initially decode the video stream and afterwards apply some processing on the vast pixel data, which additionally slows down the processing time. Thus, algorithms that base their analysis on pixel data [KIKUK, ZHANG, NAGAS] require substantial processing time. Block-based algorithms [KASTURI, SHAHR, XIONG] and methods based on histogram comparison [ZHANG1, GHARGI, SWANB] achieved considerable improvement in both processing requirements and sensitivity to camera and object motion, but far from the efficiency of the compressed domain analysis. However, reported precision and recall of some algorithms presented in this section [ZHANG1, NAGAS] are almost the same as the same parameters of the algorithm proposed in this thesis.

III.3.2. CLUSTERING-BASED TEMPORAL VIDEO SEGMENTATION

The approaches discussed so far rely on suitable thresholding of similarities between successive frames. However, the thresholds are typically highly sensitive to the type of input video. This drawback is overcome by the application of *unsupervised clustering* algorithm. More specifically, the temporal video segmentation is viewed as a 2-class clustering problem ("scene change" and "no scene change") and the well-known K-means algorithm [PAPPAS] is used to cluster frame dissimilarities. Then the frames from the cluster "scene change" which are temporary adjacent are labelled as belonging to a gradual transition and the other frames from this cluster are considered as cuts. Two similarity measures based on colour histograms were used: χ^2 statistics and the histogram difference defined in Equation (III.10), both in RGB and YUV colour spaces. The experiments show that the χ^2-YUV detects the larger number of correct transitions but the histogram difference-YUV is the best choice in terms of overall performance (i.e. number of false alarms and correct detections). As a limitation we can note that the approach is no able to recognize the type of the gradual transitions. The main advantage of the clustering-based segmentation is that it is a generic techniques that not only eliminates the need for threshold setting but also allows multiple features to be used simultaneously to improve the performance. For example, in their subsequent work Ferman and Tekalp [FERMAN] incorporate two features in the clustering method: histogram difference and pair-wise pixel comparison. It was found that when filtered these features supplement one another, which results in both high recall and precision. A technique for clustering based temporal segmentation on-the-fly was introduced as well.

Due to the fact that clustering techniques presented here apply even more complex analysis on the features extracted from the uncompressed domain, efficiency reported is even worse in comparison to techniques presented in Section III.3.1. Furthermore, robustness to the camera and object motion and algorithm's precision and recall has not been improved. This is due to the fact that clustering based approach doesn't take into account the temporal nature of the shot detection task, but analyses only a set of perceptual frame features.

III.3.3. FEATURE BASED TEMPORAL VIDEO SEGMENTATION

An interesting approach for temporal video segmentation based on features is described by Zabih, Miller and Mai [ZABIH]. It involves analyzing intensity edges between consecutive frames. During a cut or a dissolve, new intensity edges appear far from the locations of the old edges. Similarly, old edges disappear far from the location of new edges. Thus, by counting the entering and exiting edge pixels, cuts, fades and dissolves are detected and classified. To obtain better results in case of object and camera movements, an algorithm for motion compensation is also included. It first estimates the global motion between frames that is then used to align the frames before detecting entering and exiting edge pixels. However, this technique is not able to handle multiple rapidly moving objects. As the authors have pointed out, another weakness of the approach are the false positives due to the limitations of the edge detection method. In particular, rapid changes in the overall shot brightness, and very dark or very light frames, may cause false positives.

Although introducing a novel approach to temporal parsing, especially the detection of gradual changes, this algorithm doesn't bring any improvement regarding efficiency. It extracts edges from the uncompressed domain, and by that intensifies feature extraction so that the overall processing time increases even more in comparison to the techniques presented in Section III.3.1. Therefore, considering the importance of real-time shot detection, this approach, as well as the whole group of algorithms that base their analysis on features extracted from uncompressed domain, cannot compete the efficiency and robustness of the algorithm presented in this thesis.

III.3.4. MODEL DRIVEN TEMPORAL VIDEO SEGMENTATION

The video segmentation techniques presented so far are sometimes referred to as *data driven*, *bottom-up* approaches. They address the problem from data analysis point of view. It is also possible to apply *top-down* algorithms that are based on mathematical models of video data. Such approaches allow a systematic analysis of the problem and the use of several domain-specific constraints that might improve the efficiency.

Hampapur, Jain and Weymouth [HAMPA] present a shot boundary identification approach based on the mathematical model of the video production process. This model was used as a basis for the classification of the video edit types (cuts, fades, dissolves). For example, fades and dissolves are chromatic edits and can be modelled as:

$$S(x,y,t)=S_1(x,y,t)\cdot\left(1-\frac{t}{l_1}\right)+S_2(x,y,t)\cdot\left(1-\frac{t}{l_2}\right) \tag{III.18}$$

where S1(x,y,t) and S2(x,y,t) are two shots that are being edited, S(x,y,t) is the edited shot and l1,l2 are the number of frames for each shot during the edit.

The taxonomy along with the models are then used to identify features that correspond to the different classes of shot boundaries. Finally, feature vectors are fed into a system for frames classification and temporal video segmentation. The approach is sensitive to camera and object motion.

Another model-based technique, called differential model of motion picture, is proposed by Aigrain and Joly [AIGRAIN]. It is based on the probabilistic distribution of differences in pixel values between two successive frames and combines the following factors:

[1] a small amplitude additive zero-cantered Gaussian noise that models camera, film, digitizer and other noises;

[2] an intra shot change model for pixel change probability distribution resulting from object and camera motion, angle, focus and light change;

a shot transition model for the different types of abrupt and gradual transitions. The histogram of absolute values of pixel differences is computed and the number of pixels that change in value within a certain range determined by the models is counted. Then shot transitions are detected by examining the resulting integer sequences. Experiments show 94-100% accuracy for cuts and 80% for gradual transitions detection.

Yu, Bozdagi and Harrington [YU] present an approach for gradual transitions detection based on a model of intensity changes during fade out, fade in and dissolve. At the first pass, cuts are detected using histogram comparison. The gradual transitions are then detected by examining the frames between the cuts using the proposed model of their characteristics. For example, it was found that the number of edge pixels have a local minimum during a gradual transition. However as this feature exhibits the same behaviour in case of zoom and pan, additional characteristics of the fades and dissolves need to be used for their detection. During a fade, the beginning and end image is a constant image, hence the number of edge pixels will be close to zero. Furthermore, the number of edge pixels gradually increases going away from the minimum in either side. In order to distinguish dissolves, the so called double chromatic difference curve is examined. It is based on the idea that the frames of a dissolve can be recovered using the beginning and end frames. The approach has low computational requirements but works under the assumption of small object movement.

Boreczky and Wilcox [BOREC1] use Hidden Markov Models (HMM) for temporal video segmentation. Separate states are used to model shot, cut, fade, dissolve, pan and zoom. The arcs between states model the allowable progressions of states. For example, from the shot state it is possible to go to any of the transition states, but from a transition state it is only possible to return to a shot state. Similarly, the pan and zoom states can only be reached from the shot state, since they are subsets of the shot. The arcs from a state to itself model the length of time the video is in that particular state. Three different types of features (image, audio and motion) are used:

- a standard grey-level histogram distance between two adjacent frames;
- an audio distance based on the acoustic difference in intervals just before and just after the frames and
- an estimate of object motion between the two frames.

The parameters of the HMM, namely the transition probabilities associated with the arcs and the probability distributions of the features associated with the states, are learned by training with the Baum-Welch algorithm. Training data consists of features vectors computed for a collection of video and labelled as one of the following classes: shot, cut, fade, dissolve, pan and zoom. Once the parameters are trained, segmenting the video is performed using the Viterbi algorithm, a standard technique for recognition in HMM.

Thus, thresholds are not required as the parameters are learned automatically. Another advantage of the approach is that HMM framework allows any number of features to be included in a feature vector. The algorithm was tested on different video databases and has been shown to improve the accuracy of the temporal video segmentation in comparison to the standard threshold-based approaches.

III.3.4.1. Algorithm Comparison

Unlike the algorithms presented in previous sections, model driven algorithms for temporal video parsing tackled the problem of robustness and precision by applying more complex analysis of extracted feature set. Methods that model the way videos are being edited [HAMPA, YU] resulted in similar approaches that utilised compressed domain features. In addition, reported precision/recall in [BOREC1] are high and even show that the algorithm is very reliable and robust to camera and object motion. Compared to the algorithm proposed here, these results are better regarding its robustness and precision. However, efficiency of these algorithms is questionable since the features used to model transitions are extracted from the uncompressed domain.

III.3.5. TEMPORAL VIDEO SEGMENTATION IN MPEG COMPRESSED DOMAIN

The previous approaches for video segmentation process uncompressed video. As nowadays video is increasingly stored and moved in compressed format (e.g. MPEG), it is highly desirable to develop methods that can operate directly on the encoded stream. Working in the compressed domain offers the following advantages. First, by not having to perform decoding/re-encoding, computational complexity is reduced and savings on decompression time and decompression storage are obtained. Second, operations are faster due to the lower data rate of compressed video. Last but not least, the encoded video stream already contains a rich set of pre-computed features, such as motion vectors (MVs) and block averages, which are suitable for temporal video segmentation.

Several algorithms for temporal video segmentation in the compressed domain have been reported. According to the type of information used, they can be divided into six non-overlapping groups - segmentation based on:

- DCT coefficients;
- DC terms;
- DC terms, MacroBlock (MB) coding mode and MVs;
- DCT coefficients, MB coding mode and MVs;
- MB coding mode and MVs and
- MB coding mode and bitrate information.

III.3.5.1. Temporal Video Segmentation Based on DCT Coefficients

The pioneering work on video parsing directly in compressed domain is conducted by Arman, Hsu and Chiu [ARMAN1] who proposed a technique for cut detection based on the DCT coefficients of I frames. For each frame a subset of the DCT coefficients of a subset of the blocks is selected in order to construct a vector $V_i=\{c_1, c_2, c_3,\ldots\}$. V_i represents the frame *i* from the video sequence in the DCT space. The normalized inner product is then used to find the difference between frames i and i+φ :

$$D(i,i+\varphi)=\frac{V_i\cdot V_{i+\varphi}}{|V_i|\cdot|V_{i+\varphi}|} \tag{III.19}$$

A cut is detected if $1-|D(i,i+\varphi)|>T_1$ where *T1* is a threshold.

In order to reduce false positives due to camera and object motion, video cuts are examined more closely using a second threshold *T2* ($0<T_1<T_2<1$). If $T_1 < 1-|D(i,i+\varphi)| < T_2$ the two frames are decompressed and examined by comparing their colour histograms.

Zhang *et al.* [ZHANG2] apply a pair-wise comparison technique to the DCT coefficients of corresponding blocks of video frames. The difference metric is similar to pixel comparisons. More specifically, the difference of block l from two frames which are φ frames apart is measured as:

$$DP(i,i+\varphi,l)=\frac{1}{64}\sum_{k=1}^{64}\frac{|c_{l,k}(i)-c_{l,k}(i+\varphi)|}{\max\left[c_{l,k}(i),c_{l,k}(i+\varphi)\right]}>T_1 \tag{III.20}$$

where $c_{l,k}(i)$ is the DCT coefficient of block l in the frame i, k=1,2,…64 and l depends on the size of the frame.

If the difference exceeds a given threshold T_1, the block l is considered to be changed. If the number of changed blocks is larger than another threshold T_2, a transition between the two frames is declared. The pair-wise comparison requires far less computation than the difference metric used by Arman. The processing time can be further reduced by applying Arman's method of using only a subset of coefficients and blocks.

It should be noted that both of the above algorithms [ARMAN1, ZHANG2] may be applied only to I frames of the MPEG compressed video, as they are the frames fully encoded with DCT coefficients. As a result, the processing time is significantly reduced but the temporal resolution is low. In addition, due to the loss of the resolution between the I frames, false positives are introduced and, hence, the classification accuracy decreases. Also, neither of the two algorithms can handle gradual transitions or false positives introduced by camera operations and object motion.

However the processing of these algorithms is minimised, the time needed for feature extraction and analysis is higher than in the algorithm proposed in this thesis. Furthermore, the feature set extracted is only partial, since the features are extracted only from I frames. Therefore, the robustness of these methods is lower compared to the proposed technique.

Following this approach, Yeo and Liu [YEO] proposed a method where so called DC-images are created and compared. DC-images are spatially reduced versions of the original images: the (i,j) pixel of the DC image is the average value of the corresponding block of the compressed frame (Figure III.8).

As each DC term is a scaled version of the block's average value, DC images can be constructed from DC terms. The DC terms of I frames are directly available in the MPEG stream while those of B and P frames are estimated using the MVs and DCT coefficients of previous I frames. It should be noted that the reconstruction techniques is computationally very expensive - in order to compute the DC term of a reference frame (DC_{ref}) for each block, eight 8x8 matrix multiplications and 4 matrix summations are required. Then, the pixel differences of dc-images are compared and a sliding window is used to set the thresholds because the shot transition is a local activity.

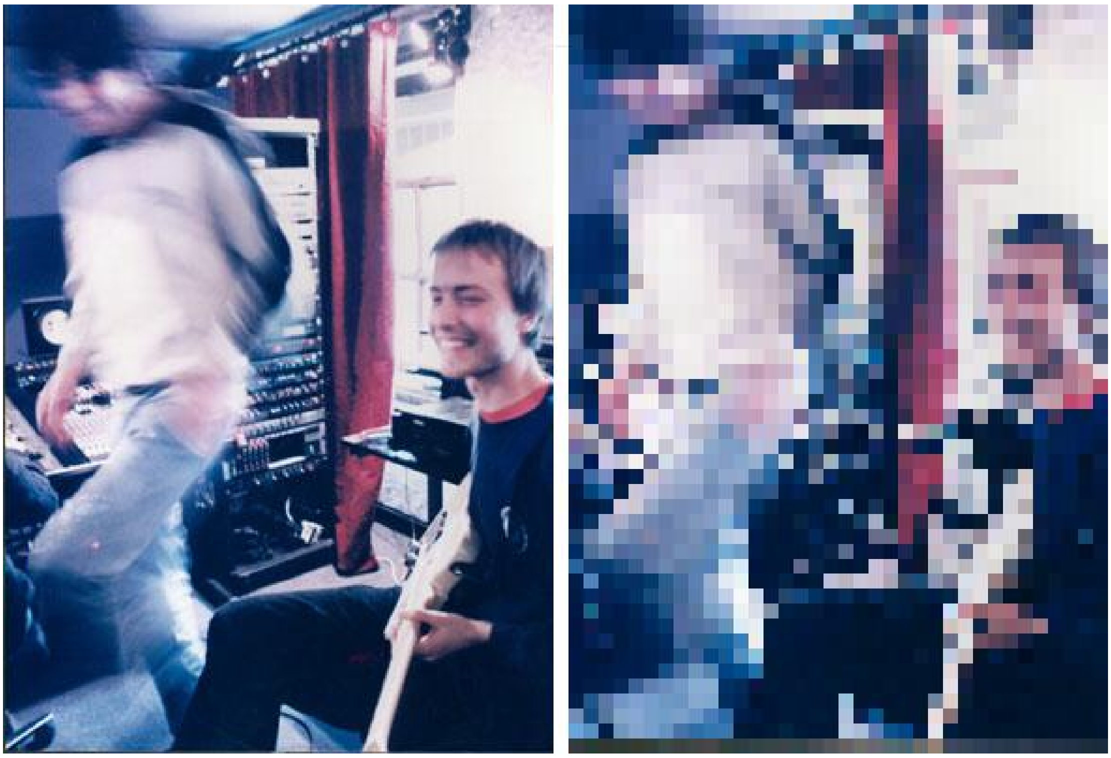


Figure III.8 A full resolution image and its DC image

In order to find a suitable similarity measure, the authors compare metrics based on pixel differences and colour histograms. They confirm that when full images are compared, the first group of metrics is more sensitive to camera and object movements but computationally less expensive than the second one. However, when DC-images are compared, pixel differences based metrics give satisfactory results as DC-images are already smoothed versions of the corresponding full images.

Hence, as in the pixel domain approaches, abrupt transitions are detected using a similarity measure based on the sum of absolute pixel differences of two consecutive frames (DC images in this case):

$$D\left(l,l+1\right)=\sum_{i,j}\left|P_l\left(i,j\right)-P_{l+1}\left(i,j\right)\right| \qquad \text{(III.21)}$$

where l and l+1 are two consecutive DC-images and $P_l(i,j)$, is the intensity value of the pixel in l-th DC-image at the coordinates (i,j).

In contrast to the previous methods for cut detection that apply global thresholds on the difference metrics, Yeo and Liu propose to use local thresholds as scene changes are local activities in the temporal domain. In this way false positives due to significant camera and object motions are reduced. More specifically, a *sliding window* is used to examine m successive frame differences. A cut between frames l and l+1 is declared if the following two conditions are satisfied:

- D(l,l+1) is the maximum within a symmetric sliding window of size 2m-1
- D(l,l+1) is n times the second largest maximum in the window.

The second condition guards against false positives due to fast panning or zooming and camera flashes that typically manifest themselves as sequences of large differences or two consecutive peaks, respectively. The size of the sliding window m is set to be smaller than the minimum duration between two transitions, while the values of n typically range from 2 to 3.

Gradual transitions are detected by comparing each frame with the following kth frame where k is larger than the number of frames in the gradual transition. A gradual transition g_n in the form of linear transition from c_1 to c_2 in the time interval (α_1,α_2), is modelled as

$$g_n = \begin{cases} c_1 & n < \alpha_1 \\ \dfrac{c_2 - c_1}{\alpha_2 - \alpha_1} \cdot \left[n - (\alpha_1 - k) \right] & \alpha_1 \le n < \alpha_2 \\ c_2 & n \ge \alpha_2 \end{cases} \tag{III.22}$$

Then if $k > \alpha_2 - \alpha_1$ the difference between frames *l* and *l+k* from the transition g_n will be

$$D_{gn}(l, l+k) = \begin{cases} 0 \quad n < \alpha_1 - k \\ \dfrac{|c_2 - c_1|}{|\alpha_2 - \alpha_1|} \cdot \left[n - (\alpha_1 - k) \right] \quad \alpha_1 - k \le n < \alpha_2 - k \\ |c_2 - c_1| \quad \alpha_2 - k \le n < \alpha_1 \\ -\dfrac{|c_2 - c_1|}{|\alpha_2 - \alpha_1|} \cdot \left[n - (\alpha_2) \right] \quad \alpha_1 \le n < \alpha_2 \\ 0 \quad n \ge \alpha_2 \end{cases} \tag{III.23}$$

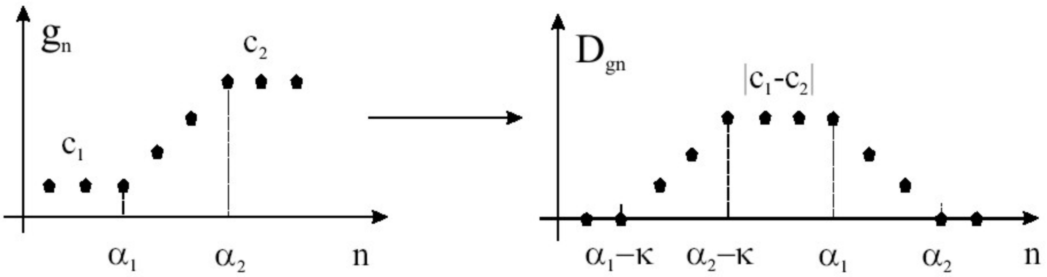


Figure III.9 Gradual transition g_n and pixel difference model $D_{gn}(l,l+k)$ in dissolve detection.
Figure taken from [YEO]

As $D_{gn}(l/l+k)$ corresponds to a symmetric plateau with sloping sides (see Figure III.9), the goal of the gradual transition detection algorithm is to identify such plateau patterns. The algorithm of Yeo and Liu needs eleven parameters to be specified.

In [SHEN] shots are detected by colour histogram comparison of DC term images of consecutive frames. Such images are formed by the DC terms of the DCT coefficients for a frame. DC terms of I pictures are taken directly from the MPEG stream, while those for P and B frames are reconstructed by the following fast algorithm. First, the DC term of the reference image (DC_{ref}) is approximated using the weighted average of the DC terms of the blocks pointed by the MVs, Figure III.10:

$$DC_{ref} = \frac{1}{64}\sum_{\alpha \in E} N_\alpha \cdot DC_\alpha \qquad \text{(III.24)}$$

where DC_α is the DC term of block E is the collection of all blocks that are overlapped by the reference block and N_α is the number of pixels in block that is overlapped by the reference block.

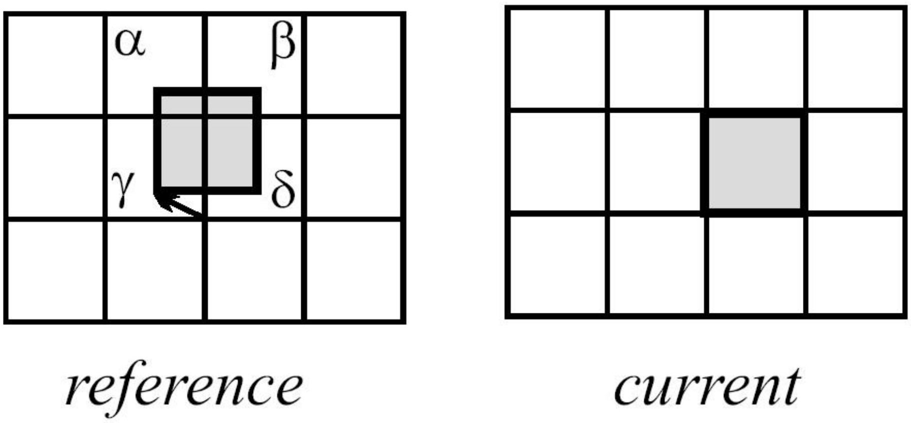


Figure III.10 DC term estimation in the method of Shen and Delp

Then, the approximated DC terms of the predicted pictures are added to the encoded DC terms of the difference images in order to form the DC terms of P and B pictures:

$$DC = DC_{diff} + DC_{ref} \qquad \text{(III.25)}$$

for only forward or only backward prediction

$$DC = DC_{diff} + \frac{1}{2}\left(DC_{ref1} + DC_{ref2}\right) \qquad \text{(III.26)}$$

for interpolated prediction.

In this way the computations are reduced to at most 4 scalar multiplications and 3 scalar summations for each block to determine DC_{ref}.

The histogram difference diagram is generated using the measure from Equation (III.10) comparing DC term images. As it can be seen from Figure III.11, a break is represented by a single sharp pulse and a dissolve entails a number of consecutive medium-heighten pulses. Cuts are detected using a static threshold. For the recognition of gradual transitions, the histogram difference of the current frame is compared with the average of the histogram differences of the previous frames within a window. If this difference is n times larger than the average value, a possible start of a gradual transition is marked. The same value of n is used as a soft threshold for the following frames. End of the transition is declared when the histogram difference is lower than the threshold. Since during a gradual transition not all of the histogram differences may be higher than the soft threshold, similarly to the twin comparison, several frames are allowed to have lower difference as long as the majority of the frames in the transition have higher magnitude than the soft threshold.

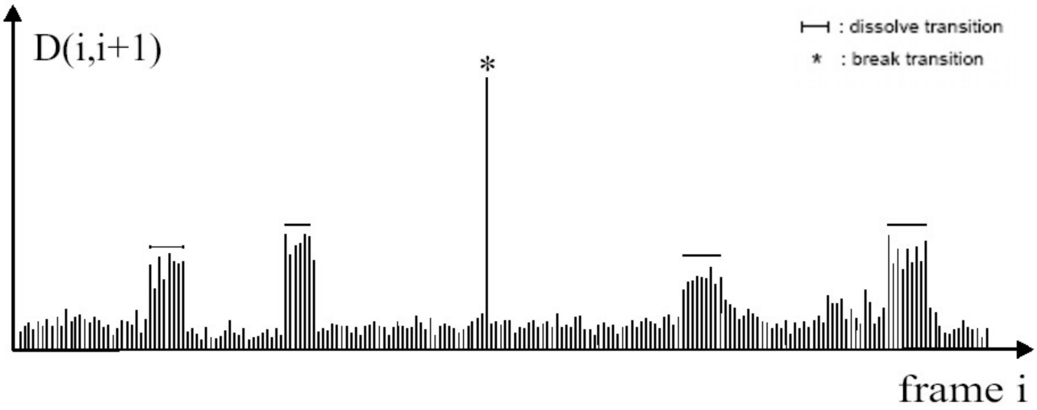


Figure III.11 Histogram difference diagram (*:cut, ---:dissolve). Figure taken from [SHEN].

As only the DC terms are used, the computation of the histograms is 64 times faster than that using the original pixel values. The approach is not able to distinguish rapid object movement from gradual transition. As a partial solution, a median filter (of size 3) is applied to smooth the histogram differences when detecting gradual transitions. There are 7 parameters that need to be specified.

An interesting extension of the previous approach is proposed by Taskiran and Delp [TASKI]. After the DC term image sequence and the luminance histogram for each image are obtained, a two dimensional feature vector is extracted from each pair of images. The first component is the dissimilarity measure based on the histogram intersection of the consecutive DC term images:

$$x_{1i} = 1 - \text{Intersection}\left(H_i - H_{i+1}\right) = 1 - \frac{\sum_{j=1}^{n} \min\left(H_i(j), H_{i+1}(j)\right)}{\sum_{j=1}^{n} H_{i+1}} \qquad \text{(III.27)}$$

where $H_i(j)$ is the luminance histogram value for the bin j in frame i and n is the number of bins used.

The second feature is the absolute value of the difference of standard deviations σ for the luminance component of the DC term images i.e. $x_{i2} = |\sigma_i - \sigma_{i+1}|$. The so called *generalized sequence trace* d for a video stream composed of n frames is defined as di = $\| x_i - x_{i+1} \|$, i=1,…,n.

These features are chosen not only because they are easy to extract. Combining histogram-based and pixel-based parameters makes sense as they complement some of their disadvantages. As it was discussed already, pixel-based techniques give false alarms in case of camera and object movements. On the other hand, histogram-based techniques are less sensitive to these effects but may miss shot transition if the luminance distribution of the frames do not change significantly. It is shown that there are different types of peaks in the generalized trace plot: wide, narrow and middle corresponding to a fade out followed by a fade in, cuts and dissolves, respectively. Then, in contrast to the other approaches that apply global or local thresholds to detect the shot boundaries, Taskiran and Delp pose the problem as a one dimensional edge detection and apply a method based on mathematical morphology.

Patel and Sethi [PATEL] use only the DC components of I frames. In [PATEL1] they compute the intensity histogram for the DC term images and compare them using three different statistics: Yakimovski likelihood ratio, χ^2 test and Kolmogorov-Smirnov statistics. The experiments show that χ^2 test gives satisfactory results and outperforms the other techniques. In their consequent paper [PATEL1] , Patel and Sethi compare local and global histograms of consecutive DC term images using χ^2 test, Figure III.12. The local row and column histograms X_i and Y_j are defined as follows:

$$X_i = \frac{1}{M}\sum_{j=1}^{M} b_{0,0}(i,j)\ ,\ Y_j = \frac{1}{N}\sum_{j=1}^{N} b_{0,0}(i,j) \qquad \text{(III.28)}$$

where $b_{00}(i,j)$ is the DC term of block (i,j), *i=1..N, j=1..M.* The outputs of the χ^2 test are combined using majority and average comparison in order to detect abrupt and gradual transitions.

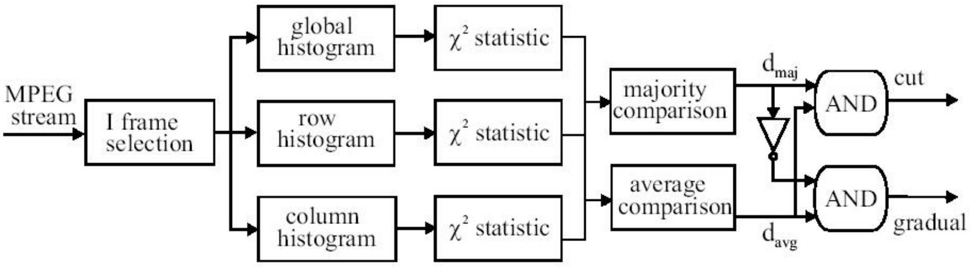


Figure III.12 Video shot detection scheme of Patel and Sethi. Figure taken from [PATEL].

As only I frames are used, the DC recovering is eliminated. However, the temporal resolution is low as in a typical GOP every 12th frame is an I frame and, hence, the exact shot boundaries cannot be labelled.

Meng, Juan and Chang [MENG] propose a shot boundaries detection algorithm based on the DC terms and the type of MB coding, Figure III.13. DC components only for P frames are reconstructed. Gradual transitions are detected by calculating the variance σ^2 of the DC term sequence for I and P frames and looking for parabolic shapes in this curve. This is based on the fact that if gradual transitions are linear mixture of two video sequences f_1 and f_2 with intensity variances σ_1 and σ_2, respectively, and are characterized by $f(t) = f_1(t)[1 - \alpha(t)] + f_2(t)$ where $\alpha(t)$ is a linear parameter, then the shape of the variance curve is parabolic: $\sigma^2(t) = (\sigma^2_1 + \sigma^2_2)\alpha(t) - \sigma^2_1\, \alpha(t) + \sigma^2_1$. Cuts are detected by the computation of the following three ratios:

$$R_p = \frac{\text{intra}}{\text{forw}},\ R_b = \frac{\text{back}}{\text{forw}},\ R_f = \frac{\text{forw}}{\text{back}} \tag{III.29}$$

where intra, forw, and back are the number of MBs in the current frame that are intra, forward and backward coded, respectively.

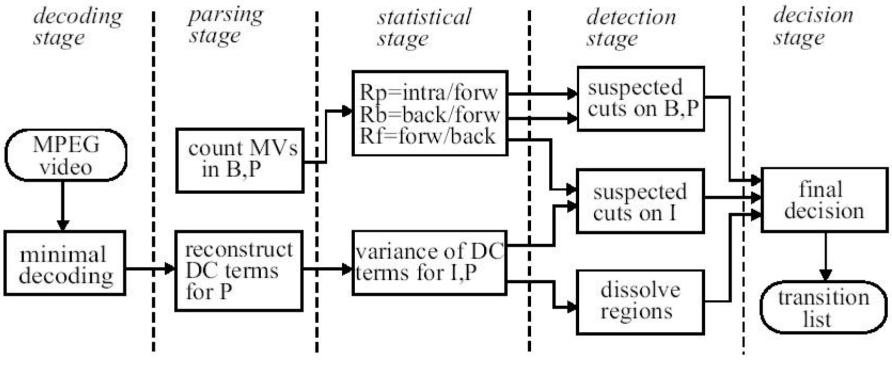


Figure III.13 Shot detection algorithm of Meng, Juan and Chang. Figure taken from [MENG].

If there is a cut on a P frame, the encoder cannot use many MBs from the previous anchor frame for motion compensation and as a result many MBs will be coded intra. Hence, a suspected cut on P frame is declared if R_p peaks. On the other hand, if there is a cut on a B frame, the encoding will be mainly backward. Therefore, a suspected cut on B frame is declared if there is a peak in R_b. An I frame is a suspected cut frame if two conditions are satisfied: 1) there is a peak in $|\Delta\sigma^2|$ for this frame and 2) the B frames before I have peaks in R_f. The first condition is based on the observation that the intensity variance of the frames during a shot is stable, while the second condition prevents false positives due to motion. This technique is relatively simple, requires minimum encoding and produces good accuracy. The total number of parameters needed to implement this algorithm is 7.

A technique by Fernando, Canagarajah and Bull [FERNA] stands out as a unified approach to scene change detection in both compressed and uncompressed domain. In this framework, an efficient algorithm estimates statistical features from the MPEG-2 compressed domain. These features can be computed from the uncompressed domain as well. The statistic properties of each image are used to identify special effects that create gradual transitions like fades, dissolves and wipes. A transition model based on the image properties is created for each type of transition. The reported results show high precision/recall values of the scene change detection, approximately at the same level as for the proposed algorithm. However, the amount of processing needed for calculation of the statistical properties for each image is much bigger in comparison to the techniques that utilise only MB coding type information due to the partial decompression needed. This conclusion can be generalised to all methods that involve DCT coefficient analysis. In order to keep the continuity and process every frame in the sequence, partial motion compensation has to be done. Therefore, this substantial processing put methods that exploit DCT coefficient information somewhere between uncompressed and compressed domain analysis when it comes to their efficiency.

III.3.5.2. Temporal Video Segmentation Based on Motion Vectors

Trying to minimise this partial decompression, a two-pass approach is presented by Zhang, Low and Smoliar [ZHANG3]. Here, the regions of potential transitions are located first applying the pair-wise DCT coefficients comparison of only I frames as in their previous approach.

The goal of the second pass is to refine and confirm the break points detected by the first pass. By checking the number of MVs M for the selected areas, the exact cut locations are detected. If M denotes the number of MVs in P frames and the smaller of the numbers of forward and backward nonzero MVs in B frames, then M<T (where T is a threshold close to zero) is an effective indicator of a cut before or after the B and P frame. Gradual transitions are found by an adaptation of the twin comparison algorithm utilizing the DCT differences of I frames. By MV analysis, though using thresholds, false positives due to pan and zoom are detected and discriminated from gradual transitions.

Thus, the second pass of the algorithm uses only information directly available in the MPEG stream. It offers higher processing speed due to the multipass strategy, good accuracy and also detects false positives due to pan and zoom. However, the metric for cut detection yields false positives in the case of static frames and the efficiency is worse than of the proposed algorithm. Also, the problem of how to distinguish object movements from gradual transitions is not addressed.

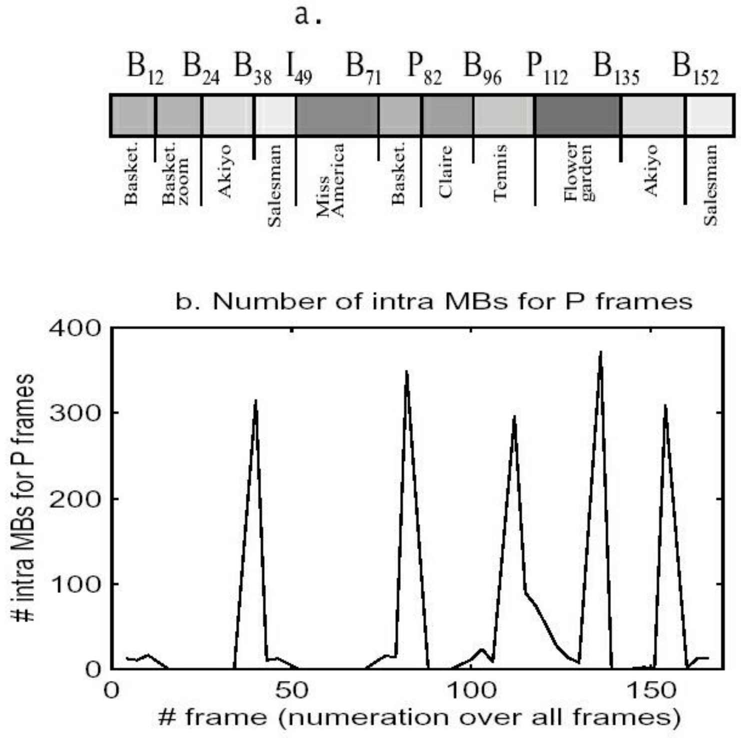


Figure III.14 Cuts: a) video structure, b) number of intra coded MBs

In [KOPRI] cuts, fades and dissolves are detected only using MVs from P and B frames and information about MB coding mode. The system follows a two-pass scheme and has a hybrid rule-based/neural structure. During the rough scan peaks in the number of intra coded MBs in P frames are detected. They can be sharp (Figure III.14) or gradual with specific shape (Figure III.15) and are good indicators of abrupt and gradual transitions, respectively. The solution is then refined by a precise scan over the frames of the respective neighbourhoods. The "simpler" boundaries (cuts and black fade edges) are recognized by the rule-based module, while the decisions for the "complex" ones (dissolves and non-black fade edges) are taken by the neural part. The precise scan also reveals cuts that remain hidden for the rough scan, e.g. B_{24}, I_{49}, B_{71} and B_{96} in Figure III.14.

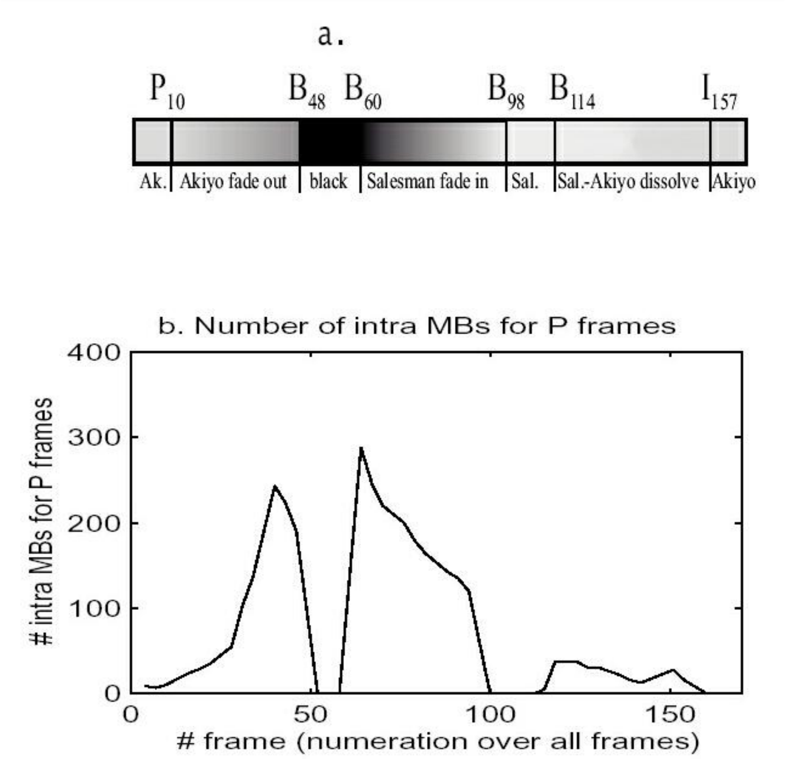


Figure III.15 Fade out, fade in, dissolve: a) video structure, b) No. of intra coded MBs for P frames

The rules for the exact cut location are based on the number of backward and forward MBs while those for the fades black edges detection use the number of interpolated and backward coded MBs. There is only one threshold in the rules that is easy to set and not sensitive to the type of video. The neural network module learns from pre-classified examples in the form of MV patterns corresponding to the following 6 classes: stationary, pan, zoom, object motion, tracking and dissolve. It is used to distinguish dissolves from object and camera movements, find the exact location of the "complex" boundaries of the gradual transition and further divide shots into sub-shots. The approach is simple, fast, robust to camera operations and very accurate when detecting the exact locations of cuts, fades and simple dissolves. The experimental results show high accuracy of this method, but efficiency-wise, due to the neural approach, this algorithm doesn't fully use the opportunity of the compressed domain features. In the next session, a group of algorithms that utilise direct access to the compressed domain features is presented.

III.3.5.3. Temporal Video Segmentation Based on MB Coding Mode

Although limited only to cut detection, a simple and effective approach is proposed in [WEISS]. It only uses the bitrate information at MB level and the number of various motion predicted MBs. A large change in bitrate between two consecutive I or P frames indicates a cut between them. As well as the proposed algorithm, this method only exploits MB coding type information, but lacks robustness to camera and object motion and gradual transitions. In order to solve this problem, Meng *et.al.* [MENG] analyses a relation between the number of backward predicted and motion compensated MBs for detecting cuts on B frames. Here, the ratio is calculated as R_b = back/mc where back and mc are the number of backward and all motion compensated MBs in a B frame, respectively. The algorithm is able to locate the exact cut locations. It operates hierarchically by first locating a suspected cut between two I frames, then between the P frames of the GOP and finally (if necessary) by checking the B frames. Following the similar idea, work presented by Dawood and Ghanbari [DAWOO] compares the way frames are referenced by exploiting the MB coding types. It assumes the standard GOP structure of MPEG stream, as it is assumed in this work. However, the algorithm proposed in this thesis is more robust to gradual transitions and camera motion, due to the fact that it calculates the continuous frame difference metric based upon the prediction behaviour within a whole SGOP.

III.3.5.4. Comparison of Algorithms for Temporal Video Segmentation in Compressed Domain

In [GARGI] the approaches of Arman et al.[ARMAN], Patel and Sethi [PATEL], Meng et al.[MENG], Yeo and Liu [YEO] and Shen and Delp [SHEN] are compared along several parameters: classification performance (recall and precision), full data use, ease of implementation, source effects. Ten MPEG video sequences containing more than 30 000 frames connected with 172 cuts and 38 gradual transitions are used as an evaluation database. It is found that the algorithm of Yeo and Liu and those of Shen and Delp perform best when detecting cuts. Although none of the approaches recognizes gradual transitions particularly well, the best performance is achieved by the last one. As the authors point out, the reason for the poor gradual transition detection is that the algorithms expect some sort of ideal curve (a plateau or a parabola) but the actual frame differences are noisy and either do not follow this ideal pattern or do not do this smoothly for the entire transition. Another interesting conclusion is that not processing of all frame types (e.g., like in the first two methods) does decrease performance significantly. The algorithm of Yeo and Liu is found to be easiest for implementation as it specifies the parameter values and even some performance analysis is already carried out by the authors. The dependence of the two best performing algorithms on bitrate variations is investigated and shown that they are robust to bitrate changes except at very low rates. Finally, the dependence of the algorithm of Yeo and Liu on two different software encoder implementations is studied and significant performance differences are reported.

Compared to the algorithm proposed here, most of the techniques that work in both uncompressed and compressed domain lack efficiency considerably. Yet only a few achieve better overall accuracy and robustness. On the other hand, algorithms that access compressed domain features without additional processing and thus having the similar efficiency, underperformed in the accuracy and robustness criteria.

III.4. KEY-FRAME EXTRACTION TECHNIQUES

Key-frames are still images extracted from original video data that best represents the content of the shot in an abstract manner. Key-frames have been frequently used to supplement the text of a video log, but identifying them was done manually in the past. The effectiveness of key-frames depends on how well they are chosen from all frames of a sequence. The image frames within a sequence are not all equally descriptive. Certain frames may provide more information about the objects and actions within the clip than other frames. In some prototype systems and commercial products, the first the first frame of each shot has been used as the only key-frame to represent the shot content. However, while such a representation does reduce the data volume, its representation power is very limited since it often does not give a sufficient clue as to what actions are presented by a shot, except for shots with no change or motion.

Key-frame-based representations views video abstraction as a problem of mapping an entire segment (both static and motion content) to some small number of representative images. The challenge is that the extraction of key-frames needs to be automatic and content-based so that they maintain the important content of the video while removing all redundant information. In theory, semantic primitives of video, such as interesting objects, actions, and events should be used. However, such general semantic analysis is not currently feasible, especially when information from soundtracks and/or closed caption is not available. In practice, we have to rely on low-level image features and readily available information instead.

An effective approach to key-frame extraction, based on temporal variation of low-level image features such as colour histograms and motion information, has been proposed by Zhang, et. al. [ZHANG1]. The key idea of this approach is that the number of key-frames needed to represent a segment should be based on temporal variation of video content in the segment; if there is a large temporal variation of content, there should be more key-frames, and vice versa. That is, after shot segmentation, key-frames in a shot will be selected based on the amount of temporal variation of colour histograms and motion in reference to the first or the last selected key-frame of the shot. It is reported that this approach achieves real-time processing speed, especially when MPEG compressed video and reasonable accuracy is used.

In more detail, in this approach, frames in the shot will be compared in terms of colour histogram changes against the last key-frame or the first frame of the shot sequentially as they are processed, based on their similarities defined by colour histogram. If a significant change occurs, the current frame will be selected as a key-frame. Such a process will be iterated until the last frame of the shot is reached. In this way, any significant action in a shot will be captured by a key-frame, while static shot will result in only one key-frame. In addition, information of dominant or global motion resulting from camera motion and large moving objects is added into the selection process according to a set of rules. For a zooming like (zooming, dollying, and perpendicular motion of large objects) sequence, the first and the last frames will be selected as key-frames; one presents a global, and the other the more focused view. For panning like (panning, tilting and tracking) sequence, the number of frames to be selected will depend on the scale of panning: ideally the spatial content covered by each frame should have little overlap, or each frame should capture a different, but sequential part of object activities.

Figure III.16 shows an example in which three key-frames from a zoom-in shot were extracted using this approach. One can see clearly that it is a zoom-in sequence, which will not be concluded reliably from any single key-frame. In this respect, extracting three key-frames is a more adequate than only a single key-frame, which is important for users (especially producers and editors) who want to choose some particular of shots from stock footages.

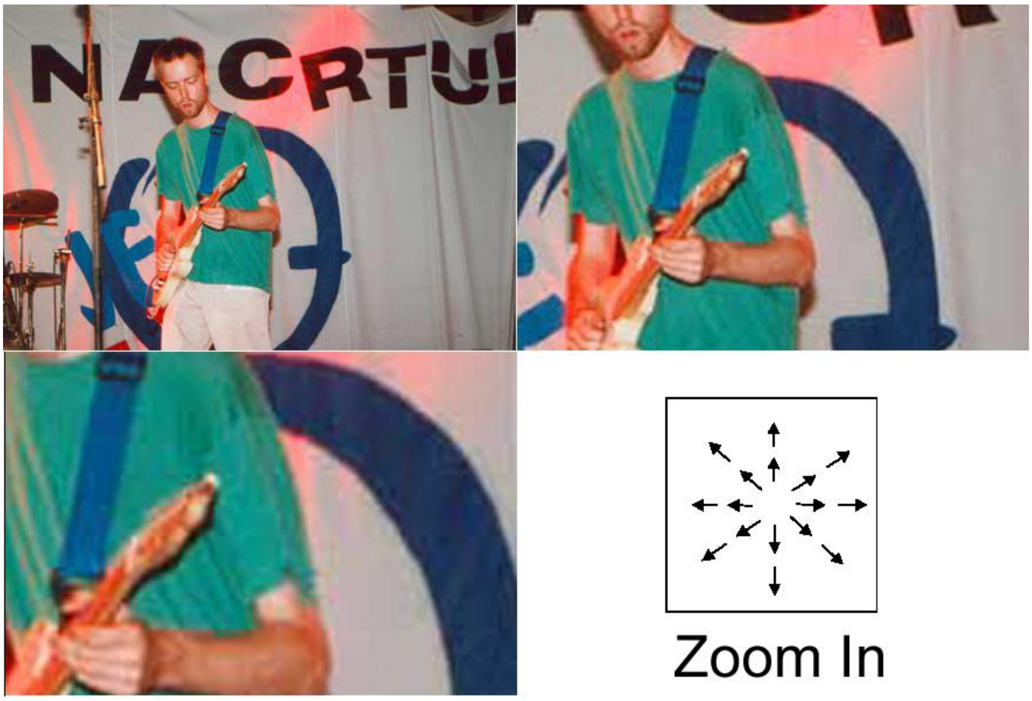


Figure III.16 Three frames showing an example of a zoom-in sequence

In this approach, the density of key-frames or the abstraction ratio can be controlled according to the user's need by adjusting the threshold for determining "significant" colour histogram changes and the overlap ratio of key-frames in panning sequences. However, the exact number of resultant key-frames will be determined *a posteriori* by the actual content of the input video. This fact has been argued to be a disadvantage of this type of key-frame extraction approach. On the other hand, predefining the absolute number of key-frames without knowing the content of video may not be desirable; assigning two key-frames for a talking head sequence of 30 minutes should still be considered having too much redundancy! In addition, assigning the same number of key-frames t, for instance, two video sequence of same length does not guarantee the same level of visual abstraction since the contents of the two sequences may have different levels of abstraction and/or totally different level of activities. Therefore, controlling the level of abstraction ratio or key-frame density is a more robust and useful approach.

A compromise to meet the need of having a predefined number of key-frames, while maintaining the content-based selection criteria and constant level of abstraction ratio among a given set of video sequences, is to set up a maximum number of key-frames allowed. In this way, an initial set of key-frames can be selected at a given abstraction ratio using the approach discussed above. Then, if the number of key-frames exceeds the maximum, a post-filtering can be applied to filter out the frames whose similarity to their immediate neighbouring two frames are high.

The key-frame extraction approach described above is based on a frame-based representation. That is, each frame is considered the basic unit for content representation. However, if we could further decompose frames into key-objects, then key-frames can be extracted based on the motion or activity of the objects. Below we outline some strategies for key-frames selection based on the motion activity and attributes of key-objects within the shot:

- a key-object enters or leaves the image frame boundaries
- key-object participate in occlusion relationship
- two key-objects are at the closest distance between them
- mean and extremes of key-object attributes, i.e., colour, shape, motion…
- key-frames should have some small amount of background object overlap

Figure III.17 shows 3 frames selected from the video sequence according to the criteria outlined above.

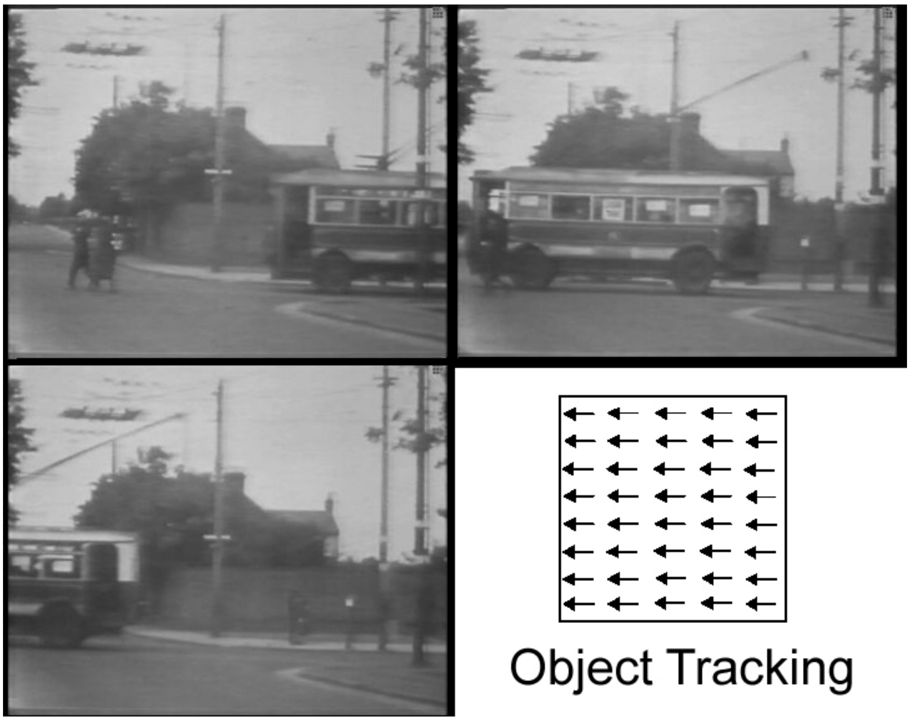


Figure III.17 Three frames depicting a trolley-bus entering the scene, being in the middle, and leaving the scene as events crucial to the object tracking based summarisation.

III.5. REPRESENTING COLOUR IN CBVIR

Colour is perhaps the most expressive of all the spatial visual features and has been extensively studied in the image and video retrieval research during the last decade. This Section presents state-of-the-art colour feature analysis in the CBVIR area. In a CBVIR system, once the key-frames are extracted as the representative set of images for a given video sequence, a set of low-level spatial features is extracted as a low-level description of the video sequence. These features include colour, texture, edges, shape, etc. The focus here is on the most expressive and widespread spatial feature: colour.

III.5.1. COLOUR HISTOGRAMS

In order to describe the variety of colours present in an image, the most suitable description is the colour distribution in the form of a colour histogram. A colour histogram is organised into a number of bins that represent non-overlapping colour ranges. Each bin contains the number of pixels that fall within each colour range. The histogram allows images with similar colour distributions to be retrieved. Colour descriptors originating from histogram analysis have played a central role in the development of visual descriptors in CBVIR.

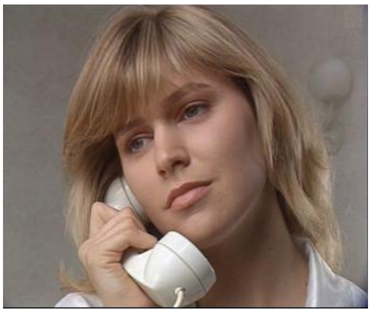


Figure III.18 Sample Colour Image suzie.jpg

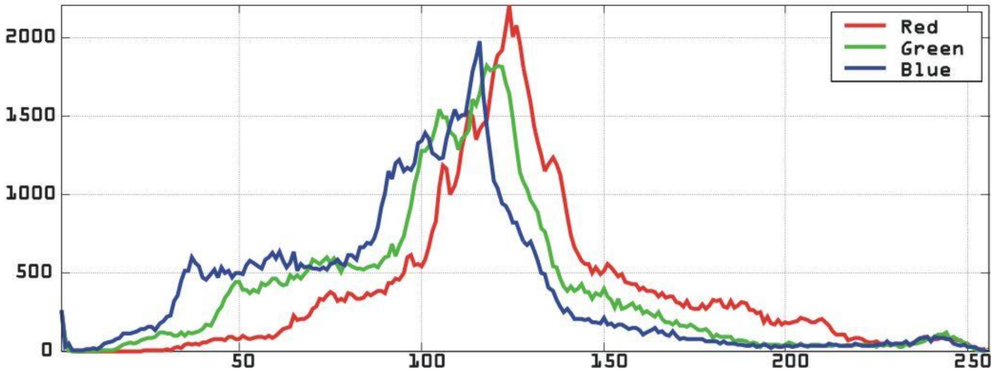


Figure III.19 Colour histogram of the image above

III.5.2. COLOUR HISTOGRAM COMPARISON

There are a number of ways to compare histograms. Two simple methods include the absolute difference between two histograms, as in Equation (III.30), or the Euclidean distance as in Equation (III.31). In these two cases a lower distance value represents a greater similarity between images.

$$d_{RGB}\left(I_i,I_j\right)=\sum_{k=1}^{n}\left(\left|H_i^r(k)-H_j^r(k)\right|+\left|H_i^g(k)-H_j^g(k)\right|+\left|H_i^b(k)-H_j^b(k)\right|\right) \tag{III.30}$$

$$d_{RGB}^2\left(I_i,I_j\right)=\sum_{k=1}^{n}\left(\left(H_i^r(k)-H_j^r(k)\right)^2+\left(H_i^g(k)-H_j^g(k)\right)^2+\left(H_i^b(k)-H_j^b(k)\right)^2\right) \tag{III.31}$$

Another method for comparing histograms is to use the histogram intersection [SWAIN]. The histogram intersection adds up the minimum values from each corresponding bin in the histograms. Two images are considered similar if they have a large intersection. The intersection is then divided by the total number of pixels in the second image to normalise the value. A disadvantage with these approaches is that the computational complexity depends linearly on the product of the size of the histogram and the size of the database. Only comparing the bins with the largest number of pixels can reduce the complexity. Swain combined this technique with histogram intersection to perform an incremental intersection. Using incremental intersection the computational complexity can be reduced from O(nm) to O(n log(n) + cm), where c is the number of bins to compare from each histogram.

$$d\left(I_i,I_j\right)=\frac{\sum_{k=1}^{n}\min\left(H_i(k),H_j(k)\right)}{\sum_{k=1}^{n}H_j(k)} \tag{III.32}$$

Another problem with these histogram comparison techniques is that bins are not compared with adjacent bins that may represent perceptually similar colours. The QBIC (Query by Image Content) [NIBLA] system uses the colour histogram cross-distance, which considers the cross-correlation between histogram bins based on perceptual similarity expressed in Equation (III.33). The cross-correlation is determined by a matrix with entries a_{pq}. When the matrix is an identity matrix the formula becomes the Euclidean distance.

$$d\left(I_i,I_j\right)=\sum_{p=1}^{n}\sum_{q=1}^{n}\left(H_i(p)-H_j(p)\right)\cdot a_{pq}\cdot\left(H_i(q)-H_j(q)\right) \tag{III.33}$$

Stricker and Orengo [STICKER] argue that the problem is not with histogram comparison techniques but with the formulation of the histogram. They propose a cumulative histogram where each bin Ci in the cumulative histogram is the sum of all bins $H_{j\leq i}$ in the colour histogram. However, their results do not show a significant improvement over standard colour histograms.

In addition to the cumulative histogram Stricker and Orengo propose using central moments to describe the features of the histogram rather than the histogram itself. Moments have the general form:

$$M_n = \sum \frac{\left(x - \overline{x}\right)^n}{N} \qquad \text{(III.34)}$$

where N is the number of data points and n is the order of the moment. The first moment is related to the mean, the second relates to the variance, the third determines the skewness and the fourth can be used to calculate kurtosis. Stricker and Orengo use the following formulae to determine mean, variance and skewness:

$$E_i = \frac{1}{N}\sum_{j=1}^{N} p_{ij}\ ,\sigma_i = \left(\frac{1}{N}\sum\left(p_{ij} - E_i\right)^2\right)^{\frac{1}{2}}\ ,s_i = \left(\frac{1}{N}\sum\left(p_{ij} - E_i\right)^3\right)^{\frac{1}{3}} \qquad \text{(III.35)}$$

where p_{ij} is the j-th pixel of the i-th colour channel. The moments for each colour channel are stored separately resulting in only nine floating-point numbers per image. The similarity between two image entries can be determined using the similarity function d_{mom}:

$$d_{mom}\left(H,I\right) = \sum_{i=1}^{r} w_{i1}\cdot\left|E_i - F_i\right| + w_{i2}\cdot\left|\sigma_i - \zeta_i\right| + w_{i3}\cdot\left|s_i - t_i\right| \qquad \text{(III.36)}$$

The weights, w_{il}, allow varying emphasis to be placed on different moments. For example, an indoor scene may have non-varying lighting conditions so more importance may be placed on the average colour because the average colour should not change considerably between shots with similar lighting.

III.5.3. SELECTION OF COLOUR SPACE

Images can be faithfully reproduced using an additive RGB colour space because the photoreceptors in the eye, which loosely represent the red, green, and blue wavelengths, combine their outputs so that all colours of the visible spectrum can be perceived. However, analysing images based on the RGB colour space does not always give perceptually accurate results. The human vision system doesn't see colours as three separate dimensions ranging from black to red, green or blue. Rather colours are interpreted on a colour wheel (Figure III.20) where each colour mixes into the next and completes a circle. Colour is then best represented as the angle on the wheel rather than individual strengths of red, green or blue. To represent all visible wavelengths a colour solid or colour spindle [MATLIN] can be constructed (Figure III.21). The colour spindle allows colour to be represented in terms of hue, saturation, and brightness.

A colour system designed to imitate human colour perception is the Munsell colour coordinate system that has the three components hue, value and chrominance (HVC colour space) [MIYAH].

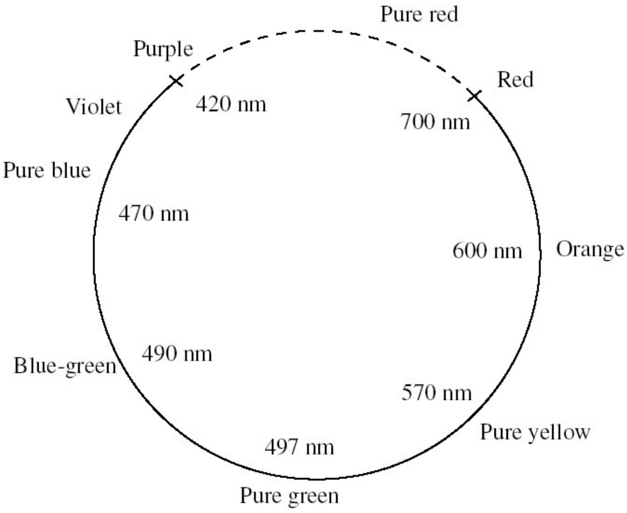


Figure III.20 Colour circle

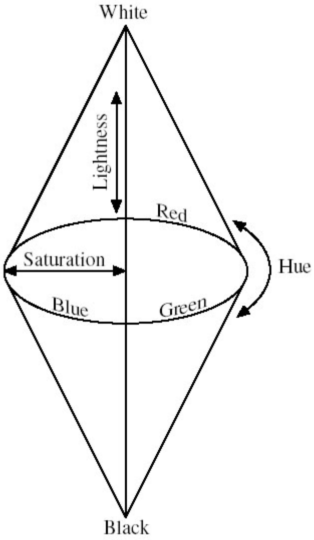


Figure III.21 Colour cone

Gong [GONG] calculates an approximation to the Munsell colour coordinate system by converting the RGB values into CIE XY Z values using the formulae

$$\begin{aligned} X &= 0.607 \cdot R + 0.174 \cdot G + 0.201 \cdot B \\ Y &= 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \\ Z &= 0.066 \cdot G + 1.117 \cdot B \end{aligned} \tag{III.37}$$

Equations (III.38), (III.39), and(III.40) show how the L*a*b* values can then be obtained from the X Y Z values, where X_o, Y_o, and Z_o represent the X, Y, and Z values for the reference white.

$$L^* = 116 \cdot \left(\frac{Y}{Y_o} \right)^{\frac{1}{3}} - 16 \tag{III.38}$$

$$a^* = 500 \cdot \left[\left(\frac{X}{X_o} \right)^{\frac{1}{3}} - \left(\frac{Y}{Y_o} \right)^{\frac{1}{3}} \right] \tag{III.39}$$

$$b^* = 200 \cdot \left[\left(\frac{Y}{Y_o} \right)^{\frac{1}{3}} - \left(\frac{Z}{Z_o} \right)^{\frac{1}{3}} \right] \tag{III.40}$$

Finally the H V C values can be derived from the L*a*b* values

$$H = \arctan\left(b^*/a^*\right) \tag{III.41}$$

$$V = L^* \tag{III.42}$$

$$C = \sqrt{\left(a^*\right)^2 + \left(b^*\right)^2} \tag{III.43}$$

Determining the HVC values from RGB can be a difficult process as can be seen with the preceding formulas. Smith and Chang [SMITH] used a more tractable transform to HSV colour space. The algorithm assumes input in the range $R,G,B \in 0 \rightarrow 1$ and produces output, $H \in 0 \rightarrow 6$ and $S,V \in 0 \rightarrow 1$. Even though much faster to compute, the HSV colour space is not as perceptually accurate as the HVC colour space.

Other colour spaces that are suited for compression, such as YUV and XYZ, use opponent colour axes. Swain and Ballard [SWAIN] used the opponent colour axes that are defined as:

$$rg = r - g \tag{III.44}$$

$$by = 2 \times b - r - g \tag{III.45}$$

$$wb = r + g + b \tag{III.46}$$

Even though this colour space isn't perceptually uniform it can be computed quickly and allows the intensity axis (wb) to be more coarsely sampled to reduce the effects of lighting variation.

III.5.4. COLOUR SPACE QUANTISATION

With the intent to make the colour analysis simpler, the continuous colour space is in the most cases quantised into a partitioned and discrete colour space. In general, quantizer Qc is a mapping vector of dimension k and size M that transforms from a vector in k dimensional colour space into a finite set C containing M outputs [GERSHO]. Thus, a quantizer Qc is defined as:

$$Q_c : \Re^k \longrightarrow C, C = \left\{ y_0, y_1, \ldots, y_{M-1} \right\} \wedge \forall m \in \left\{ 0,1,\ldots M \right\}, y_m \in \Re^k \tag{III.47}$$

In general, the set C is called the codebook and has size M. In the case of colour space quantisation, k=3 and each entry in the codebook y_m corresponds to a colour vector. Therefore, the codebook C represents the gamut or collection of colours. The quantizer Qc is a covering of $\Re^k$ into M partitions, where each partition R_m contains all points w_c in the continuous colour space that are assigned the same codeword y_m:

$$R_m = w_c \in \Re^k : Q_c(w_c) = y_m \qquad \text{(III.48)}$$

From the definition of the partitions it follows that they completely cover $\Re^k$ and are non-overlapping:

$$\bigcup_m R_m = \Re^k \quad \wedge \quad R_m \bigcap R_n = \varnothing, \forall m \neq n \qquad \text{(III.49)}$$

so that these partitions form a complete partitioning of $\Re^k$.

Practically, since the colour histograms are already a discrete representation of the continuous colour space giving the colour distribution of the analysed image/key-frame, one-dimensional colour space quantisation is automatically done by generation of the colour histogram with M bins.

III.5.5. HISTOGRAM QUANTISATION

In addition to the colour space quantisation, CBVIR systems use the colour histogram quantisation to manage the retrieval process in a scalable and hierarchical way. This process is sometimes referred to as colour histogram compaction. The best example of the quantisation in the descriptor domain is the Scalable Colour Descriptor of (SCD) defined in the MPEG-7 standard.

The SCD addresses the interoperability issue by fixing the colour space to HSV, with a uniform quantization of the HSV space to 256 bins. The bin values are non-uniformly quantized to a 11-bit value. This method achieves full interoperability between different resolutions of the colour representation, ranging from 16 bits/histogram at the low end to approximately 1000 bits/histogram at the high end. Of course, the accuracy of the feature description is highly dependent on the number of bits used. However, core experiments have shown that good retrieval results are still achievable using only 64 bits, while excellent results can be obtained using medium or full resolution of the descriptor.

The HSV space is uniformly quantized into a total of 256 bins. This includes 16 levels in H, four levels in S, and four levels in V. The histogram values are truncated into a 11-bit integer representation. To achieve a more efficient encoding, the 11-bit integer values are first mapped into a "nonlinear" 4-bit representation, giving higher significance to the small values with higher probability. This 4-bit representation of the 256-bin HSV histogram would require 1024 bits/histogram, which is too large a number in the context of many MPEG-7 applications. To lower this number and make the application scalable, the histograms are encoded using a Hadamard transform. The basic unit of the Hadamard transform consists of a sum operation and a difference operation [see Figure III.22(A)], which relate to primitive low- and high-pass filters.

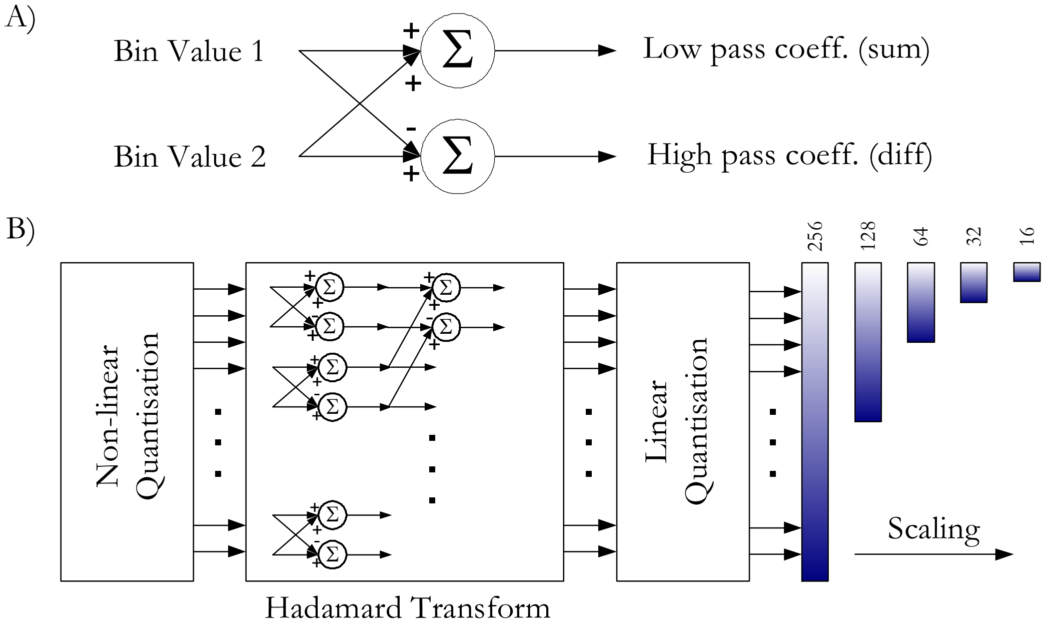


Figure III.22 A) Basic unit of Hadamard transform, B) A schematic diagram of SCD generation

Summing pairs of adjacent histogram lines is equivalent to the calculation of a histogram with half number of bins. If this process is performed iteratively, usage of subsets of the coefficients in the Hadamard representation is equivalent to histograms of 128, 64, 32 bins, which are all calculated from the source histogram. The high-pass (difference) coefficients of the Hadamard transform express the information contained in finer-resolution levels (with higher number of bins) of the histogram. This procedure relies o the assumption that natural image signals usually exhibit high redundancy between adjacent histogram lines. This can be explained by the "impurity" (slight variation) of colours caused by variable illumination and shadowing effects. Hence, the high-pass coefficients expressing differences between adjacent histogram bins usually have only small values. Exploiting this property, possibility to truncate the high-pass coefficients to integer representation with only a low number of bits is claimed. Figure III.22(B) shows the block diagram of the complete system.

The output representation is scalable in terms of numbers of bins, by varying the number of coefficients used. Interoperability between different resolution levels is retained due to the scaling property of the Hadamard transform. Thus, matching based on the information from subsets of coefficients guarantees an approximation.

Table III.1 shows the relationship between numbers of Hadamard coefficients as specified in the SCD and partitions in the components of a corresponding HSV histogram that could be reconstructed from the coefficients. A different type of scalability is achieved by scaling the quantized (integer) representation of the coefficients to different numbers of bits.

SCALING	H	S	V
16	4	2	2
32	8	2	2
64	8	2	4
128	8	4	4
256	16	4	4

Table III.1 Equivalent Partitioning of the HSV colour Space for different configurations of the MPEG-7 Scalable Colour Descriptor

Although this method argues that the histogram simplification procedure gradually removes least relevant information first, it doesn't establish its argument on the perceptual distortion to the image but on the distortion of the colour histogram. In Chapter IV a colour histogram simplification algorithm based on the perceptual distortion of the represented image is given.

III.6. SEMANTIC EFFORTS IN CBVIR

Numerous CBIR systems have explored the possibilities of indexing image and video content by using low-level visual features (see Virage [BACH], QBIC [FLICK], and VisualSeek [SMITH2]). These systems work by (1) automatically extracting features directly from the visual data; (2) indexing the extracted descriptors for fast access; and (3) querying and matching descriptors for the retrieval of the visual data. Beyond these basic capabilities, there has been an effort to support relevance feedback to refine queries and learn through examples what the user may be looking for [RUI].

More recently, there has been focused effort on automatically producing certain semantic labels that could contribute significantly to retrieving visual data. For example, recent work has focused on portrait vs. landscape detection, indoor vs.

outdoor classification, city vs. landscape classification, sunset vs. forest classification [SZUMMER], [VAILAYA], and other attempts to answer basic questions of who, what, when, and where about the visual content. Most of the approaches rely on traditional machine learning techniques to produce semantic labels, and some degree of success has been reached for various constrained and sometimes skewed test sets. However, these efforts represent only a small initial step towards achieving the real understanding of the visual content.

III.6.1. SEMIOTIC THEORIES

Semiotics is a discipline that studies the relationships between signs and their meanings and provides a sound framework for the automatic extraction of semantics in video streams from recognition of basic visual and audio primitives and their combination according to a suitable set of rules [SANTINI]. In the following section we discuss the semiotic perspective to derive meaning from signs in video. Frequent buzzwords like text and language are to be placed in the context of the film theory although the semiotic approach at this level doesn't involve any particular medium.

Semiotics is a theoretical framework for the study of meaning in film, TV and other media, identifies underlying structure of their symbolic values, their use and interpretation. The term semiotics stems from the Greek word *semeiotikos*, which denotes the study of signs, what they represent and signify, and how we act and think in their universe. Founder of semiotics and modern linguistics, Ferdinand de Saussure [SAUSS], argued first that language is a system, in which it is the relations between elements and not elements themselves that are responsible for meaning. Humans make meanings through creation and interpretation of signs as mental concepts, with a signifier as its material aspect (letter, icon, sound, etc.). Sign has an arbitrary nature of the bond between signifier and signified, so that the sign signifies by virtue of its difference from other signs. Structuralism, an analytical method based on Saussure's linguistic model, has been employed by many semioticians that described the overall organization of sign systems as languages with their grammars. They engaged in a search for deep structures underlying the 'surface features' of phenomena.

Structuralism produced the first semioticians of the film language. Initial attempts to analyse the underlying structure of signs in visual media were made in the 1950s by Rolan Barthes on photography and in late 1960s by Christian Metz on cinema [METZ]. Metz examined the ways in which film could be considered as language, the nature of a shot opposed to the word, and what the grammar of cinematic narrative might be. He developed a classification of sequences and scenes known as la grande syntagmatique, based on editing strategies and their role in conveying narrative form. By identifying five levels of cinematic codification that create basic significations in a film, i.e. perceptual, cinematic, diegetic, connotative and subtextual level, Metz sowed the seeds of computational semantic and semiotic analysis in visual media.

Much earlier, in 1920s, Soviet filmmakers Eisenstein and Pudovkin introduced the fact that editing is the crucial expressive element in cinema. Kuleshov [KULESH] experiment proved that meaning appears to derive from the relationship of contiguous shots rather than from the content of the individual shots themselves.

Almost concurrently with structuralism in film theory, in the years immediately preceding the student uprising in 1968, a new semiotic theory was emerging. It brought fundamental critics of Western philosophy in general by returning the crucial role of the human subject in signification process. It was called post-structuralism. As the creator of the most influential method in post-structuralism, Deconstruction, Jacques Derrida gave definite consequences for the human's relation to the system of representations [DERRIDA], central to our problem of multimedia management.

Deconstruction states that it is impossible for a text to have one fixed meaning, and emphasizes the role of the observer in the production of meaning. Language does not reflect meanings, which pre-exist in the world; it is the site for the production of meaning. This appeared to be the case for all signifying practices like film, television and other media that we are eager to explore.

Structuralism claimed to provide a scientific method that located unity and order in the underlying structures of texts but assumed that analysts' meanings coincided with those of the observer. Deconstruction is, though, characterised by a shift away from the determining structures of texts, a concern with signifiers as against signs, and a foregrounding of the role of the observer in the process of producing meaning [BRUNET].

Clearly structured analysis enabled breaking down the problem complexity to the level bearable for computational implementation. Structuralistic approach to computational problems tempted its technical developers: it offered straightforward and stable solution to the complex problem of multimedia semantics. On the other hand, deconstruction offered undecidability and deep involvement in interaction with the human subject.

The main idea of deconstruction is in a way analogous to the Werner Heisenberg's uncertainty principle which asserts that at the quantum level the observer effects that which is observed, thus making truly objective observation impossible. Subject of observation is constantly disturbing the signification chain from its equilibrium giving the new paths to the meaning.

III.6.2. SEMIOTICS IN CBVIR

There has been some recent work connecting the fields of semiotics and multimedia information systems. Smoliar et al. [SMOLIAR] describes some of the implications to multimedia search from the point of view of writing and reading multimedia signs. Multimedia material such as images and words are considered to signify notions of objects in the world (e.g., an image of a carrot and the word "carrot" signify the notion of carrot); and search, fundamental for the processes of reading and writing. Joyce et al. [JOYCE] proposes a semiotics framework to integrate high-level metadata (e.g. "carrot") and low-level metadata (e.g. colour histogram extracted from the image of a carrot) by formally adding a second representation level to the [SMOLIAR]. This level consists of features extracted from the multimedia material acting as signs of the multimedia material itself as depicted in Figure III.23.

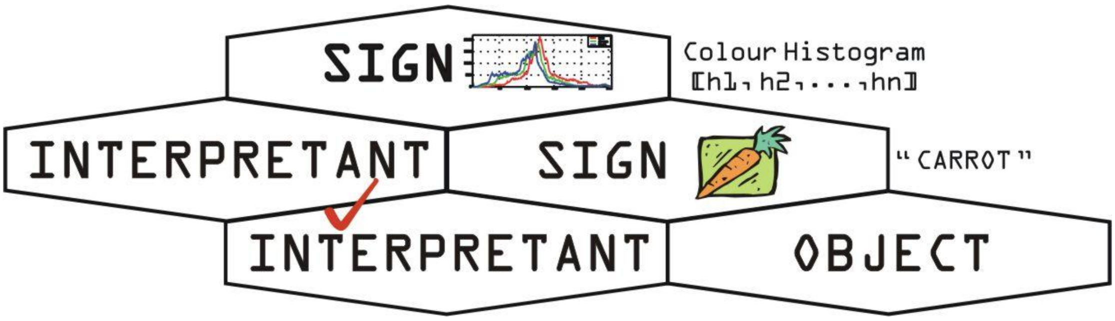


Figure III.23 Semiotic framework for multimedia and features extracted from multimedia

Textual and non-textual features signs are identified as high-level and low-level metadata, respectively. The link between the two is established with the Multimedia Thesaurus [TANSLAY] and neural-network classification agents [JOYCE]. Del Bimbo [DELBIM] applies the semiotics idea of producing meaning at two levels, the narrative level and discourse level, to automatically annotate and retrieve videos of commercials. The narrative level includes basic signs and the results of sign combinations; the discourse level describes how to use narrative elements to create a story.

Though the efforts to approach the task of semantic video indexing and retrieval from the semiotic perspective are radical change to the existing mainstream CBVIR systems, they all followed structuralistic theories in the process of signification. In order to involve the user's influence focus of the semiotic approach should move towards post-structuralist theories and let user and retrieval context shape the signification space on a deeper level. However, for that purpose, we need to develop appropriate representations that can adapt themselves to the environment.

III.6.3. RELEVANCE FEEDBACK

Early attempts in the filed of CBIR aimed at fully automated, open-loop system. It was hoped that current computer vision and image processing techniques would be good enough for image search and retrieval. The modest success rates experienced by such systems encouraged researchers to try a different approach, emphasizing interactivity and explicitly including the human user in the loop. An example of this shift can be seen in the work of MIT Media Lab researchers in this field, when they moved from the "automated" Photobook [PENTL] to the "interactive" FourEyes [MINKA].

Relevance feedback is a powerful technique first introduced in traditional text-based information retrieval systems. It is the process of automatically adjusting an existing query using the information fed back by the user about the relevance of previously retrieved objects such that the adjusted query is a better approximation to the user's information need [BUCKLEY] In an interactive system, neither the user nor the system designer need to specify any weights. The user only needs to mark which images he/she thinks are relevant to his/her query. The weights associated with the query object are dynamically updated to model the user's information need and perception subjectivity. In general, there are three approaches to relevance feedback in image and video retrieval. One is based on artificial intelligence (AI) learning techniques [PICARD], one on the probabilistic methods like a Bayesian framework [COX], and the last on information retrieval techniques [RUI]. Furthermore, this feedback might be provided in many different ways and each system might use it in a particular manner to improve its performance. The expected effect of relevance feedback is to "move" the query in the direction of relevant images (or any other media) and away from the on-relevant ones.

There are many ways of using the information provided by the user interactions and refining the subsequent retrieval results of a CBVIR system. One approach concentrates on the query phase and attempts to use the information provided by relevance feedback to refine the queries. Another option is to use relevance feedback information to modify feature weights, such as in the MARS project [RUI1]. A third idea is to use relevance feedback to construct new features on the fly, as exemplified in [MINKA1]. A fourth possibility is to use the relevance feedback information to update the probability of each image in a database being the target image, in other words, to predict the goal image given the user's interactions with the system [COX].

III.7. REPRESENTATIONS OF VIDEOS IN CBVIR

There are a few in-depth research efforts focused on the problem of knowledge representation of videos. The initial contribution to the area was made by M. Davis in [DAVIS] where an early definition of a video representation and the problems involved were presented. Here, four main ontological issues in video are outlined as: space, identity, action and time. Most of the arguments raised in this work referred to the experimental cinematic work of Lev Kuleshov [KULESH]. Davis underlines that the task in front of the researchers is to gather insights from disciplines that have studied the structure and function of video data and to use these insights in the design of new representations for video which are adequate to the task of representing the medium. This idea in its essence is identical to the Computation Media Aesthetics paradigm, but coming from the AI background it has more practical influence in CBVIR. In fact, this work precedes CMA acquiring the groundbreaking status in the field. Furthermore, in [DAVIS1] Davis proposed the concept of Media Streams as a visual language for video representation. It utilises a hierarchically structured semantic space of iconic primitives, which are combined to form a set of compound descriptors. Though not directly connected with CBVIR task, these representations opened a huge space indispensable to the development in the field.

Nowadays, following the wave of research activities that attempt to address the problem of semantic gap, various approaches are proposed. In [NAPHA] a graph framework of probabilistic multimedia objects called *multijects* attempts to formulate relationships of the low-level descriptors, semantic labels and contextual information. Multiject representation achieved good results in semantic labelling, but the efficiency of the probabilistic approach appeared to be a drawback of the system as a result of high computational complexity involved. Another statistical approach in [VASCO] suggested a statistical model for content analysis relying on shot duration and activity. They apply Bayesian formulation for shot segmentation and later semantic labelling.

Attempts to formulate a prolific representation of video that should enable semantic CBVIR have had two common characteristics. One is that they all turned to knowledge of the content producers and theoreticians searching for a strategy that would facilitate their aims. The other, less progressive characteristic is that the results achieved didn't allow adaptable and scalable representations due to their high complexity and structural approach adopted. Thus, the results of semantic CBVIR that fundamentally depend on the user's preferences, contextual information and all the relationships between the instances involved couldn't achieve much. Without adaptive and scalable representations computed efficiently, the final target of the semantics in the CBVIR will stay on the other side of the semantic gap.

III.7.1. EXPRESSIVE ELEMENTS USED IN VIDEO REPRESENTATIONS

This section presents expressive elements used by film and TV producers and exploited in the CBVIR area to extract essential perceptual and structural features of video media. In some publications these elements are called mid-level descriptors, explaining their position in the signification chain. Certainly, this is only a limited and the simplest set of expressive elements used, but some of the biggest film and TV theoreticians argue that some of these elements generate basic impressions in minds of the audiences.

III.7.1.1. Shot Pace

Tempo or pace is often used interchangeably in film appreciation, and refers to the ”rate of performance or delivery”. Zettl [ZETTL] makes a distinction in defining pace as the perceived speed and tempo as the perceived duration. Thus tempo/pace is a reflection of both the speed and time of the underlying events being portrayed and affects the overall sense of time of a movie. Tempo is crafted and manipulated in different ways. One technique is the montage that allows a director to manipulate the shot lengths used in the creation of a scene, thus deliberately controlling the speed at which a viewer's attention is directed. Another means by which a viewer's perception of speed can be manipulated is through controlling object and camera motion. Fast motion gives us the feeling of fast events, while no or little motion has the opposite effect on our perception of pace. Film audio is a third factor that increases or decreases our sense of the performance delivery. There may be other more subtle factors besides the story itself, but we argue that one can construct a computable and powerful expressive element, pace, that reasonably captures the flow of time in a movie based on the underlying primitives of shot length and motion.

III.7.1.2. Rhythm

Film rhythm is another complex narrative concept used to endow structure and form to film. Mitry defines it as an "organization of time" [MITRY, MITRY2]. Of the many, often elusive cinematic devices contributing to film rhythm, Bordwell and Thompson [BORDW] state that "frame mobility involves time as well as space, and film makers have realized that our sense of duration and rhythm is affected by the mobile frame". They list camera position/movement, sound rhythm, and editing as constituent elements of rhythm. Further they label resulting rhythms types in higher-level terms by stating that a "camera motion can be fluid, staccato, hesitant and so on". Thus, because a film is structured in time with editing, it manifests a natural beat, and has an intrinsic rhythm. To find this rhythm, one must examine a neighbourhood of shots. In addition, since both shot length and motion contribute to rhythm, one can examine the rhythm that arises individually and jointly from these contributing elements. Since shot length and motion are computable, motion rhythm and editing rhythm are likewise derivable from them.

III.7.1.3. Motion

The most discernible difference between still images and moving pictures stems from movements and variations. In order to obtain a more precise and complete semantic information from video, we need the ability to classify objects appearing in a video sequence based on features such as shape or colour, as well as their movements. Besides providing information on objects trajectories, analysis of motion is useful to detect objects, to recover the kind of camera operation (e.g. zoom, pan, tilt), and to create salient video stills by mosaicking several frames.

III.7.1.4. Camera Motion

Camera operation information is very significant for the analysis and classification of video shots [HIRZAL], since it often explicitly reflects the communication intentions of the film director. The seven basic camera operations are fixed, panning (horizontal rotation), tracking (horizontal transverse movement), tilting (vertical rotation), booming (vertical transverse movement), zooming (varying the focusing distance), dollying (horizontal lateral movement) and Roll (coaxial rotation), as shown in Figure III.24.

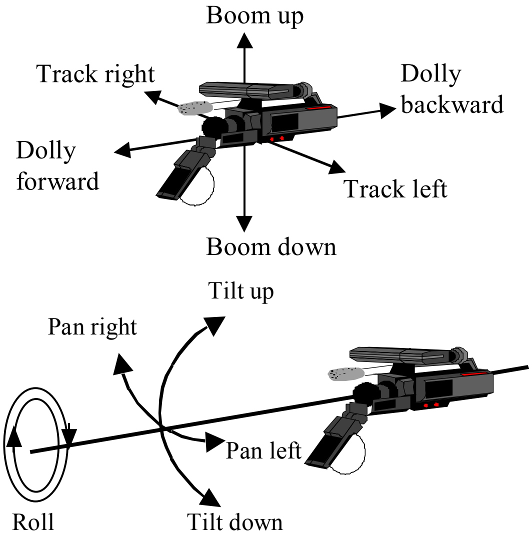


Figure III.24 Camera Motion Types

Camera operations include the basic operations and all the different possible combinations [ZHANG]. Each of these operations induces a specific pattern in the field of motion vectors from a frame to the next. Simple methods for detecting panning (tilting) and zoom operations have been proposed in [ZHANG1]. In order to detect camera operation, the motion vectors can be obtained by optical flow techniques or by coding algorithms such as MPEG or H.263.

The first step aims at discriminating between static/motion scenes; this can be done simply by looking at the average size of the motion vectors. The motion vector field for any combination of panning and tilting will exhibit a single strong modal vector value which corresponds to the direction of camera movement. Most of the motion vectors will be parallel to this vector. This may be checked by analyzing the distribution of the direction of the motion vectors; a pan/tilt is characterized by a small standard deviation of the distribution or by a small absolute deviation from the modal direction as suggested in [ZHANG]. Zooming is characterized by a flat appearance of the direction distribution. Alternatively zooming operations are characterized by vectors of opposite sign at the frame edges. This means that the magnitude of the difference between vertical (or horizontal) components exceed the magnitude of both components. This simple approach can be fooled by the motion of large objects. More generally, the problem of recovering camera motion can be seen as that of estimating an affine transformation which accounts for the dominant global view transformation.

Akutsu et al. [AKUTSU] have used motion vectors and their Hough transforms to identify the seven basic camera operations. The motion vectors pattern is characterized physically and spatially by (i) the magnitude of the motion vectors and (ii) the divergence/convergence point. For ex-ample, in case of a simple zoom in (Figure III.25a) and pan right (Figure III.26a) at a constant speed, the motion vectors are shown in Figure III.25b and Figure III.26b respectively. The algorithm has two stages. The first stage employs block matching to determine the motion vectors between successive frames. In the second stage the motion vectors are transformed to the Hough space. The Hough transform of a line in the spatial domain is just a point in the Hough space. A group of lines in the spatial domain are represented by

$$\rho = x_0 \cdot \cos(\varphi) + y_0 \cdot \sin(\varphi) \tag{III.50}$$

in the Hough space, where (x0 , y0) is the point of divergence or convergence. The least-squares method is used to fit the transformed motion vectors to the curve represented in the formula above. Seven categories of camera operations have been estimated: pan, zoom, tilt, pan and tilt, pan and zoom, tilt, zoom, and pan. We note this technique based on motion vectors is noise sensitive and has a high computational complexity.

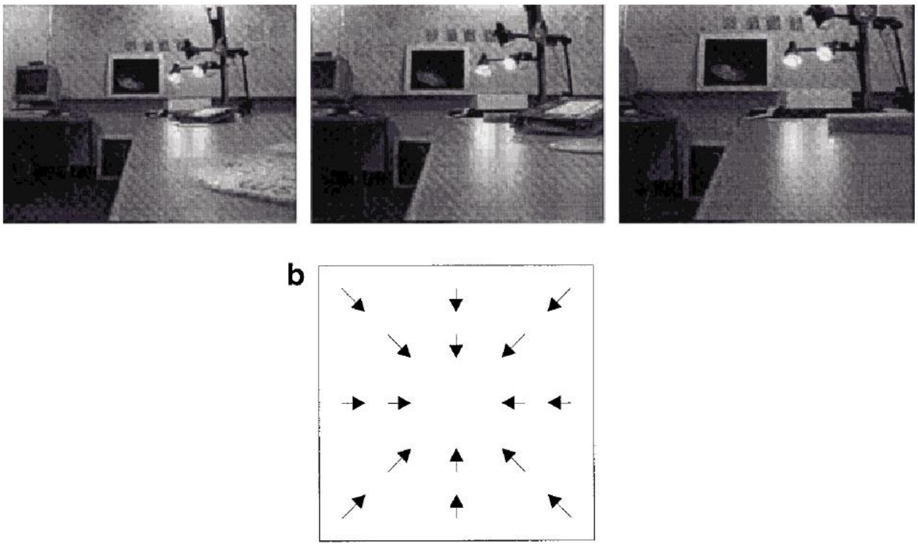


Figure III.25 Camera zoom operation and corresponding motion vectors

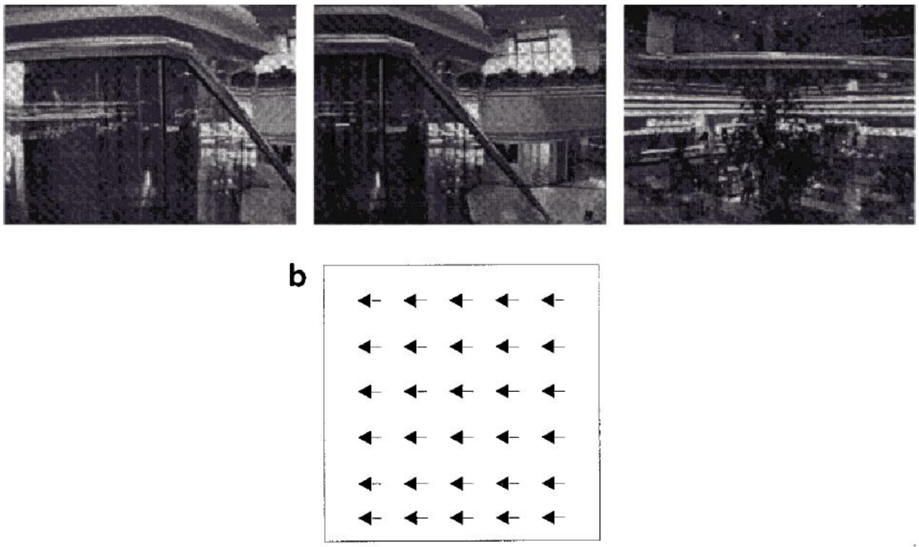


Figure III.26 Pan operation and corresponding motion vectors

An alternate approach in detecting camera operations is to examine what are known as the X-ray images [AKUTSU]. Edge detection is first performed on all the frames within a shot. A horizontal X-ray image is then obtained by taking a weighted integral of the edge frames in the horizontal direction. Similarly, a vertical X-ray image is obtained by taking a weighted integral of the edge frames in the vertical direction. Camera operations are obtained by approximating the spatial distribution of the edge angles of the horizontal and vertical X-ray images. We note that performing edge detection on all frames in the sequence is time consuming.

We note that in all the previous techniques, only a subset of the camera operations is extracted. In addition, it is not possible to distinguish tracking from panning, and booming from tilting. Recently, Srinivasan et al. [SRINI] have proposed a technique based on optical flow in order to distinguish tracking from panning, and booming from tilting. This technique is based on the idea that if the components of the optical flow due to camera rotation and zoom are subtracted from the optical flow, the residual flow will be parallel.

We note that in all these techniques for the detection of camera operations, it is assumed that there is no large moving object dominating the visual field in the video sequences. In case of the presence of a large moving object dominating the visual field, false detection of a camera operation may occur. The effect of a large moving object on the detection process can be reduced by employing techniques to detect the moving objects and compensate for their effects.

III.8. GENRE CLASSIFICATION

In a recent review on multimodal video indexing Snoek and Worring define genre as a set of video documents sharing similar style, putting genre information as the first level of semantic index hierarchy [SNOEK]. This standpoint is supported in the discussion in Section 6.3 of the previous Chapter. Genre acts as the main contextual guideline of video indexing. Thus, there has been a lot of research effort put in the genre classification and verification task.

Foundations of genre classification were laid by Fischer et. al. [FISCHER] where the genre classes were mapped in a three level processing sequence. On the first level, syntactic properties of videos, like shot boundaries, colour descriptors, camera and object motion and audio, are extracted from the sequence. These properties are analysed on a more abstract level trying to define the main attributes of the film style, like camera zooms or pans, speech, music, etc. Finally, the style attributes are mapped to previously defined genre classes.

Following similar concepts, some more detailed analyses of particular style attributes have been published since. Interestingly, the most prolific medium having been analysed is audio. The main reason for that is the extremely high computational complexity of the visual media computation, so the pragmatic researchers turned towards audio classification tools. Jasinschi and Louie present classification of TV program genre based on audio patterns defined as a set of relative probabilities for a set of mid-level audio categories [JANSCH]. Research work by Roach and Mason on video genre classification using audio features applies various algorithms like mel-frequency cepstral coefficients, short term spectral estimates, etc. [ROACH1]. Roach proposes a system that firstly does a discrete Fourier transform (DFT) applied to a short time frame of the time domain signal and the magnitude terms obtained. The second step is to apply a log function to the magnitude spectrum. This serves to reduce the dynamic range of the spectrum. Then a mel filter bank is applied and finally a discrete cosine transform (DCT) is applied to give the cepstral coefficients used in the classification process.

If the contextual information limits the classification environment then the classification task can become more specific as for example in [IDE] were semantic attributes of captions are used for classification of news videos. A combination of static and dynamic features used in a limited environment is presented by Haering et. al. in [HAERING] where event detection is applied to detecting hunts in wildlife videos. Similar approach is applied to sports sequences by Yow et al in [YOWJ who analyse soccer video for highlights, where the ball is tracked and the static up-rights of the goal posts are detected to indicate a shot on goal. These applications require constrained inputs for success; they rely on the video being pre-classified into news, wildlife and sport respectively. It is this high level of video classification to which our approach is applied.

Approaches that use less complex motion measures to classify video sequences are presented by Bouthemy et. al. For example in [FABLE] and [BOUTH] local motion measures and global motion features are used to classify temporal textures such as fire and foliage; they also claim that these measures can be used to retrieve clips of similar global motion properties such as sports.

Work of Troung and Dorai [TROUNG] examines a set of features that would be useful in distinguishing between sports videos, music, news, cartoons, and commercials. In contrast to audio based algorithms they concentrate on features that can be extracted only from the visual content of a video. Rather than learning features from video data sets, they use human perception and discernment of video genre characteristics as a starting point, and extract computational features that would reflect those visual characteristics such as editing, motion, and colour. They address the related issue of the length of a clip required to be processed for reliable genre identification and its impact on the classification performance using proposed features.

Likewise, Rasheed and Shah [RASHE] analyse Film Grammar or Cinematic Principles (camera movements, sound effects, lighting, etc.) by which one can create mood and atmosphere, induce emotional reactions and convey information to the viewers. They first classify movies into action and non-action classes by estimating the visual disturbance and average shot length using a very simple but robust technique. Visual disturbance is defined as the motion content of a video clip. Using the colour and audio information and combining that with the Cinematic Principles to classify movies they make three subclasses: comedy, horror and drama/other under non-action group. Finally they classify action movies into explosion/fire category and other-action category. This is done by analyzing audio information and identifying the peaks in sound energy while testing corresponding video frames for the occurrence of an explosion.

Various other approaches have been applied to the task of genre classification using different modalities of the video media, like textual transcripts, TV schedules, and other available metadata, next to the audio-visual domain analysis. Related work includes Infomedia's Universal Genre Classification System [INFOME]. This system is text-based and it employs country-specific and language-specific program classifications; it describes a TV program into three levels, were each of these levels is broken down in 12 general headings, followed by sub-categories.

Utilising multimodal information for genre classification in a dynamic and adaptive way is a new challenge. Up-to-date research has offered classification into relatively small number of genre categories with too broad meanings. By following the CMA paradigm in a more general way we are offered broad and complex information that is computationally too expensive for the system and unable to scale down to the level bearable for the implementation resources.

IV. EFFICIENT LOW-LEVEL FEATURE EXTRACTION

IV.1. OVERVIEW

This Chapter brings in-detail description of algorithms applied in the low-level feature extraction process. First part gradually introduces methods for temporal segmentation of MPEG videos that exploit temporal prediction information embedded in the stream. Following that, a technique for key-frame extraction based on the presented scalable temporal segmentation is described. In addition to temporal parsing, an algorithm for efficient global and camera motion categorisation is presented. Finally, a hierarchical colour descriptor is generated by applying a scalable quantisation of colour information in the descriptor domain.

IV.2. METRIC EXTRACTION

In order to apply temporal analysis to a video stream one has to extract representative information on the way visual features change in time. The major goal of the temporal analysis is to run in real time, i.e. that the processing period is shorter than the frame rate of the streamed video. Although the processing power is big nowadays, requirements for real-time broadcast quality video processing in spatial domain haven't been met yet. Therefore, the focus of this research are the algorithms for temporal analysis in the compressed domain, particularly applied to the widespread video compression standards like MPEG-2 and H.261.

IV.2.1. PREDICTION INFORMATION IN THE MPEG STREAM

The major contribution to the high MPEG1/2 compression rate lies in the exploitation of the temporal redundancy present in the sequence of frames that form a video stream. By analysing the behaviour of the way the redundancy is being minimised by temporal prediction it is possible to detect global visual changes present in the stream without decompressing it. This Section introduces the initial terms and notations of the applied methodology.

As described in Chapter III, MPEG-2 encoders compress video by spatially dividing each frame into 8x8 blocks and quantising its DCT coefficients. Besides that, a group of 6 to 12 blocks form a so called *MacroBlock* of size 16x16 pixels. In addition to encoded pixel values, MacroBlock unit contains information about the type of temporal prediction and values of the corresponding vectors used for motion compensation. The character of the MacroBlock prediction is defined in a MPEG variable called *MBType*. There are four types of MacroBlock prediction:

- *Intra* coded
- *Forward* referenced
- *Backward* referenced
- *Interpolated.*

Each block of an *Intra* coded MacroBlock is encoded without any temporal prediction, being equivalent to a 8x8 block in the JPEG compression standard. Pixels in the *Forward* referenced MacroBlock are predicted by a region in the preceding reference frame, whether if it's I or P frame. On the contrary, *Backward* referenced MacroBlocks are predicted by the subsequent reference frame. Finally, pixels of the *Interpolated* MacroBlocks are predicted by both preceding and subsequent reference frame with equally weighted fraction of the prediction value.

Temporal prediction is applied to the frame sequence in order to minimise high temporal redundancy present in the stream. To avoid flicker and to produce an impression of the continuous motion visual changes between the displayed frames are small. Therefore, if there is no abrupt visual change present in the frame sequence the preceding frames can predict well the visual content of the subsequent frames. This is the main concept that will be followed in the development of the temporal analysis algorithm.

Because of the present prediction within a shot, a continuously strong inter-frame reference will be present in the stream as long as no significant changes occur in the scene. The "amount" of inter-frame reference in each frame and its temporal changes can be used to define a metric, which measures the probability of a visual change in the given frame. Therefore, the analysis of the MBType information embedded in MPEG stream is an efficient way to measure the "amount" of inter-frame reference. By exploiting the reference information, a frame-to-frame difference metric is to be generated so as to detect visual changes and parse the video in the temporal domain into visually homogeneous units - shots.

IV.2.2. MPEG SEQUENCE STRUCTURE

Although thee are variations in the MPEG sequence structure and its profiles, the majority of MPEG encoders utilise bidirectional prediction. Only the bidirectional prediction brings the high compression ratios and at the same time minimises perceptual distortion of the perceived video quality. Therefore a random MPEG stream is likely to have B type frames present.

With this in mind, it is assumed that in analyzed MPEG stream *Group Of Pictures* (GOP) will have the standard structure [IBBPBBPBBPBBPBB] i.e. there will be two bidirectional frames between two reference frames with encoder parameter M=3. Observe that this frame structure can be split into groups of three having the form of a triplet: IBB or PBB. In the sequel, both types of the reference frames (I or P) are denoted as R_i, front bi-directional frame of the triplet as B_i (uppercase), while the second bi-directional frame is denoted as b_i (lowercase). Thus, the MPEG sequence can be analyzed as a group of frame-triplets in the form

$$R_1\, B_2\, b_3\, R_4\, B_5\, b_6 \ldots R_i\, B_{i+1}\, b_{i+2} \ldots$$

This triplet structure representation simplifies the notation in the future calculus, so it will be user throughout this Chapter. Having defined the main variables and notation, the next Section brings the first definitions of the frame difference metric.

IV.2.3. FRAME DIFFERENCE METRIC WITHIN ONE SGOP

Possible locations of a cut in a frame triplet are depicted in Figure IV.1. Considering the previously defined triplet structure in the MPEG sequence, there are three possible positions of the shot boundary:

- Shot ends with reference frame, and the new one begins with front bi-directional frame B_i
- Shot ends with rear bi-directional frame b_{i-1}, and the new one begins with reference frame R_i
- Shot ends with front bi-directional frame B_i and the new one begins with b_i.

Let us analyse the behaviour of the temporal prediction present in the frame triplet depending upon the shot boundary position. If the front referenced frame B_i is the first frame of the next shot (Figure IV.1a), the next reference frame R_{i+2} predicts a significant percentage of inter-frame MBs in both B_i and b_{i+1}. This is due to the fact that the majority of MBs is visually similar to the reference MBs present in the R_{i+2}

frame. If the scene change occurs at R_i (Figure IV.1b), then the previous bi-directional frames B_{i-2} and b_{i-1} will be mainly referenced to R_{i-3}. Finally, if the scene change occurs at b_i (Figure IV.1c), then B_{i-1} will be referenced to R_{i-2} unlike b_i that will be predicted mainly by R_{i+1} reference frame.

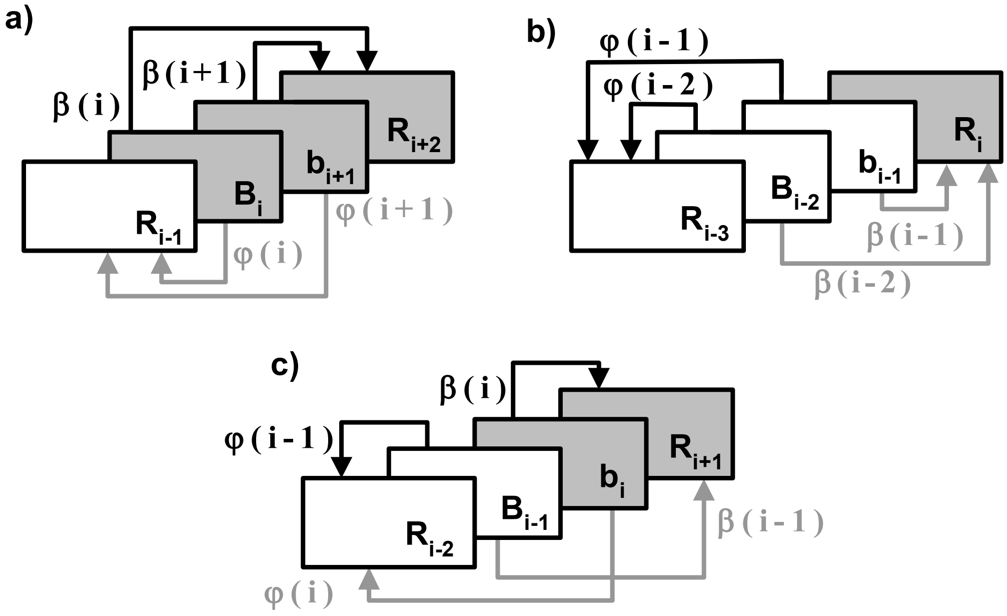


Figure IV.1 Possible positions of the cut in a frame triplet

By analysing which MBType is predominant in the analysed frame, one can gain the information about the "amount" of inter-frame referencing between a reference frame and the predicted frame in a given SGOP. If two frames are strongly referenced and thus visually similar their MBType variable will be predominantly either forward referenced, backward referenced or interpolated. On the other hand, if there is a visual change present between the reference frame and the predicted frame, predominant MBType will be intra-coded. For example, if the bi-directional frame is strongly referenced to its preceding reference frame, then there will be a lot of forward referenced MBs in the frame.

Having this in mind, a frame-to-frame difference metric is generated by analyzing the percentage of MBs having a specific prediction type in a given frame. Let $\Phi_T(i)$ be the set containing all forward referenced MBs and $B_T(i)$ the set containing all backward referenced MBs in a given frame with index i and type T. Then the cardinality of $\Phi_T(i)$ is denoted as $\varphi_T(i)$ and the cardinality of $B_T(i)$ as $\beta_T(i)$. The metric $\Delta(i)$ used to determine the measure of frame-to-frame difference is defined as:

$$\Delta(i) = \begin{cases} \beta_B(i) + \beta_b(i+1), \text{ if } i^{th} \text{ frame is a B frame} \\ \varphi_B(i-2) + \varphi_b(i-1), \text{ if } i^{th} \text{ frame is a R frame} \\ \varphi_B(i-1) + \beta_b(i), \text{ if } i^{th} \text{ frame is a b frame} \end{cases} \quad \text{(IV.1)}$$

Δ(i) is directly proportional to the probability of strong content change event at the frame with index i. It means that the proposed metrics is not only the shot change detection evaluator, but also the estimator of general difference between two adjacent frames in a sequence. This is inherent property of the proposed metrics **Δ**(i), and it could be used in the key-frame extraction algorithm and the shot characterisation.

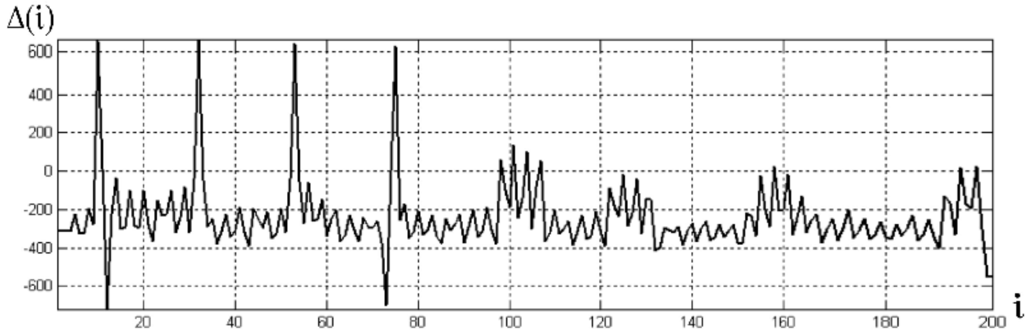


Figure IV.2 Difference metric Δ(i) for the sequence ulosci.mpg (frames 230-430)

IV.2.3.1. Adaptive thresholding

Since **Δ**(i) is a frame-to-frame difference metrics, the peaks in **Δ**(i) present strong and abrupt changes in the visual content as depicted in Figure IV.2. Cut positions are determined by thresholding the metric applying the adaptive threshold algorithm [YUSOFF, DUGAD]. The algorithm is based on the assumption that the probabilistic model of the not-a-shot-boundary event N is unimodal and stationary. With this assumption, the decision threshold m_T is recalculated for each new frame as follows:

1. The mean μ_N and the variance σ_N are estimated dynamically from the difference metric Δ of M neighbouring frames,
2. The value of the adaptive threshold is calculated following the Dugad model as:

$$\delta_T = \mu_N + T_d \cdot \sqrt{\sigma_N} \quad \text{(IV.2)}$$

where T_d is empirically determined in the literature [DUGAD] as $T_d = 5$.

3. Decision is made whether the current frame is a shot boundary or not. After a shot cut is detected, no new decisions are made until M/2 frames have elapsed.

However, due to the existence of gradual changes, like wipes or dissolves, where there is no significant change in the visual content and where the prediction is partially existing even during the transition, a method for detection of gradual changes has to be developed. In order to achieve this by keeping the processing in the compressed domain, particularly on the prediction data, one has to take a more general approach to frame difference metric. In the next section, an attempt to reuse the prediction data for detection of gradual transition is presented. It follows the approach of the previous difference metric extraction algorithm.

IV.2.4. METRIC EXTENSION FOR GRADUAL CHANGE DETECTION

The next step in the implementation of a shot changes detection algorithm is the detection of gradual changes. Gradual transitions do not show such a significant changes in any of the features, and thus are more difficult to detect. Due to advances in digital video editing, there are various types of gradual changes: *dissolves*, where the first shot frames become dimmer, while the second ones become brighter and are superimposed on the first shot frames; *wipes*, where the image of the second shot replaces the first one in a regular pattern, such as vertical line, etc. Since there is inevitably additional processing in feature analysis for gradual changes extraction, real time implementation is more difficult than the basic cut detection. Because of this, the main efforts were directed towards the improvement of the algorithm for gradual changes detection.

IV.2.4.1. Motion Information Based Change Detection

Given that the initial approach was to use information incorporated in the process of motion estimation and temporal prediction, the first feature to be analysed was the set of *Motion Vectors* (MV) from the MPEG stream. The extracted set of vectors is a three-dimensional vector field, and within it there are numerous features that could be analysed for changes detection, such as statistical distribution of vector intensities and angles, gradients, divergences, etc. Unfortunately, experimental results appeared to be poor regardless the choice of the feature utilised to detect the visual changes.

Theoretically, the set of MV should show very typical behaviour during gradual transitions. However there is a decrease in the amount of defined MV per frame in transition regions due to the increase in the number of *Intra* coded MBs. Therefore the result of MV analysis becomes highly unstable. This problem becomes less important in MPEG streams with higher bit rates, but is never avoided completely. Fortunately, there is additional information that can be extracted using MV features like camera movement, panning, zooming, object segmentation, etc.

If one wonders if it is possible to use the significant instability of MV information as a sign of the shot changes, attention should be drawn to the fact that the information whether the MV are defined is actually information stored in MBType variable. Moreover, since the approach to the abrupt shot change detection was based on the analyses of inter-frame referencing, it would be natural to apply the same paradigm to the algorithm for the detection of gradual transitions.

IV.2.4.2. Random Distance Metric for Gradual Change Detection

Obviously, the metrics and the analyses for gradual change detection have to be different from the ones used in cut detection. The conventional process of gradual changes "calculates a frame-to-frame distance and then performs some kind of tracking over it" [BESCOS]. Thus, there is a need for a difference metrics between two frames within a random distance. Again, the amount of inter-frame referencing can be used as an inversion of the difference metrics, but must be generalised to random distance and random frame type for this purpose.

Having in mind the previously defined notation and definitions, let us analyse Figure IV.3, which shows a general frame structure of two frame triplets.

Since it is important to define an inter-frame reference for any frame type at any frame distance, there will be five different types of local inter-frame references d(i) within a frame triplet that will form the overall inter-frame reference $1/\Delta_D(i)$ at random distance D:

- d_{RR} – distance between two R frames
- $d_{RB,}$ d_{BR}– distance between R frame and the closest B frame and vice versa
- $d_{Rb,}$ d_{bR} - distance between R and the closest b frame and vice versa

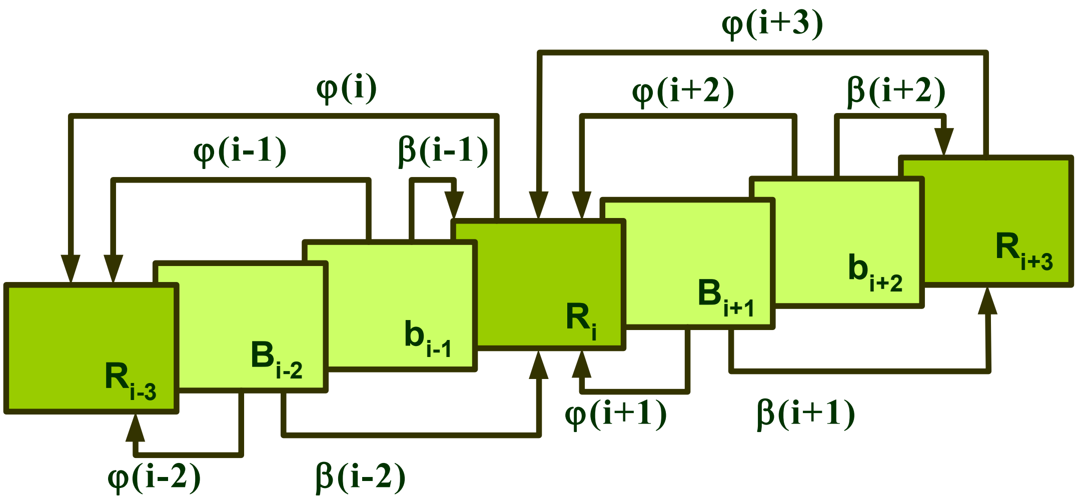


Figure IV.3 Structure of two frame triples

The definition of local inter-frame reference is given as follows:

$$d_{RR}(i) = \max\left(\varphi_B(i+1)\cdot\beta_B(i+1), \varphi_b(i+2)\cdot\beta_b(i+2), \varphi_R(i+3)\right) \tag{IV.3}$$

$$d_{RB}(i) = \max\left(\varphi_B(i+1), d_{RR}(i)\cdot\beta_B(i+1)\right) \tag{IV.4}$$

$$d_{Rb}(i) = \max\left(\varphi_b(i+2), d_{RR}(i)\cdot\beta_b(i+2)\right) \tag{IV.5}$$

$$d_{BR}(i) = \max\left(\beta_B(i), d_{RR}(i-1)\cdot\varphi_B(i)\right) \tag{IV.6}$$

$$d_{bR}(i) = \max\left(\beta_b(i), d_{RR}(i-2)\cdot\varphi_b(i)\right) \tag{IV.7}$$

To calculate the overall inter-frame reference, the local values are multiplied to evaluate cross-referencing, with the frame types and their positions in mind:

⇔ If exists, first element in product is distance from current frame to the nearest R frame,

⇔ Second element is the product of distances between each two R frames from the nearest to the last R frame in the analysed sequence part

⇔ If exists, last element is distance from the last R frame to the last frame of the analysed sequence part

$$\Delta_D(i) = \frac{1}{\overbrace{d_{XR}(i)}^{\exists d}\cdot\prod_{\forall j} d_{RR}(j)\cdot\overbrace{d_{RX}(i+D)}^{\exists d}} \tag{IV.8}$$

Detection of changes is implemented by applying the twin comparison algorithm proposed by Zhang *et al.* [ZHANG2] to the inter-frame difference $\Delta_D(i)$ as the algorithm metrics. The algorithm is based on analysis of the difference measure of two frames at random distance. The distance between the frames should have similar value to the shot transition length. The more similar are these values, the stronger peak will occur at the transition location. For more detailed explanation refer to the previous chapter.

IV.2.5. LOW-PASS FILTERING OF THE METRIC

Since the noise in the extracted metric is very strong, and the curve slopes are weak during the gradual transitions, the metric needs pre-processing in order to detect and locate the shot boundaries' positions. After twin-comparison algorithm the final difference metric is formed. The noise is reduced by low-pass filtering. A convolution with LP Gaussian filter is applied to $\Delta_D(i)$. The pulse response of the filter applied is:

$$h(i) = \frac{1}{\sqrt{2\pi\sigma}} \cdot e^{-\frac{i^2}{2\sigma^2}} \qquad \text{(IV.9)}$$

where i=1,...,N and σ=3 is determined empirically. Filtering is done by convolution of the pulse response and the metrics:

$$\delta(i) = \Delta(i) \otimes h(i) \qquad \text{(IV.10)}$$

After filtering, the difference curve δ(i) is smooth (see Figure IV.4), so that the detection algorithm that locates local maxima in the curve now can be applied.

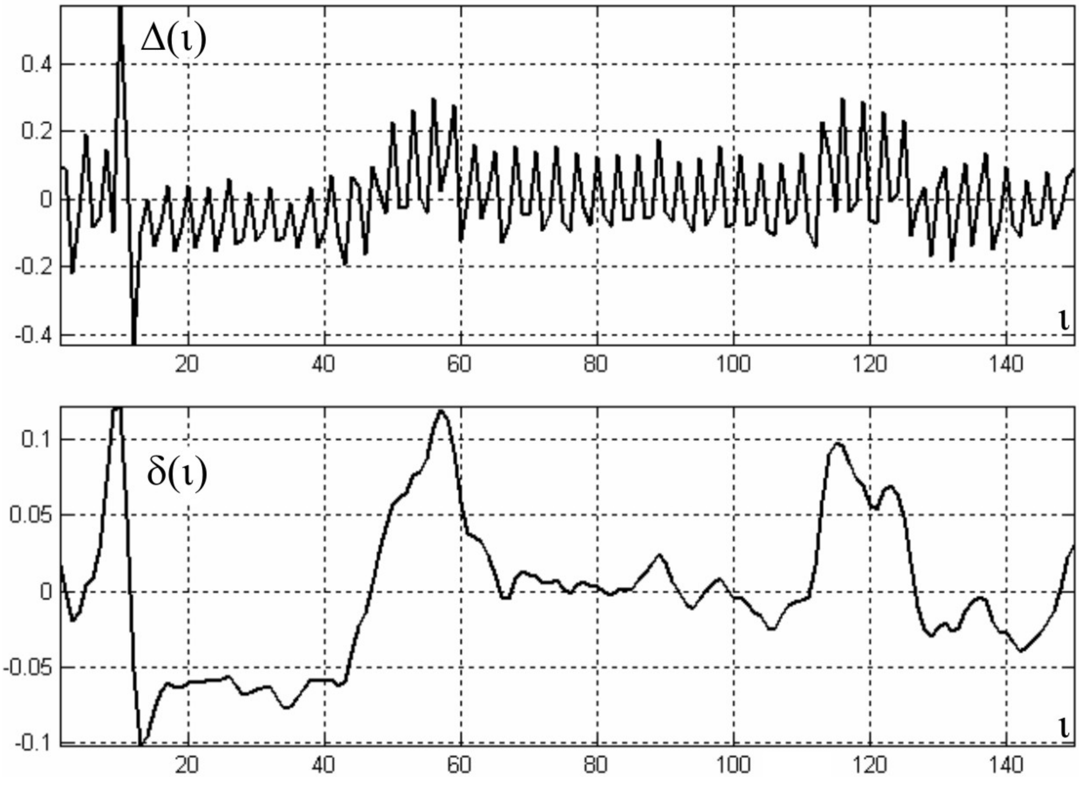


Figure IV.4 Gaussian Smoothing of the Metric Curve of the sequence ulosci.mpg (frames 250-400)

The algorithm is based on the analysis of the first derivation in time of the difference metric function. It detects shot changes at metric's zero-crossings:

$$\left.\frac{\partial\delta(\mathrm{i})}{\partial\mathrm{i}}\right|_{\mathrm{i}=\mathrm{L}}=0\Rightarrow \mathrm{L}\ \text{ is local maxima position} \qquad \text{(IV.11)}$$

Frame indexes of zero-crossings L are locations of peaks that define positions of the shot transitions.

IV.2.6. GENERALIZED FRAME DIFFERENCE METRICS

Although very efficient, the presented difference metric lacks prediction continuity between two SGOP needed for more reliable detection of the shot changes. The results showed poor consistency for instances of the difference metric that compared similarity between the frames in separate SGOPs, and as the distance increases, the reliability of the metric rapidly decreases. Thus, the development process continues in the same direction utilising the prediction information embedded in the MPEG stream, but analysing the differences between the frames with the direct prediction bond, i.e. within the same SGOP. However, the information on the overall change within the SGOP has to be added to the difference metric in order to be able to detect the gradual changes. In addition to that, there is a need for a scalable and hierarchical analysis of the temporal features of video. Based on the approach presented above, following section introduces a novel scheme for scalable video analysis and parses video on a visual event basis, rather than following the standard shot change paradigm. As mentioned before, a high visual similarity within a sequence should result in high percentage of predicted MBs in both bi-directional B frames and predicted P frames and lack of intra coded MBs. More precisely, if two frames are strongly referenced then the most of the MBs in predicted frame would have the corresponding prediction type: forward, backward or interpolated, depending on the type of reference.

Let $\Phi_T(i)$ be the set containing all forward referenced MBs and $B_T(i)$ the set containing all backward referenced MBs in a given frame with index i and type T. In the same manner, sets of intra coded MBs are defined as $I_T(i)$ and interpolated MBs as $\Pi_T(i)$. Then the cardinalities of the corresponding sets are denoted as: $\varphi_T(i)$, $\beta_T(i)$, $\iota_T(i)$ and $\pi_T(i)$. The metric $\Delta(i)$ used to determine a visual difference measure within a frame triplet is defined as:

$$\Delta(\mathrm{i})=\mathrm{k}_{\varphi\mathrm{B}}\varphi_{\mathrm{B}}+\mathrm{k}_{\varphi\mathrm{b}}\varphi_{\mathrm{b}}+\mathrm{k}_{\beta\mathrm{B}}\beta_{\mathrm{B}}+\mathrm{k}_{\beta\mathrm{b}}\beta_{\mathrm{b}}+\mathrm{k}_{\iota\mathrm{B}}\iota_{\mathrm{B}}+\mathrm{k}_{\iota\mathrm{b}}\iota_{\mathrm{b}}+\mathrm{k}_{\pi\mathrm{B}}\pi_{\mathrm{B}}+\mathrm{k}_{\pi\mathrm{b}}\pi_{\mathrm{b}} \qquad \text{(IV.12)}$$

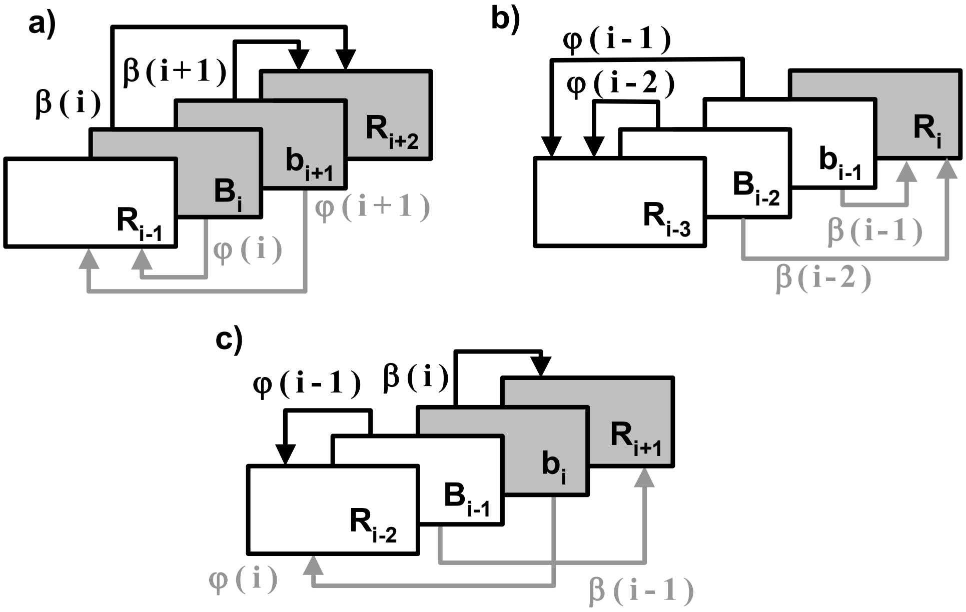


Figure IV.5 Content change in a frame triple

By analysing the prediction character and behaviour in one frame triplet (see Figure IV.5), the changes in visual content within can be estimated. Depending on the frame type, there are three different linear combinations of variables $\varphi_T(i)$, $\beta_T(i)$, $\iota_T(i)$ and $\pi_T(i)$ for both bi-directional frames in a frame triplet. Each linear combination has two main coefficients that are directly proportional to the visual content change within predicted and reference frame in a frame triplet (k=+1), and two that are inversely proportional (k=-1) to it. Additional factors k_π and k_ι are describing overall change in a triplet, one in direct (k_ι) and one in inverse (k_π) proportion. The coefficient values are determined by the rule of thumb, and are presented in Table IV.1. The possible subject of the future research could be the development of optimised and adaptable process of linear coefficient generation in order to improve the difference metric.

	T(i)=R	T(i)=B	T(i)=b
KφB	+1	-1	+1
Kφb	+1	-1	-1
KβB	-1	+1	-1
Kβb	-1	+1	+1
kιB,kιb	+0.5		
kπB,kπb	-0.5		

Table IV.1 Coefficients in the linear combination

IV.2.7. CURVE SMOOTHING

After the metric generation, just as with the random distance metric, the raw difference curve is extremely noisy. Thus, a Gaussian smoothing is applied. However, in this case the noise frequency is known. The source of the noise is the prediction discontinuity between frame triplets. Since there is a reference frame that breaks the prediction bond, often the amount of prediction is rapidly changing from triplet to triplet, though there is no visual change. Therefore, in order to eliminate the discontinuities, a smoothing algorithm is applied.

Since the metrics value is determined separately for each frame and the content change is based on frame triplet element, low-pass filtering with kernel proportional to triplet length would eliminate the noise. The filter with Gaussian pulse response (Figure IV.6) is applied:

$$h(i) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{i^2}{2\sigma^2}} \tag{IV.13}$$

Where i∈[-4σ,4σ], and σ=1.5 . The value for σ is chosen to maximize the smoothing within one frame triplet.

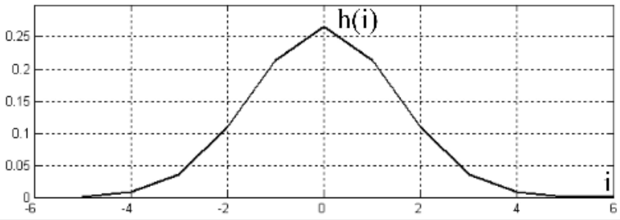


Figure IV.6 Belll shaped pulse response of the applied Gaussian filter

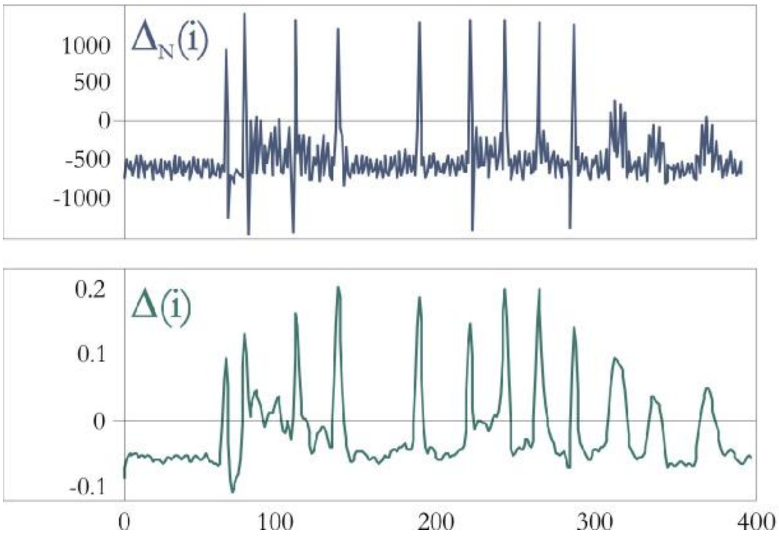


Figure IV.7 Noise suppression for the sequence ulosci.mpg (frames 0-400)

Metrics with suppressed noise is calculated as a convolution of Gaussian filter pulse response and the raw noisy metrics:

$$\Delta = \Delta_N \otimes h \tag{IV.14}$$

Example of noise suppression from a difference metrics is given in Figure IV.7.

IV.3. METRIC SIMPLIFICATION

In order to extract a number of representative frames from the sequence previously filtered difference metrics **Δ**(i) is simplified in a way that spurious and small changes in the metrics curve are discarded without any influence on the main features of the difference metrics. The algorithm that has these features is Discrete Curve Evolution (DCE). Main properties of DCE are [LATEC]:

- It leads to the simplification of curve complexity, in analogy to evolutions guided by diffusion equations, with
- No blurring (i.e. peak rounding) effects and no dislocation of relevant features, due to the fact that the remaining vertices do not change their positions
- The relevance measure K is stable with respect to noisy deformations, since noise elimination takes place in the early stages of the evolution
- It allows us to find digital line segments in noisy metrics due to the relevance order of the repeated process of digital linearization.

Flowchart of DCE algorithm is depicted in Figure IV.8.

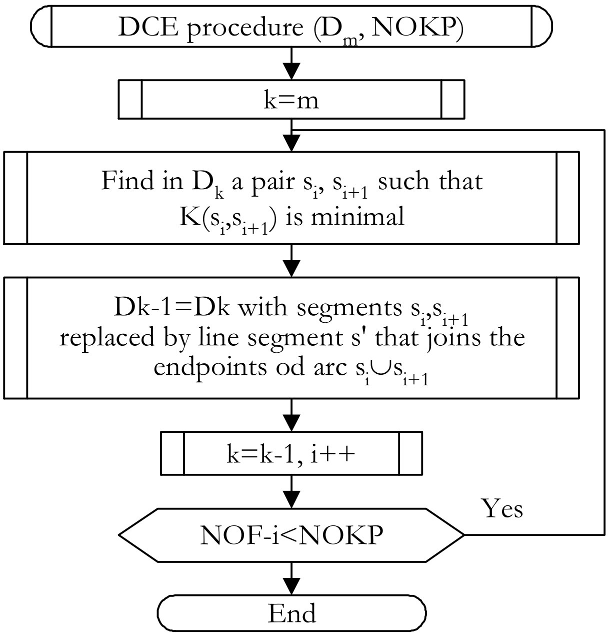

DCE procedure (D_m, NOKP)
k=m
Find in D_k a pair s_i, s_{i+1} such that $K(s_i,s_{i+1})$ is minimal
Dk-1=Dk with segments s_i,s_{i+1} replaced by line segment s' that joins the endpoints od arc $s_i \cup s_{i+1}$
k=k-1, i++
NOF-i<NOKP
Yes
End

Figure IV.8 Flowchart of the DCE algorithm

Let $D_m = s_0, \ldots, s_{m-1}$ be a decomposition of a digital curve S into consecutive digital line segments. The algorithm that computes the decomposition D_k for each stage of the discrete curve evolution k>3 until it reaches wished number of key points (NOKP). Approximate number of key-frames (NOKF) after the DCE algorithm is half of NOKP. The exact NOKF can be determined only *a posteriori*. The input video sequence has NOF frames.

Key-frames positions are determined by locations of the local minima in simplified metrics curve, while shot change central points are located as the local maxima.

IV.3.1. RELEVANCE ORDER

The evolution process is guided by a relevance order. To every pair of two adjacent line segments s_1,s_2 in a decomposition of a given digital curve S is assigned a cost function value $K(s_1,s_2)$ which represents the significance of the contribution of arc $s_1 \cup s_2$ to the shape of S. The pairs of adjacent line segments are ordered with respect to this significance cost. This order is called a *relevance order*. The linearization cost $K(s_1,s_2)$ of any supported arc s_1,s_2 depends on its length, its global curvature and area below the arc. It seems that an adequate measure of the relevance of arc $s_1 \cup s_2$ for the shape of a given object can be based on turn angle $\beta(s_1,s_2)$, on the lengths of the segments $l(s_1)$, $l(s_2)$ and the area of the region enclosed by $s_1 \cup s_2$. It is assumed that the larger both lengths, area enclosed and the total turn of the arc, the greater is its contribution to the shape of a difference curve. Thus, the cost function K is monotonically increasing with respect to the arc lengths, area enclosed and the total curvature. This assumption can be justified by the simple analysis of the Figure IV.9.

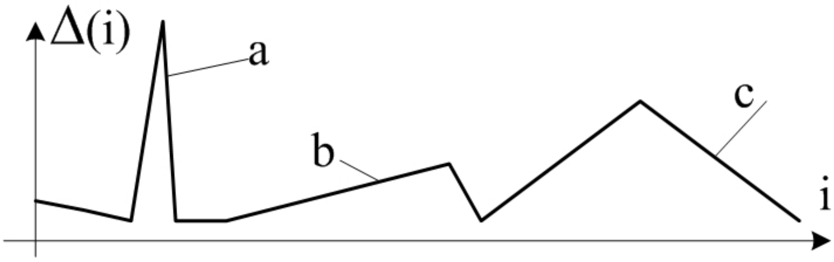


Figure IV.9 Relevance order examples

The peak-like change labelled a) is strong and fast change in visual content. Obviously it is result of a cut change in the video sequence and it has a strong turn angle. Case b) is very long change with long arc segments. Third case c) shows a gradual transition with big area enclosed, but without huge turn angle. These three simple examples depict three criteria for relevance order, and introduce the main ideas for definition of the relevance measure.

IV.3.2. RELEVANCE MEASURE

For each two adjacent line segments s_1, s_2 in the decomposition of a digital curve S, the relevance measure $K(s_1, s_2)$ is determined, which represents the significance of the contribution of arc $s_1 \cup s_2$ to the shape of S. The value $K(s_1, s_2)$ can be interpreted as the cost required for linearization of arc $s_1 \cup s_2$. Let $s_1=AB$ and $s_2=BC$ be two consecutive line segments in the decomposition of curve S, so that $\beta=\alpha_1+\alpha_2$ is the turn angle. The corresponding cost function $K(s_1, s_2)$ is given by the equation:

$$K(s_1,s_2)=\left|\beta(s_1,s_2)\cdot(l_1+l_2)\cdot P_{\Delta(ABC)}\right| \qquad \text{(IV.15)}$$

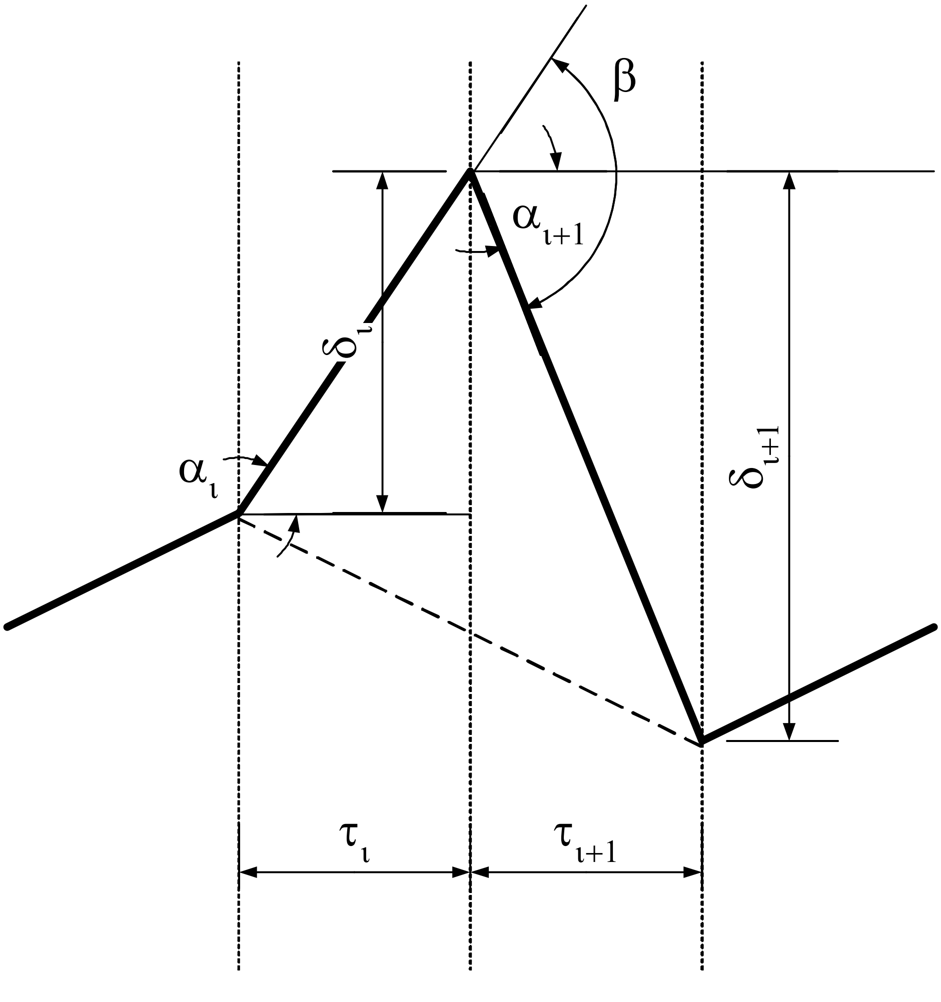


Figure IV.10 DCE linearization of two adjacent line segments

Observing an arc linearization example given in Figure IV.10, formulae for each element in equation above for relevance measure are given as:

$$\delta_i = \Delta(i+1) - \Delta(i)\ ,\ l_i = \sqrt{\tau_i^2 + \delta_i^2} \qquad \text{(IV.16)}$$

$$\beta(s_i, s_{i+1}) = \mathrm{acrtg}\left(\delta_i/\tau_i\right) - \mathrm{acrtg}\left(\delta_{i+1}/\tau_{i+1}\right) \qquad \text{(IV.17)}$$

$$P_{\Delta ABC} = \frac{1}{2}\left(\delta_i\tau_i + \delta_{i+1}\tau_{i+1}\right) \qquad \text{(IV.18)}$$

IV.4. MOTION FLOW EXTRACTION

IV.4.1. MOTION VECTORS AS OPTICAL FLOW APPROXIMATION

As presented in the overview of the MPEG standards, motion vectors embedded in the compressed stream present the translation of the reference MacroBlock in the prediction process in order to minimise the prediction error. As a result of the fact that the only criterion for the choice of the motion vector values is minimal prediction error encoded, one can easily conclude that the set of motion vectors is nowhere near the approximation for optical flow of the video sequence. The best example of motion vectors giving completely wrong picture of motion flow in the video is black break sequence, where the motion estimator in MPEG encoder is constantly trying to minimise the prediction error and searches through the neighbourhood and gives the set of vectors that show almost random, chaotic and strong motion all over the completely black frame! However, information needed for camera motion analysis is quite robust in statistical terms so that the flow extracted from motion vectors could serve that purpose.

IV.4.2. FLOW ESTIMATION

Having the motion vector values extracted from the MPEG video stream as a set of two-dimensional pairs (forward and backward prediction) of motion vectors for each frame, first step in the process of the optical flow approximation would be to generate optical flow using extrapolation of the existing motion vectors.

Extrapolation procedure is depicted in Figure IV.11, Figure IV.12 and Figure IV.13. Union set of all existing motion vectors is labelled as M. For instance, to generate the flow vector for a MacroBlock in the reference frame (I or P) with the index i, one needs to check if there is a corresponding forward motion vector in the frame i+1, and if it exist, the flow vector will be just the inverted forward motion vector of that

MacroBlock. If there is no forward motion prediction in that MacroBlock, the corresponding forward motion vector in the frame with the index i+2 is found, etc.

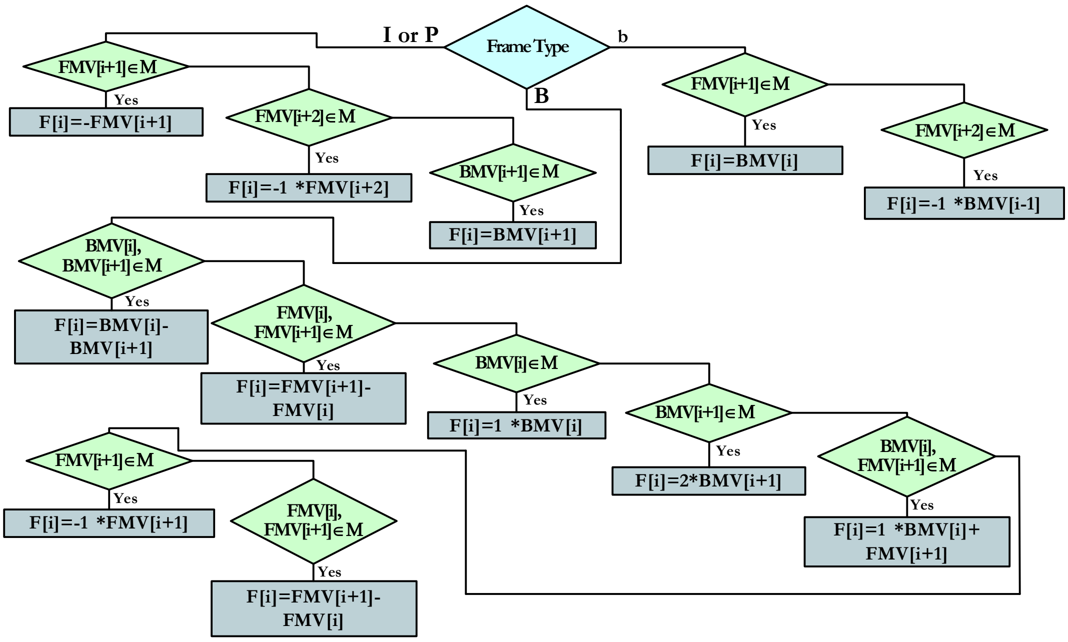


Figure IV.11 Flowchart of the flow vectors extrapolation procedure

By following the algorithm, a majority of the flow vectors are generated. This procedure is based on the motion prediction links depicted in Figure IV.12 and Figure IV.13.

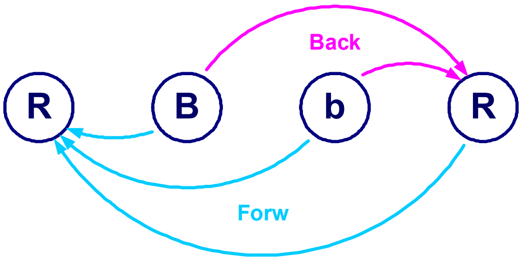


Figure IV.12 Flow Extraction: Analysis within one SGOP

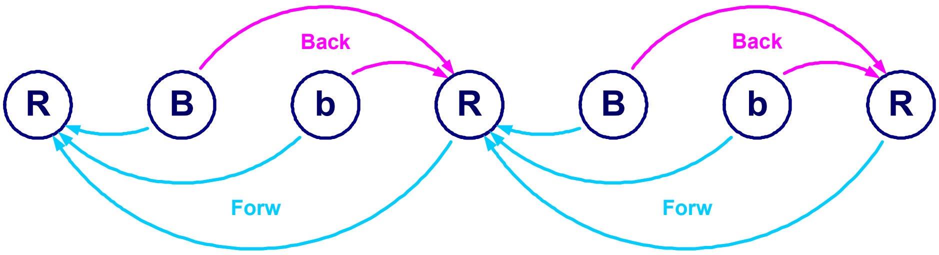


Figure IV.13 Flow Extraction: Analysis of two SGOP

Although the majority of flow vectors are defined after this procedure, it is important to fill in the gaps and output homogeneous optical flow field. Hence, the missing flow vectors are generated by either median filtering of the existing flow vectors or averaging. Both procedures need to repeat iterations until the flow field is smooth and completely defined. However, the whole flow extraction process is fast and it doesn't obstruct the real-time capability of the whole system.

IV.4.3. CAMERA MOTION ANALYSIS

Once the optical flow approximation is generated, categorisation of the camera work is straightforward. Four basic descriptions of camera motion are defined, and the others are just the combination of these four:

1. Horizontal motion
2. Vertical Motion
3. Zoom
4. Rotation

To detect either Horizontal or Vertical motion, the overall sum of the motion field is calculated:

$$\overrightarrow{\mathrm{Sum}} = \sum\nolimits_{i,j} \vec{F}(i,j) \tag{IV.19}$$

If the resultant vector of the motion filed has intensity greater than the predefined threshold, than the motion is directed towards the resultant vector, horizontal, vertical, or diagonal (rare).

In order to detect zooms, he resultant vector is compared to the predefined lower threshold, and if the threshold is not reached, the camera work is either zoom or rotation, or there is no camera motion.

If so, the rotation is detected if the RotSum value is greater than the predefined threshold, where:

$$\mathrm{RotSum} = \sum_{i=0,j}^{I/2} F_h(i,j) - \sum_{i=I/2,j}^{I} F_h(i,j) - \sum_{i,j=0}^{J/2} F_v(i,j) + \sum_{i,j=J/2}^{J} F_v(i,j) \tag{IV.20}$$

Zoom class is assigned if the intensity of ZoomSum is greater than a predefined threshold:

$$\mathrm{ZoomSum} = \sum_{i=0,j}^{I/2} \overrightarrow{F_v}(i,j) - \sum_{i=I/2,j}^{I} \overrightarrow{F_v}(i,j) + \sum_{i,j=0}^{J/2} \overrightarrow{F_h}(i,j) - \sum_{i,j=J/2}^{J} \overrightarrow{F_h}(i,j) \tag{IV.21}$$

If ZoomSum is negative, camera work is classified as zoom out, while if it is positive it is zoom in. The algorithm flow of the classification process is depicted in Figure IV.14.

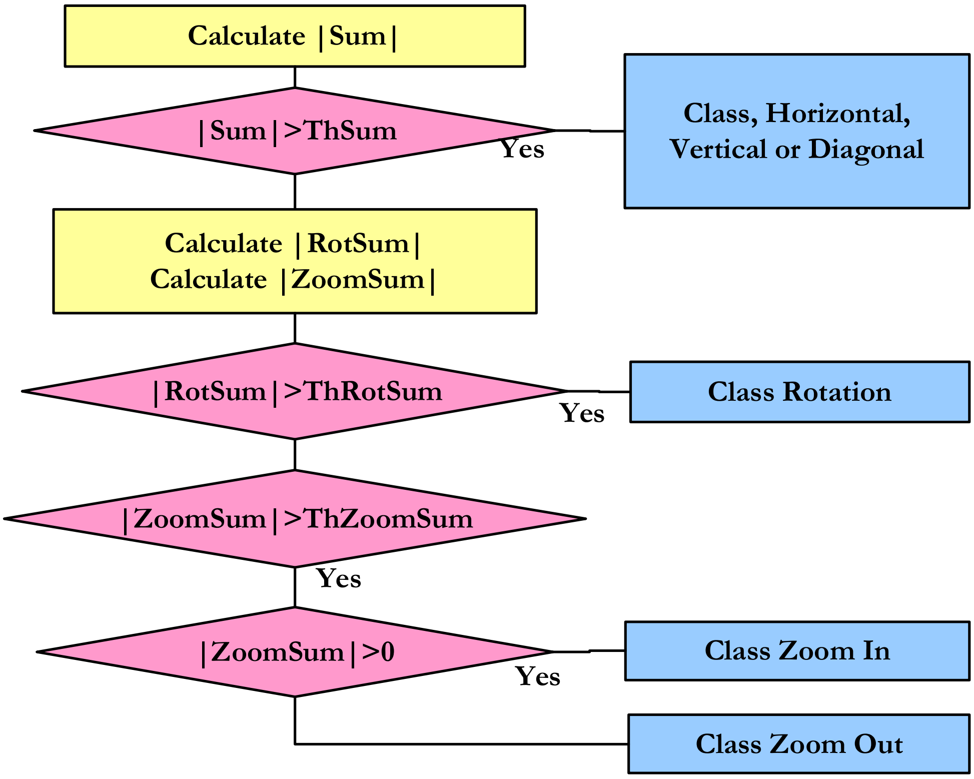


Figure IV.14 Flowchart of the Camera Classification Algorithm

IV.5. KEY-FRAME EXTRACTION

Since the difference metric depicts the overall visual activity within a shot, the best representative frame in visual terms could be detected by analysing the metric. Obviously, peaks of the metric curve are representing a strong visual change, while the valleys represent the visually homogeneous parts.

After metric simplification using discrete contour evolution algorithm metric curve has finite number of linear segments. It enables simple detection of peaks and valleys by applying the second derivation to the simplified difference metric:

$$\Delta_{detect} = \frac{\partial^2 \Delta_{DCE}(n)}{\partial n^2} \qquad \text{(IV.22)}$$

Peaks in the metric are detected as distinctive positive values, while the lowest values are detected as the distinctive negative values as seen in Figure IV.15. The applied threshold for peak detection is a variable more in an adjustment process for the sensitivity on visual changes. Higher values mean that the change was short and strong, meaning it was a cut.

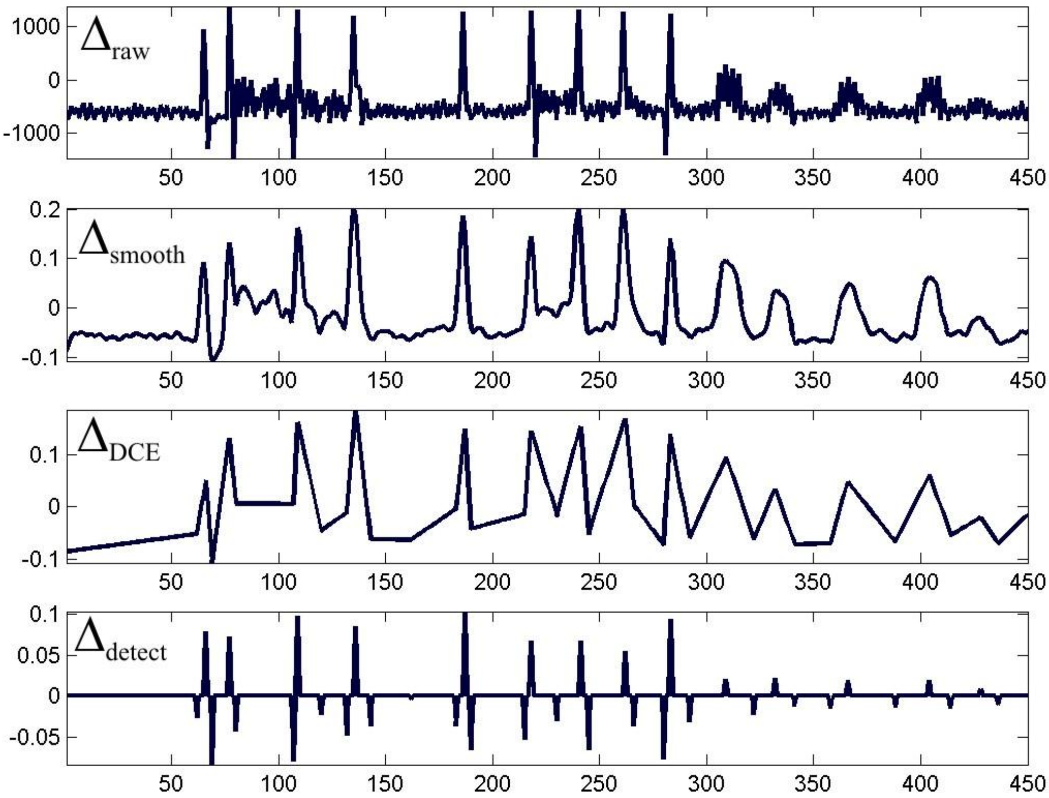


Figure IV.15 Detection Process using DCE of the sequence ulosci.mpg (frames 10-450)

What are the detected peaks and valleys of the metric representing? Evidently, peak positions define the strongest visual changes within the analysed sequence. That would be a detected visual event, as defined previously. On the other hand, valleys represent the lowest visual activity, and in this case, they are used as a detection area for the key-frame. Furthermore, the lowest value in the valley denotes that the visual changes in that part of the shot are smallest so that the particular frame with the lowest metric value is the closest in the visual terms to the neighbouring frames in the shot. With considerable approximation because of the algorithm speed and simplicity, negative peaks are the positions of the extracted key-frames. The customisable BMP or JPG format frames are extracted and saved in a particular directory, as the starting point for the future metadata extraction. As the main feature in the metadata generation process, the HSV scalable hierarchical colour descriptor is applied.

IV.6. COLOUR HISTOGRAM QUANTISATION

Since the computational analysis of the image colour features is the most developed part of the computer vision, every retrieval engine in the world offers colour-based analysis. In order to evaluate the implemented algorithm and exploit this reach and descriptive feature, the hierarchical colour indexing of the extracted set of key-frames is developed.

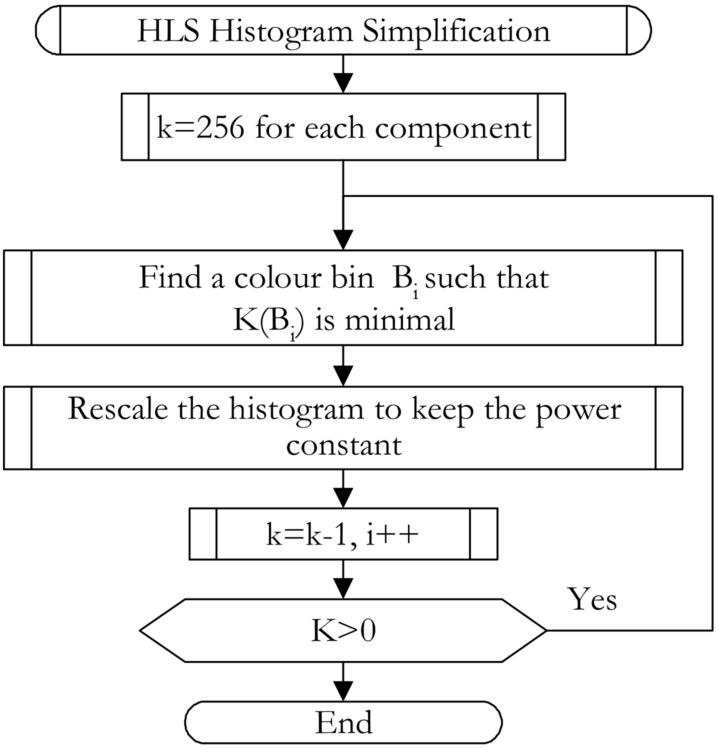

HLS Histogram Simplification
k=256 for each component
Find a colour bin B_i such that $K(B_i)$ is minimal
Rescale the histogram to keep the power constant
k=k-1, i++
K>0
Yes
End

Figure IV.16 Flowchart of the histogram simplification algorithm

Initial task is to choose the optimal colour space. Among different colour representations the HLS model has two important characteristics: it is easy to use and it produces colour components that closely follow those perceived by humans [SWAIN]. For this reason a family of quantised colour histograms in the HLS colour space is used as the set of the image descriptors.

To generate a hierarchical family of histograms, a continuous histogram simplification algorithm is implemented. In each step of the algorithm the least significant colour component is removed and the image degradation measure is calculated. By reaching the desired measure of image degradation the representing histogram is extracted as the image descriptor at that particular level of detail.

The simplification algorithm is similar to the DCE algorithm applied to the frame difference metrics and its flowchart is given in Figure IV.16. It removes colour components gradually using specific relevance measure. The relevance measure function in this case is defined as:

$$K(i) = \text{hist}(i) \cdot \log(\tau_i + \tau_{i+1}) \quad \text{(IV.23)}$$

where *hist(i)* is the image histogram and τ_i is interval between components *i* and *i-1*. The algorithm removes the colour component with the lowest relevance measure value.

The image degradation function Df(n) at the algorithm step n is defined as the cumulative sum of the previously removed histogram bin values:

$$Df(n) = \sum_{\forall m, hist(m) \in hist'} hist(m) \qquad (IV.24)$$

where hist' is a set of previously removed components. The value of Df(n) is equal to the number of image pixels that got removed during the histogram simplification process. The user can predefine the levels of the image degradation according to the addressed application.

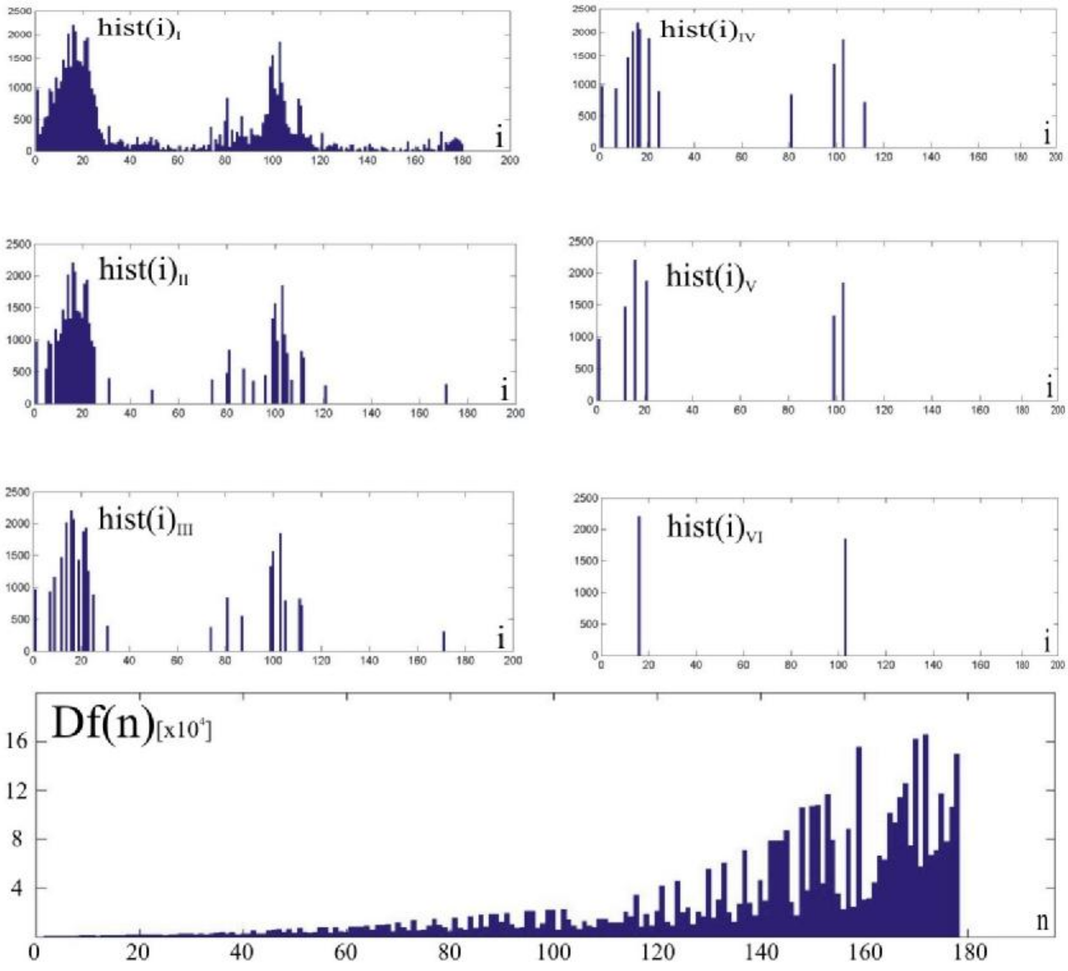


Figure IV.17 Colour Histogram Simplification Process

Figure IV.17 shows six simplification stages of the HLS colour histogram hist(i)$_{I-VI}$.as well as the image degradation function Df(n) thoughout the simplification process. First graph hist(i)$_I$ shows the Hue histogram with 180 colour bins.

Bin removal process is done with the respect to the cylindrical nature of the HLS colour space. The calculation of the cost function in the simplification algorithm starts with the highest value in the Hue histogram, and ends with it. For that reason, the distance of the end points in the Hue histogram is kept constant during histogram quantisation.

To measure colour similarity between key-frames at a given scale, the Hausdorf metric is applied. Each histogram is represented by a set of points $A = \{p_1, p_2, \ldots, p_k\}$ for $k \in [0, 360)$. The distance from any $p \in A$ to another set $B=\{q_1, q_2, \ldots, q_l\}$ is defined as:

$$d(p,\ B) = \underset{q \in B}{\min} \|p - q\| \tag{IV.25}$$

The directed Hausdorf distance from A to B is given by:

$$\mathrm{hdist}(A,\ B) = \sum_{p \in A} d(p, B) \tag{IV.26}$$

Using that, the final distance between A and B is:

$$D(A,\ B) = \mathrm{hdist}(A,\ B) + \mathrm{hdist}(B,\ A) \tag{IV.27}$$

Distance metric D(A,B) is used to determine the visual similarity of two processed key-frame images. It gives a comparison in terms of colour, but it lacks information about texture, shape, localisation and spatial distribution. However, it is used in this work as a simplest low-level perceptual media description for further video representation. In particular, distance metric D(A,B) is used in the experiment described in Section VII.3.2 as a global-colour change representation in the fuzzy learning unit.

V. VIDEO REPRESENTATION MODEL

V.1. OVERVIEW

This Chapter presents a proposed video representation model and a genre classification system that enable automatic annotation of videos. In the opening section a critique of the current representation models is given, as well as the basics of the proposed approach. Section 3 describes the representation model in more detail. Section 4 describes the genre classification algorithm, while Section 5 gives a brief summary of the chapter.

V.2. INTRODUCTION

As described in Chapter 2, the current CBVIR paradigm represents the records in the digital media database, i.e. videos, clips or single images, as points in a metric space. This space is generated in a way that dissimilar videos are distant from each other while similar videos are located close to each other. The distance function that defines the metric space has to follow the user's concept of the visual similarity [CASTELLI]. Unfortunately, current video representation models are nothing more than a subset of well known low-level features extracted from the video, e.g. colour, texture, shape, motion, etc. The major requirement of the distance definition to capture the visual similarity relevant to the user has been never fulfilled.

How can we expect a semantic user centred retrieval if we describe videos with the colour histograms or a wavelet based texture descriptor? What do these features mean to the end-user? These questions have been brought up only recently, with the appearance of the "semantic gap" and the gradual abandonment of the "query by example" paradigm in CBVIR (see Chapter II). Therefore a meaningful video representation model that delivers information relevant to the user in a given context has to be developed. In the following sections a novel video representation model that tackles this problem is presented. It follows computational media aesthetics paradigm and represents videos as a dynamic form, giving information on its dynamic features, namely shot pace and visual activity. Later, a genre classification algorithm is given. This classification algorithm, relying on the aforementioned representation model, embarks upon the context issues, essential for a future automatic video annotation system.

V.3. VIDEO REPRESENTATION MODEL

The transformation of low-level feature vectors into the semantic concepts that are natural to the user is a critical task of a current CBVIR system. This task can be seen as a metric space transformation from the points in the descriptor space to the discrete conceptual semantic space. Unfortunately often this artificial transform is very complex in its nature, being discontinuous and non-linear. Furthermore, the kernel of this "transformation" has to be modelled on the basis of the human visual perception. How do we link a set of luminescent stimuli with the meaningful concepts in our mind is still a mystery even for psychologists. There are numerous models of the human semantic cognition [RUMEL, MCCLE, HARDT], but majority of them appear to have questionable reliability and haven't achieved much in the visual perception domain.

Nevertheless, having the common digital media as the domain of expertise, the contextual limitations could draw the finite boundaries of this transformation, given a set of axiomatic rules present in the digital media production domain. These rules could be seen as the grammar of the visual media language. Moreover, in the film and TV studies a buzzword "grammar" has been the key theoretic tool for objective analysis and synthesis of the media. Its foundations lie in the semiotic theory grounds, as presented in Chapter III. Editing dynamics, camera work, narrative structure in space and time, etc are all film grammar rules appropriate for computational analysis and at the same time important in the creation of high-level semantic concepts.

For example, numerical values of the editing pace will fall into different classes for different genres like commercials and documentaries. While a given shot length could be characterised as "short shot" in a documentary, the same shot length in a commercial could be labelled as "long shot". Furthermore, while flashy shots lasting a fraction of a second can be important "special effects" in a commercial, they can be only the result of editing artefacts in a documentary. Thus, a good knowledge about characteristics of the targeted end-users, environment and application is essential for the design of highly effective knowledge representation.

V.3.1. REPRESENTATION OBJECTIVES

Let us set up the objectives and requirements of the video representation model. As mentioned before, the choice of the low-level features and their extraction process are essential to achieve annotation efficiency. Certainly, a computer cannot extract meaning from low-level features without any additional inference strategy or learning process. Thus, by choosing the most appropriate feature subset and its form to represent a video, the process of giving meaning to the lowest instances of video description becomes more feasible. The main objectives of the model are its efficiency, classification separability, scalability and ability to utilise and adapt to user relevance feedback information.

V.3.1.1. Efficiency

Having in mind the processing resources and costs needed for analysis of digital video media, the computational requirements appear to be an important issue in the design of the representation model. Trade-off between the algorithm's computational efficiency and ability to distinguish between videos conveying different higher-level concepts were the main concern during the development process. Since the backbone of the present feature extraction system is a highly efficient temporal analysis of MPEG compressed video, the major part of the representation model consists of the temporal features extracted directly from the MPEG compressed domain. Therefore, the processing cost of the feature extraction module is minimised.

V.3.1.2. Separability for Genre related issues

In addition to the generation of the links between high-level semantic concepts and low-level video features the focus of the representation model development is the enhancement of the demarcation between videos conveying different concepts in a given context. This requirement is essential in the classification process. This is due to the fact that the genre classification module relies entirely on the separation characteristics of the video representation model. Therefore, the model should transform the feature vectors into their representation that will form groups in the conceptual space around the same concepts, but at the same time make groups as distant as possible. The representation model and the distance metric have to be defined with great precaution. If successful, this module will enable contextual classification of the videos into genres.

V.3.1.3. Scalability & Adaptability

As highlighted in Section II.6.2. one of the major challenges for the future developments of the CBVIR systems is its capability to self-adapt to the contextual circumstances of the retrieval process, and thus limit the signification space in order to gain more precise retrieval hits. Furthermore, it is not only the revaluation of the difference measure in the metric space that should show the adaptive behaviour, but the representation model as well. If the feature extractor module of the system generates a set of low-level visual features, the representation model should adapt itself from the user relevance feedback output. However, to achieve this goal, video representation model has to have capability to change its structure and scale its representations to the desired level. This capability is very important, though in this stage of the research, the system that supports the self-adaptive behaviour has not been developed. However, this functionality of the model, together with all others, will be presented in the following section.

V.3.2. PROPOSED REPRESENTATION MODEL

The set of the features selected to be a part of the representation model has been backed by the efficient algorithm for the compressed domain video temporal analysis. A set of video sequence descriptors is generated using temporal expressive elements, i.e. editing pace and the overall visual activity within a shot. The numerical values that concisely describe these elements are shot length distribution and shot activity.

V.3.2.1. Shot Length Distribution

Following the tendencies of the computational media aesthetics approach, temporal features of video and its structure are considered as the foremost expressive elements to be analysed. The pace and rhythm of the video sequence appear to convey information vital for higher-level concept creation to the user. Therefore these expressive elements have been chosen to be a part of the representation model.

Having in mind that during the initial process of temporal parsing a positions of the shot boundaries are determined, generating the shot length distribution (SLD) of a given video clip would be the most economical representation resourcewise.

As described in the previous chapter, the shot detection peak curve is derived from the frame difference metric as:

$$\Delta_{\text{detection}} = \frac{\partial^2 \Delta_{\text{DCE}}(\text{i})}{\partial \text{i}^2} \tag{V.1}$$

The set of shot boundary locations Λ is determined by thresholding the detection curve with the constant threshold Ψ:

$$\Lambda(i)=\left\{i \mid \Delta_{\text{detection}}(i)\geq\Psi,\Psi=E_{\Delta}+2*\sigma_{\Delta}\right\} \tag{V.2}$$

where E_Δ and σ_Δ are the mean and the standard deviation of the peak metric $\Delta_{\text{detection}}$:

$$E_{\Delta}=\frac{1}{M}\sum_{j=1}^{M}\Delta_{\text{detection}}(j),\quad \sigma_{\Delta}=\frac{1}{M}\sum_{j=1}^{M}\left(\Delta_{\text{detection}}(j)-E_{\Delta}\right)^{2} \tag{V.3}$$

The values of the shot durations are calculated as:

$$\lambda(i)=\left\{\Lambda_{k+1}-\Lambda_{k},\forall k\in\Lambda\right\} \tag{V.4}$$

The process of shot duration extraction is depicted in Figure V.1.

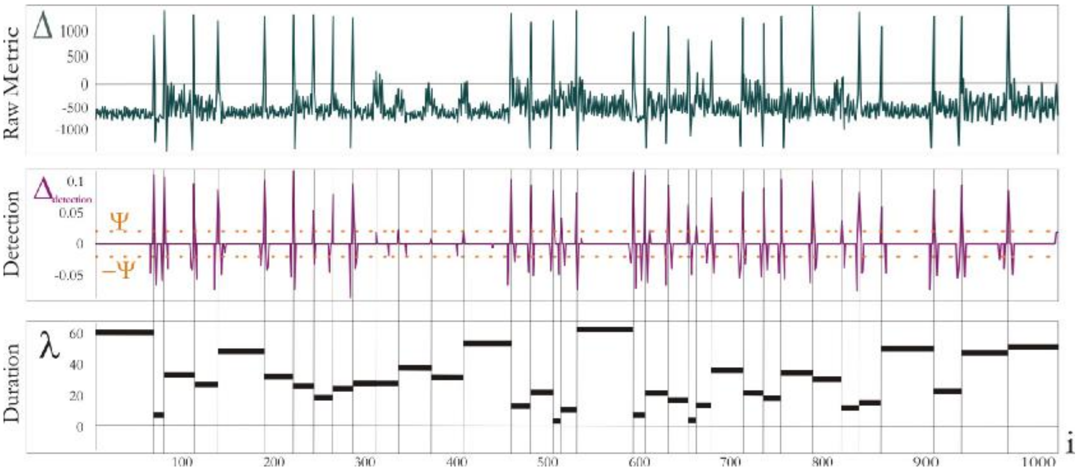


Figure V.1 Shot duration extraction for sample MPEG file ulosci.mpg

The next step in the creation of the shot length distribution model is histogram generation from the set of shot duration λ. As mentioned in the requirements section, distribution representation has to achieve scalability and separability of the classification involved in the process. Therefore, there are three basic ways to determine the histogram bin boundaries: linear, production based and normalised division.

V.3.2.1.1 Linear Division

The duration classes are divided linearly, each having the same range of values, as depicted in Figure V.2. Bin boundaries β are defined as:

$$\beta(i)=C\cdot i, C=\text{const.}=10 \tag{V.5}$$

This is the easiest way to generate shot length distribution and the only one that has been used before. It is typical for the ignorant approach to the user adaptability and leads to the inefficient clustering and classification. It involves all ranges of the shots lengths with the same importance in the process of further classification and conceptualisation. Therefore, the results achieved by applying this division haven't been successful.

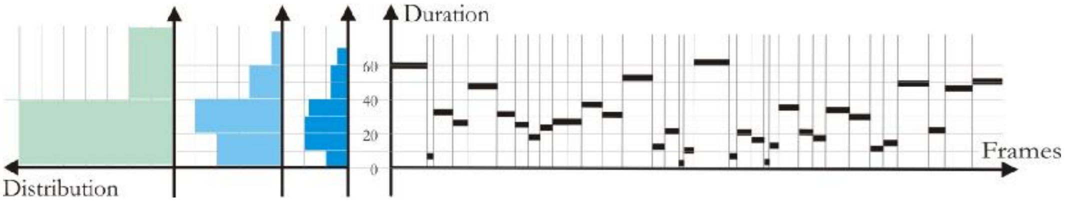


Figure V.2 Scalable shot length distribution generation applying uniform division

V.3.2.1.2 Exponential Division

This model is a rough approximation of the shot length impact on the clustering efficiency. It models it in a way that as the shots get longer, their number is decreasing so that the range of values for longer shots increases. An exponential function is used to calculate bin boundaries β:

$$\beta(i) = 2^{i}, i \in \{2,3,...\} \qquad (V.6)$$

This division is depicted in Figure V.3. Due to the fact that the granularity of the bins for short cuts is too detailed, this approach was abandoned.

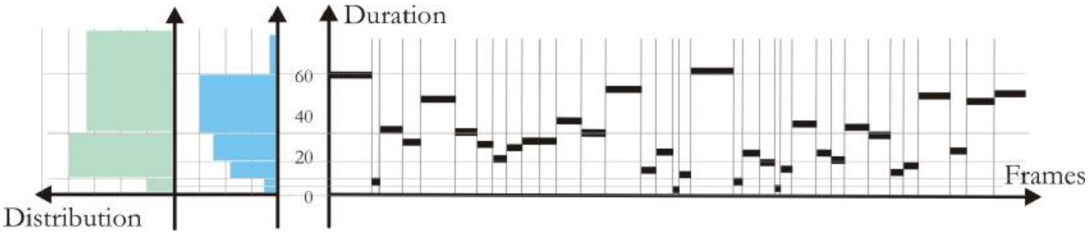


Figure V.3 Exponential division

V.3.2.1.3 Production Based Division

This is a non-linear division where the duration value ranges are determined by following the rules an editor follows during the editing process. Unlike previous methods, even the simplification of the representation is driven by the empiric rules, as depicted in Figure V.4. The bins are grouped by the empirical rules, rather than just grouped in pairs.

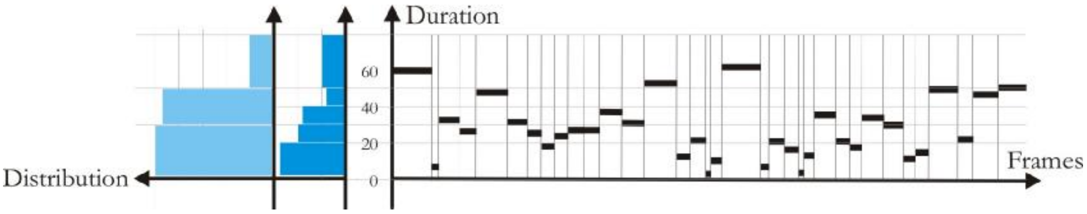


Figure V.4 Distribution generation based on the production rules: non-linear division

V.3.2.1.4 Normalised Division

As the previous division, this is a non-linear process. The bin boundaries are determined by achieving the uniform distribution of the whole dataset present in the classification learning stage, or in the whole database. Thus, the union set of all shots in the database should have the uniform distribution after applying this division. The bin boundaries are determined by following the next algorithm:

1. Form the union set of the all *N* shots in the domain (database, learning set, etc.)

$$\Omega = \bigcup_{\forall m} \lambda_m, \#\Omega = N \qquad (V.7)$$

2. Sort the union set Ω in ascending order to form $\underline{\Omega}$.
3. Extract desired number K of bin boundaries β as:

$$\beta(k) = \underline{\Omega}\left(k \cdot \frac{N}{K}\right), \forall k \le K \qquad (V.8)$$

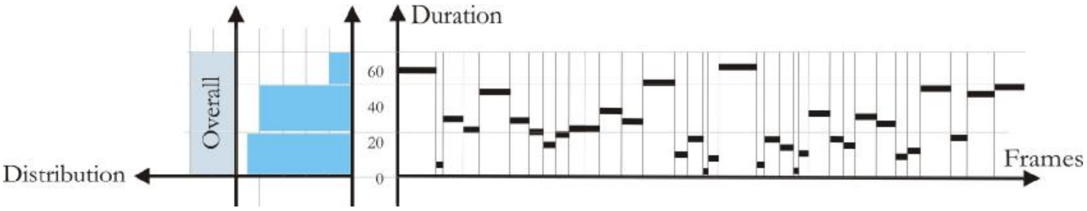


Figure V.5 Normalised division: overall database SLD histogram is uniform

V.3.2.1.5 Applied SLD Representation Models

Two shot length distribution representations are used in the development process. The first one is involved in the genre classification process. Because it is the only classification feature used in this process, a higher precision is needed. Therefore the shot length histogram consists of 6 bins, as shown in Table V.1. The bin boundaries are determined by following the production rules appropriate for the types of programme involved, i.e. news, commercials, soaps, etc.

FLASH	VERY SHORT	SHORT	MID SHORT	MID	LONG
0-10	10-25	25-50	50-100	100-200	200+

Table V.1 Shot length rage division in genre classification representation

A typical SLD distribution for a commercial and a news clip is depicted in Figure V.6. This model clearly distinguishes between these two genre classes, and thus improves the separability of the classification process.

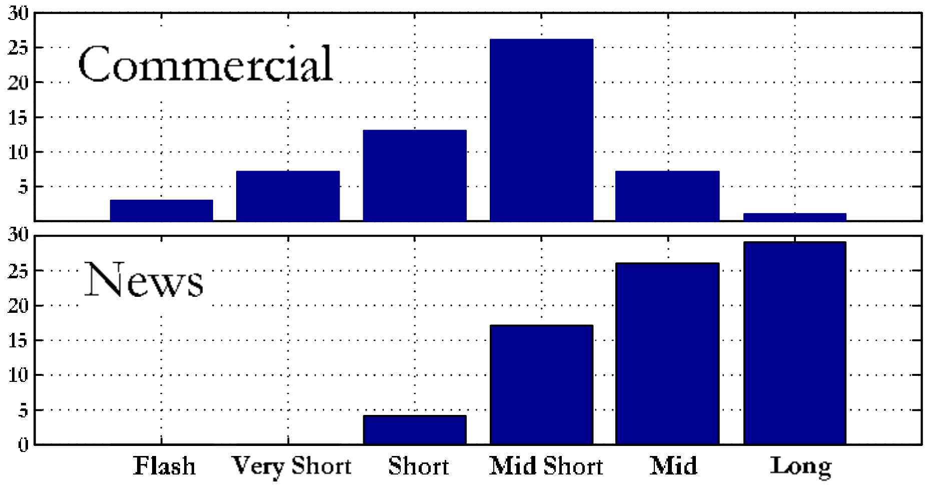


Figure V.6 Typical SLD representation as applied in the genre classification algorithm

On the other hand, the shot length distribution descriptor used in the algorithm for automatic video annotation (described in the following chapter) is generated as a normalised 3-bin histogram. This is due to the fact that the fuzzy annotation module involves heavy computation and there fore the representation dimension has to be minimal. In addition, the SLD representation is that case is strengthened with shot activity descriptor. The bin boundaries are defined applying the normalised division algorithm, because of the presence of the learning set that enables normalisation. The final 3-bin boundaries applied in this module, presented in Table V.2, group shots by SLD into short, mid and long shots.

SHORT	MID	LONG
0-27	28-156	157+

Table V.2 Normalised SLD representation applied in the annotation algorithm

V.3.2.2. Shot Activity

In addition to the strictly temporal representation of SLD model, the major distinction of video media, unlike still images, is in its capability to show motion. The motion, actions and camera work brings magic to the video media. In order to represent the action within a video scene, one needs to analyse the spatial domain features. However, our goal is to analyse the video in its compressed domain. The solution to that problem is to utilise prediction information that has already analysed the spatial domain characteristics, and to give an overall impression of the activity involved in the shot.

V.3.2.2.1 Shot Activity And The Frame Difference Metric

Implemented shot activity representation model describes the normalised distribution of the frame difference metric for each shot throughout the whole video clip. The frame difference metric shows the amount of visual change between them, whether it is camera motion, major object movement or any other kind of visual activity present in the video sequence. All of these events make an important impact on the overall impression of the clip. Therefore, different types of video have different distribution of the overall visual activity. For example, a news clip showing an anchorperson for couple of minutes clearly has no visual activity whatsoever while a commercial clip would have to attract attention of the audiences by offering the most of action and information in a limited duration of an add.

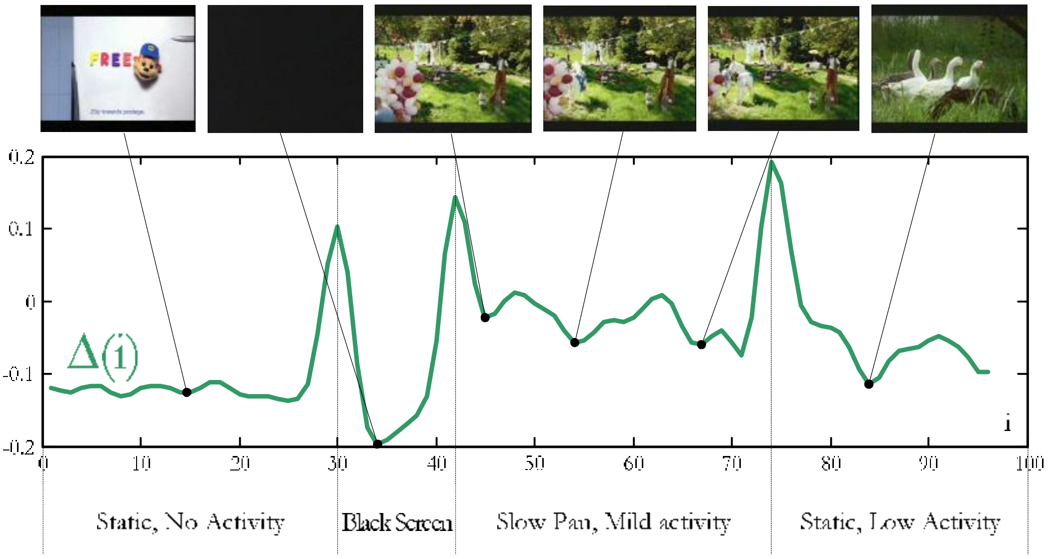


Figure V.7 An example of the shot activity derived from frame difference metric Δ(ι) for sequence ulosci.mpg (frames 0-100)

Let's analyse the shot activity example in Figure V.7. The depicted clip starts with the end of a programme block, being very static with no camera or object motion. The next shot is a program break with black screen shot with duration of approximately 15 frames. The activity in this shot is zero. The next shot is more active, having a slow camera pan of the scene where the objects are mildly moving through the scene. During this shot, the frame difference metric has higher values and follows the content change within the shot. Finally, the last shot is static with no camera movement, but the objects are moving in the scene. This is clearly less active shot than a previous one, but not as inactive as the first two. Therefore, it is obvious that the average value or mean of the frame difference within a shot describes the amount of motion and activity present in the shot. For that reason, mean value of the metric Δ(i) defined in the previous chapter is to be used as the shot activity descriptor. Afterwards, values of the shot activity from each shot will form the Shot Activity (SA) representation model in order to characterise the overall activity of the clip.

V.3.2.2.2 SA Extraction Algorithm

The SA representation is generated as follows. The shot activity is extracted directly from difference metric described in Chapter IV. It is defined as the average of the *Δ(i)* metric within one shot. Thus, each shot gets assigned a unique activity value. For a shot starting at the *i-th* frame and having a shot length *Ni*, the shot activity is calculated as:

$$Sa_i = \frac{1}{Ni}\sum_{j=1}^{Ni}\Delta(i+j) \tag{V.9}$$

The SA model is formed as a normalised histogram of shot activities. The division of bin boundaries is uniform. The sum of the histogram bins is normalised to be one. Because the SA model is applied in the annotation module, there is no need for more that 3 classes of shot activity. A typical example of the SA descriptor is given in the Figure V.8.

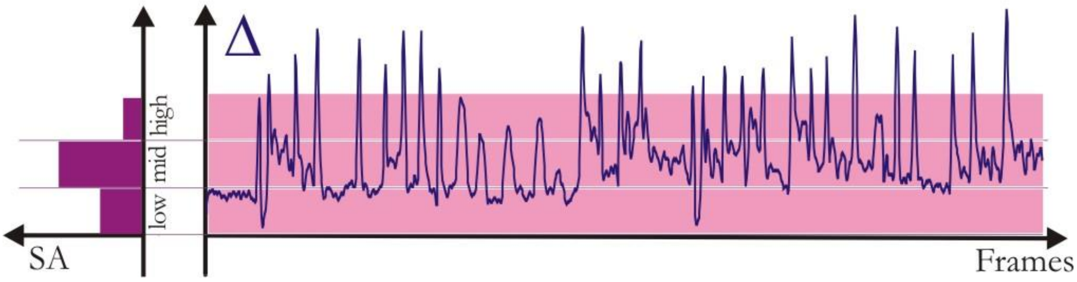


Figure V.8 SA representation example

V.3.2.3. *Hierarchical Colour Descriptor*

Continuing to utilise the set of the robust and efficient low-level descriptors presented in the previous chapter, there has been an inclination towards application of the hierarchically quantised colour descriptor in the final representation model. However, the complexity of the sensible scalable colour representation needed for the automatic annotation module made the exploitation of the colour information inappropriate. The only instance of the colour features in this work is the example of the experimental testbed in Chapter VI. There, the descriptor simplification procedure is driven to its final stage, where only one colour component is left, i.e. the dominant colour. This is described in more detail in Chapter III.

V.4. GENRE CLASSIFICATION

The complexity of the generic semantic annotation of videos is enormous. Therefore, the system needs to shrink the signification space through definition of the context involved in order to achieve any result. Often, this is done by limiting the amount of the semantic concepts to be linked with or by modelling the system for some particular type of application. However, by adopting the semiotic approach to the signification process in visual media and the new computational media aesthetics paradigm, shrinking of the contextual space has to follow the real world contextualisation. In other words, classification into subsets of the analysed media will reach semantic concepts only if the classes formed present types of media present in the real world. The most common classification in the video/TV and film theory is genre classification. This issue is described more in Chapter III.

V.4.1. CLASSIFICATION METHOD

In order to reduce the complexity of the signification process and improve its accuracy, the database is partitioned into sub-classes according to the properties of the extracted representation model. The database is clustered by applying the k-means algorithm to the low-level metric space transformed to the representation model. Two main reasons were considered to choose the k-means algorithm in order to achieve this classification: The huge population of conventional video databases and the excellent performance of the k-means clustering technique when number of clusters *k* is known and the set to be clustered is large. Since the number of sub-classes, i.e. genres, in the underlying annotation problem can be predetermined by the pre-annotated dataset, it can be assumed that the number of clusters *k* is known. Furthermore, it can be assumed that at least one video per cluster in the dataset has been classified beforehand. This entry point in the representation metric space can be used to define the initial centre of the corresponding clusters in the k-means algorithm. The genres involved in the experimental process are grouped in three classes: news, commercials and soaps. News sequences used consist mainly of either anchorperson shots or news reports. Commercials comprise variety of short advertisements, while soaps include parts of sitcoms and soap operas.

V.4.2. K-MEANS CLUSTERING

The k-means algorithm [HARTI1, HARTI2] is by far the most popular clustering tool used in scientific and industrial applications. The name comes from representing each of k clusters C by the mean (or weighted average) c of its points, the so-called centroid. While this obviously does not work well with categorical attributes, it has the good geometric and statistical sense for numerical attributes. The sum of discrepancies between a point and its centroid expressed through appropriate distance is used as the objective function. The choice of the distance function is crucial factor of the algorithm efficency.

Two versions of k-means iterative optimization are known. The first version is similar to EM algorithm and consists of two-step major iterations that (1) reassign all the points to their nearest centroids, and (2) recompute centroids of newly assembled groups. Iterations continue until a stopping criterion is achieved (for example, no reassignments happen). This version is known as Forgy's algorithm [FROGY] and has many advantages:

- It easily works with any norm, and
- It is insensitive with respect to data ordering.

The second (classic in iterative optimization) version of k-means iterative optimization reassigns points based on more detailed analysis of effects on the objective function caused by moving a point from its current cluster to a potentially new one. If a move has a positive effect, the point is relocated and the two centroids are recomputed. It is not clear that this version is computationally feasible, because the outlined analysis requires an inner loop over all member points of involved clusters affected by centroids shifts.

To cluster video representations into genre based classes, a first version of the k-means algorithm is applied. It is simple, straightforward, and is based on the firm foundation of analysis of variances. However, k-means also has its drawbacks:

- The result strongly depends on the initial guess of centroids (or assignments)
- It is not obvious what is a good k to use
- The process is sensitive with respect to outliers
- Only numerical attributes are covered

In our case, all difficulties are eliminated, due to the fact that we operate with numerical attributes with the defined number of clusters and the predefined initial centeroids.

V.4.2.1. K-means Algorithm

The k-means method aims to minimize the sum of squared distances between all points and the cluster centre. This procedure consists of the following steps, as described in [TOU].

1. Choose randomly K initial cluster centres $z_1(1), z_2(1), \ldots, z_K(1)$.

2. At the i-th iterative step, distribute the samples {x} among the K clusters using the relation,

$$x \in C_j(i) \text{if} \left\| x - z_j(i) \right\| < \left\| x - z_k(i) \right\| \tag{V.10}$$

for all k = 1, 2, …, K; k≠j; where $C_j(i)$ denotes the set of samples whose cluster centre is $z_j(i)$.

3. Compute the new cluster centres $z_j(i+1)$, j = 1, 2, …, K such that the sum of the squared distances from all points in $C_j(i)$ to the new cluster centre is minimized. The measure which minimizes this is simply the sample mean of $C_j(i)$. Therefore, the new cluster centre is given by

$$z_j(i+1) = \frac{1}{N_j} \sum_{x \in C_j(i)} x,\ j = 1, 2, \ldots, K \tag{V.11}$$

where N_j is the number of samples in $C_j(i)$.

4. If $z_j(i+1) = z_j(i)$ for j = 1, 2, …, K then the algorithm has converged and the procedure is terminated. Otherwise go to Step 2.

V.4.2.2. Distance Functions

The choice of the dissimilarity measure is central in setting up the classification algorithm. Two main distance functions were implemented: bin-to-bin histogram distance and Earth Mover's Distance.

V.4.2.2.1 Bin-To-Bin Histogram Distance Functions

In this category of distance functions only pairs of bins in the two histograms that have the same index are matched. The dissimilarity between two histograms is a combination of all the pairwise differences. A Minkowski-form distance is applied:

$$d_{L_r}(H,K) = \left(\sum_i |h_i - k_i|^r \right)^{\frac{1}{r}} \qquad \text{(V.12)}$$

The L_1 distance is often used for computing dissimilarity between colour images [SWAIN]. Other common usages are L_2 and L_∞. In [STICKER] it was shown that for image retrieval the L_1 distance results in many false negatives because neighbouring bins are not considered. Therefore, a cross bin dissimilarity measure is applied in the final system. L_2 Minkowski-form bin-to-bin measure was used only in the development process.

V.4.2.2.2 Earth Mover's Distance

The distance between two single perceptual features can be found by psychophysical experiments. For example, perceptual colour spaces were devised in which the Euclidean distance between two single colours approximately matches human perception of the difference between those colours. This becomes more complicated when sets of features, rather than single colours, are being compared. The problems caused by dissimilarity measures that do not handle correspondences between different bins in the two histograms became the main focus of the histogram-based retrieval. This correspondence is key to a perceptually natural definition of the distances between sets of features.

Intuitively, given two distributions, one can be seen as a mass of earth properly spread in space, the other as a collection of holes in that same space. Then, the EMD measures the least amount of work needed to fill the holes with earth. Here, a unit of work corresponds to transporting a unit of earth by a unit of ground distance.

Computing the EMD is based on a solution to the well-known transportation problem [HITCH]. This can be formalized as the following linear programming problem: Let $P=\{(p_1,w_{p1}),\ldots,(p_m,w_{pm})\}$ be the first signature with m clusters, where p_i is the cluster representative and w_{pi} is the weight of the cluster; $Q=\{(q_i,w_{qi}),\ldots,(q_n,w_{qn})\}$ the second signature with n clusters; and $D = [d_{ij}]$ the ground distance matrix where d_{ij} is the ground distance between clusters p_i and q_j. We want to find a flow $F = [f_{ij}]$, with f_{ij} the flow between p_i and q_j, that minimizes the overall cost

$$\mathrm{WORK}(P,Q,F) = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij} \qquad \text{(V.13)}$$

subject to the following constraints:

$$
\begin{aligned}
& f_{ij} \geq 0 \quad 1 \leq i \leq m, 1 \leq j \leq n \\
& \sum_{j=1}^{n} f_{ij} \leq w_{pi} \quad 1 \leq i \leq m \\
& \sum_{i=1}^{m} f_{ij} \leq w_{qj} \quad 1 \leq j \leq n \\
& \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{m} w_{pi}, \sum_{j=1}^{n} w_{qj} \right) \quad 1 \leq j \leq n
\end{aligned}
\tag{V.14}
$$

Constraint (1) allows moving "supplies" from P to Q and not vice versa. Constraint (2) limits the amount of supplies that can be sent by the clusters in P to their weights. Constraint (3) limits the clusters in Q to receive no more supplies than their weights; and constraint (4) forces to move the maximum amount of supplies possible. We call this amount the total flow. Once the transportation problem is solved, and we have found the optimal flow F, the earth mover's distance is defined as the work normalized by the total flow:

$$
\mathrm{EMD}(P,Q) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}} \tag{V.15}
$$

This distance function is utilised in the process of clustering as a dissimilarity measure between SLD histogram representations.

V.5. SUMMARY

This chapter described in detail the importance of the efficient and robust video representation in a CBVIR system. After bringing the critique of the sate-of-the-art methods in the video representation field, a set of essential requirements was defined in order to target more efficient representation. Furthermore, a model based on computational media aesthetics paradigm is presented. It utilises production knowledge to achieve requirements set in the beginning of the chapter. The features involved are shot length distribution and normalised shot activity. Finally, a contextual classification of videos into genres is presented. This algorithm exploits representation model to cluster video clips into syntactically similar groups applying k-means algorithm as a classification tool.

VI. EXPERIMENTAL TESTBED

VI.1. OVERVIEW

The algorithms were implemented on Windows platform, using C/C++ programming tools, particularly MS Visual C++ 6 and some specific libraries available online. The collection of C++ classes called Mpeg Development Classes, implemented by Dongge et al. [DONGGE], was used as the main tool for manipulation with MPEG streams, while Berkeley mpeg2codec was used as the reference MPEG codec. The tool used for data visualisation and some experiments was Mathworks' Matlab Suite.

All experiments were performed on a Pentium III 750MHz workstation running on the Windows 2000 platform. Experimental environment was designed to benchmark efficiency and robustness of the algorithms involved. Ground truth for the shot detection evaluation was made manually, by precise labelling the shot boundaries. Experimental dataset is described in the following section.

VI.2. EXPERIMENTAL DATASET

The majority of the test sequences involved in the experiments was produced by Multimedia & Vision Research Lab, Queen Mary, University of London. Additional dataset was provided by Computer Vision Department, Dublin City University, Dublin, Ireland.

Two video grabber cards were used in the creation of the experimental dataset locally:

i) Nebula Electronics DigiTV, courtesy of the BUSMAN project, recorded the MPEG2 digital broadcast stream directly to the local multimedia server

ii) Pinnacle DV500 required transcoding to produce the needed MPEG2 compression format form its original M-JPEG compression.

- These sequences were all in CIF resolution 352x288pixels, encoded in multiple copies with the bitrates ranging from 1 to 8 Mbps. Sequences produced by Computer Vision Department at Dublin City University were captured at their local multimedia network Físchlár (http://www.cdvp.dcu.ie/) and were encoded as CIF MPEG1 with resolution 356x288pixels. For the computational speed evaluation purposes, a range of resolutions was used, from QCIF to HDTV, all in MPEG2 format. Test sequences ulosci.mpg and news136.mpg referenced in this work can be downloaded from:

www2.elec.qmul.ac.uk/~janko/ulosci.mpg, and

www2.elec.qmul.ac.uk/~janko/news136.mpg.

Their visual summaries with the corresponding difference metrics are given in Annex X.3.

VI.3. AUTOMATIC VIDEO ANNOTATION SYSTEM

As a part of a bigger CBVIR system, video representation and classification module presented in the previous chapter is evaluated by supporting an automatic video annotation system. Low-level features extracted from the video sequences and their representations are fed to the video annotation system input. Relying upon different video representations annotation system creates a set of inference rules linking low-level features with high-level user defined concepts. Results substantiated our expectations that a choice of suitable representation significantly affects annotation outcomes even if aimed only at a modest process of labelling from a predefined keyword lexicon.

VI.3.1. ANNOTATION SYSTEM OVERVIEW

The proposed evaluation system annotates video sequences automatically using knowledge from a pre-annotated dataset. It creates representations from a set of low-level video features and infers the association rules between them and high-level concepts from a pre-defined lexicon, listed in Table VI.1. The used paradigm for automatic semantic annotation is depicted in Figure VI.1

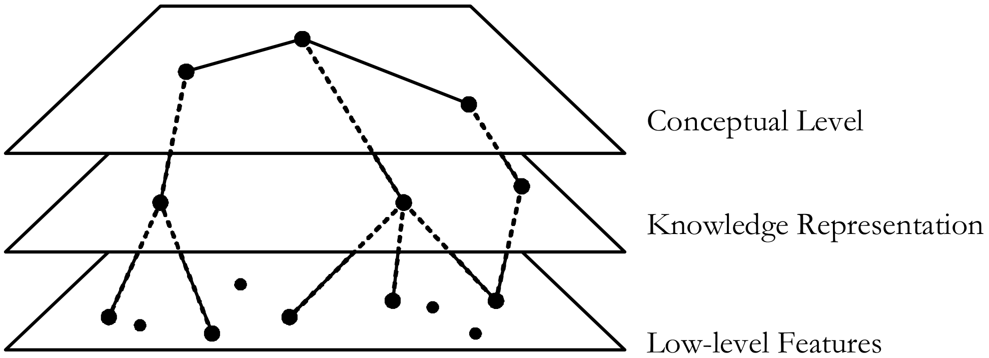


Figure VI.1. Mapping between low-level features and high-level concepts.

The system consists of two units: learning and annotation units. The learning unit consist of three sequential modules: low-level feature extraction, knowledge representation and rule mining. This unit uses pre-annotated videos to generate rules that link a particular low-level representation of the sequence with a corresponding label from the lexicon. Figure VI.2 shows a flowchart of the learning unit.

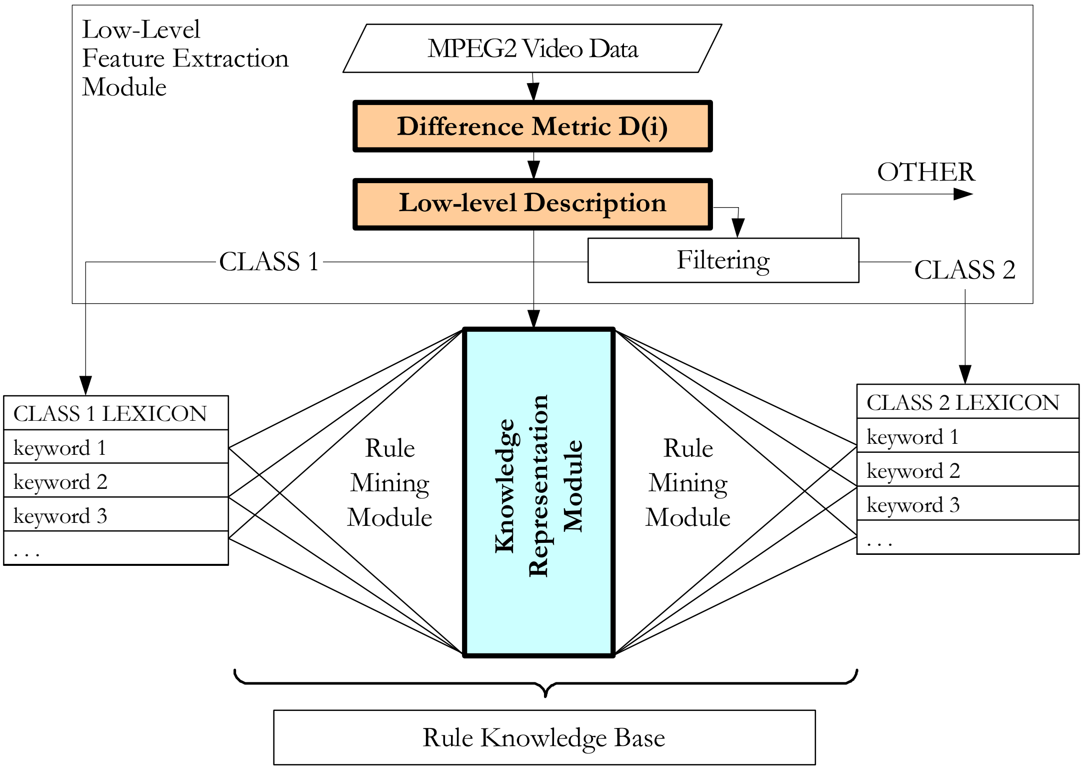


Figure VI.2 Flowchart of the learning unit

The annotation unit automatically infers concepts and assign them to videos using the rules and supports generated by the learning unit. Each time new content is added to the database new concepts can emerge from the constant evaluation of the confidence and support measures leading to continuously changing metadata and inference rules.

The low-level feature extraction algorithm and the video representation model are being implemented here as the first module of the learning unit. Firstly, system parses video into shots and extracts a representative set of key-frames. Exploiting descriptors extracted from the temporal structure and key-frames, a subsequent filtering stage classifies videos into contextual sub-classes in order to limit the signification space and reduce rule-mining complexity. Each sub-class or genre has its own lexicon, on which the rule-mining algorithm is applied. Exploiting information from the video representation model a fuzzy set is defined by assigning fuzzy boundaries to the numerical descriptors and labelling each fuzzy class with high-level representations.

The third module of the learning unit performs rule mining. Initially a set of low-level features is extracted form the pre-annotated dataset. Using the knowledge representation provided by the expert user, for each video clip a set of features is mapped into words and a log of all available transactions is created. From this log and using fuzzy membership values, rules are mined and a list of possible rules is generated. By filtering the initially generated association rules, the learning unit creates a more dedicated set of rules. These selected rules are then used as a knowledge base in the annotation process.

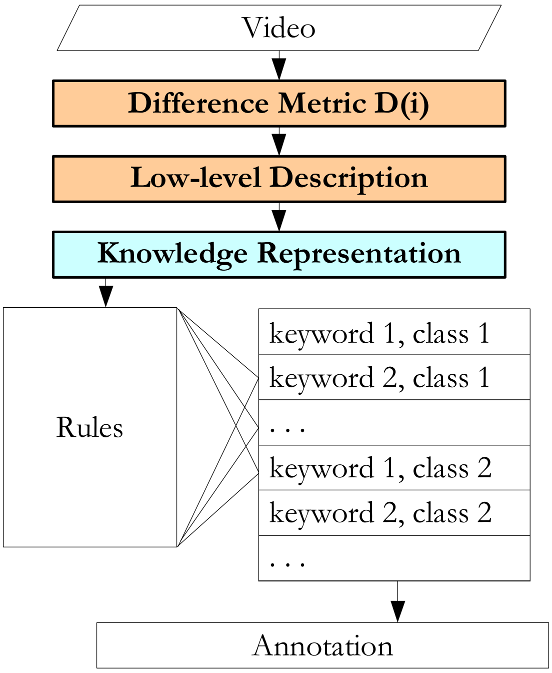


Figure VI.3 Flowchart of the annotation unit.

The annotation unit, as depicted in Figure VI.3, automatically assigns high-level concepts from the lexicon to any new video added to the database. As in the learning unit, the automatic annotation process starts with the extraction of the low-level video features and the automatic generation of descriptors. Using the video representation model defined in the previous chapter, the new videos are mapped into the corresponding feature-related words by a fuzzification process. Finally, a fuzzy inference module generates the pattern of assigned labels and outputs a set of high-level concepts from the lexicon. In this process new rules can be created and added to the rule knowledge base.

VI.3.2. THE LEARNING UNIT

For the sake of clarity, a simple system that uses only the dominant colour descriptor of automatically extracted key-frames and a simple lexicon is presented throughout the following descriptions of the implemented system. This illustrative example targets the annotation of broadcasting news and it will be referred as "dominant colour to annotate news" (DCAN) example in the sequel. For a given video sequence, the low-level feature extraction module generates a set of shot boundaries and representative key-frames, as described in the previous subsection. The dominant colour descriptor defined in the previous chapter is extracted from each key-frame. For a lexicon with two concept-related keywords a knowledge representation that exploits dominant colour variations within the sequence is designed. Changes of the dominant colour within the sequence are estimated using the quadratic colour histogram distances γ as defined in the MPEG7 XM [XM]. The video descriptor δ is then defined as the mean value of the γ-distances between the first key-frame and all the other key-frames in the same sequence. Small dominant colour variations within the sequence should correspond to the label "anchorperson" in the context of news clips, while strong changes should refer to the label "report". These two words form the concept-related part of the lexicon. Knowledge representation for this low-level feature is given by two fuzzy sets related to the mean values. The names of the fuzzy sets, mean of the distances between dominant colour high and low (ddcm_high, ddcm_low), are added to the lexicon as feature-related words, as given in Table VI.1.

concept-based keywords	news	commercial	soap
	report	anchorperson	interview
	health_beauty	program_schedule	tv_add
feature-based keywords	ddcm_high	ddcm_low	
	short	mid	long
	high_act	mid_act	low_act

Table VI.1 List of concept-based and feature-based keywords used in the annotation unit

In Figure VI.4 the results obtained for 20 samples and the representation of the low-level feature in the fuzzy space are depicted. In this figure the notation related to the knowledge representation module is used to link the obtained result with the description given in the next paragraph. More detailed description of the inference and the rule mining module can be found in the Annex VI [DORADO].

For the DCAN example the fuzzy system generates feature-related words describing low or high mean values of the difference between dominant colour descriptors as *ddcm_low* and *ddcm_high* for each video clip processed. A log of transactions was created using 4 words: ddcm_low, ddcm_high, anchorperson and report. The rule mining process found two rules with 100% of confidence:

ddcm_low → anchorperson

ddcm_high → report

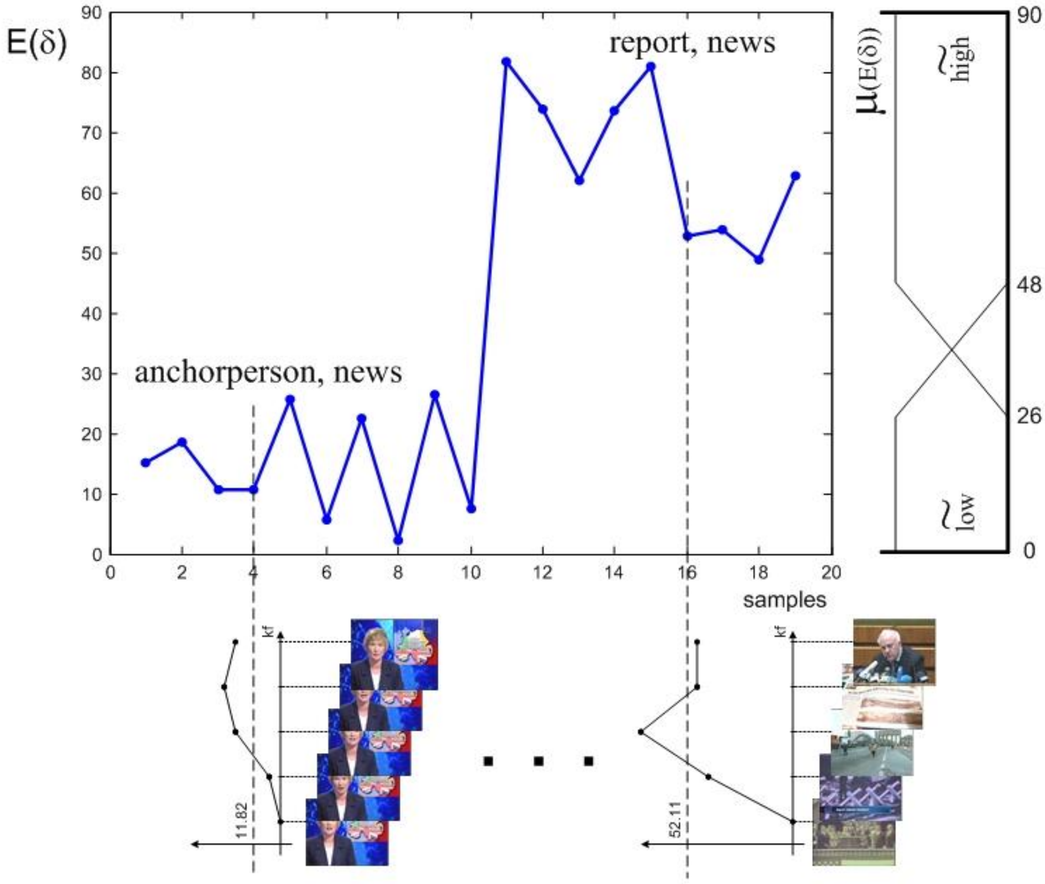


Figure VI.4 Knowledge representation for changes on dominant colour in key-frames.

This set of rules form the rule knowledge base for this example system. Basically, these two rules imply that the mean difference between the dominant colour descriptors is low for "anchorperson" clips and high for "report" clips. The real automatic annotation system utilises representation model presented in the Chapter V. Results of the evaluation are given in the following chapter.

VI.3.3. AUTOMATIC VIDEO ANNOTATION

Once a knowledge base with a set of association rules is created the system uses an inverse rule-based inference process to identify candidate concepts for the annotation of each new video added to the database. The annotation unit uses a fuzzy inference strategy which involves three basic modules: fuzzification, fuzzy inference and defuzzification. The input of the fuzzification module is a real number corresponding to an instance of a variable. Adopting the same principle as in the knowledge representation module of the learning unit, the set of low-level features forms a set of representation variables for the new video clip. The degree of membership for each fuzzy set is calculated using the membership functions defined in the knowledge representation step and mapping features into keywords. The output of this module is a fuzzy value.

The fuzzy inference module uses rules in the form IF <condition> THEN <action>. Inference rules are not a free form of the natural language; they are limited to a set of words and a strict syntax. In this case, inference rules are limited to the keywords from the lexicon. Here, <condition> expresses the instances of low-level features and <action> denotes annotations. Each <condition> of a rule corresponds to a specific value of a fuzzy input. This input value is a result of the fuzzification module. Each <action> of a rule corresponds to a fuzzy output. In addition, this kind of rules have two representative characteristics: they are qualitative rather than quantitative and each <condition> is related to an appropriate <action>. The importance of these rules lays in the possibility of representing human knowledge by a hierarchical model. Besides, these rules are relatively simple and consistent with the way human reasoning works. The fuzzy inference module calculates the fuzzy output values for the corresponding variable. It uses the relationship between input and output variables using the base of linguistic rules provided by the learning unit. At this point, a number of rules can have different degrees of truth leading to competition between the results. Using an *aggregation's operator* the instances of the <condition> part of rules are combined in order to determine the value of the rule. This value is used to determine the <action> part of the rule. The procedure is repeated for all rules from the rule knowledge base. It is possible that an output fuzzy variable has a fuzzy set as <action> in several rules. The *composition's operator* is used to determine the final value of this fuzzy set.

The defuzzification module combines the fuzzy values of each output variable to obtain a real number for each variable. In this module a weighted average method is used. It combines fuzzy values using weighted averages to obtain the resulting crisp value.

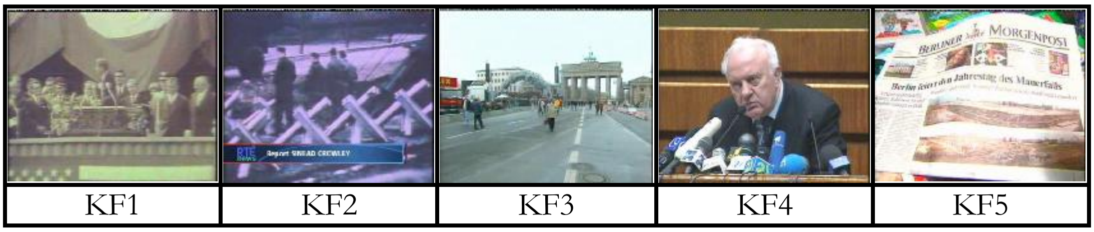


Figure VI.5 A set of five key-frames from the sequence news136.mpg.

To illustrate this process the sequence of key-frames KF1-5 extracted form the video news136.mpg is used. These five key-frames are shown in Figure VI.5. The aim of this exercise is to show the behaviour of the annotation unit for the rule knowledge database derived for the DCAN example. Following the procedure for representation, the distances from KF1 to the other key frames are calculated. These values along with the corresponding mean are given in Table VI.2.

$\delta 1$	$\delta 2$	$\delta 3$	$\delta 4$	$\bar{x}$
83.4	82.8	78.7	82.4	81.83

Table VI.2 Distances between the DCAN descriptor of KF1 and the other key frames form news136.mpg.

The fuzzification process based on the knowledge representation generates $\mu_{\tilde{A}}(\bar{x}) > 0$ and $\mu_{\tilde{B}}(\bar{x}) = 0$. So, the mapping function for this instance of the mean is $\tilde{M}(x) = \{\tilde{A}\}$, where $\tilde{A}$ is *ddcm_high*. Using this word the rule mining process suggests the word “report” for the annotation of the sequence of key-frames KF1-5.

VI.4. SUMMARY

This chapter describes the experimental environment for the evaluation of the developed algorithms. Firstly, the video dataset used in experiments is specified. Furthermore, since the objective evaluation demands a wider CBVIR system an automatic video annotation system is presented in Section 3. It benchmarks the efficiency of the video representation model by exploiting fuzzy logic methods for labelling the video data.

VII. RESULTS

VII.1. OVERVIEW

This chapter presents experimental results of the evaluation process. First of all, results of the temporal analysis evaluation are given. This includes difference metric extraction for the three presented algorithms, metric simplification process, key frame extraction and motion filed estimation. Experiments conducted on the hierarchical quantisation of the colour descriptor are presented afterwards. The chapter concludes with an assessment of the video representation model and the involved genre classification system.

VII.2. METRIC EXTRACTION

In order to evaluate the temporal video analysis a set of manually labelled video sequences are used as the ground truth. The precision of the manual labels is frame accurate with classification of the transition types into abrupt or gradual. The overall video material consists of over 7 hours of various content types. The dominant programme types are news, commercials and soaps.

VII.2.1. FRAME-TO-FRAME DIFFERENCE METRICS

To begin with, the typical behaviour of the first temporal segmentation algorithm is depicted in the following example. Therefore, a sample MPEG2 video sequence *shot.m2v* is generated having three abrupt shot changes at 6th, 16th and 23rd frame.

As depicted in Figure VII.1, the first cut is positioned at rear **b** frame, and as proposed, it is clear that the level of forward reference is high at previous **B** frame $\beta(5)$, and that at the present frame there is strong backward referencing $\beta(6)$. On the other hand, for the 16th **I** type frame there are significant levels of forward prediction on both 13th and 14th frame, i.e. $\varphi(13)$ and $\varphi(14)$ are high. Finally, the 23rd **B** type frame brings the strong visual change and therefore both bi-directional frames have high values of $\beta(23)$ and $\beta(24)$. The reason for that is because both frames are predicted only by the coming reference frame an not at all by the previous reference frame.

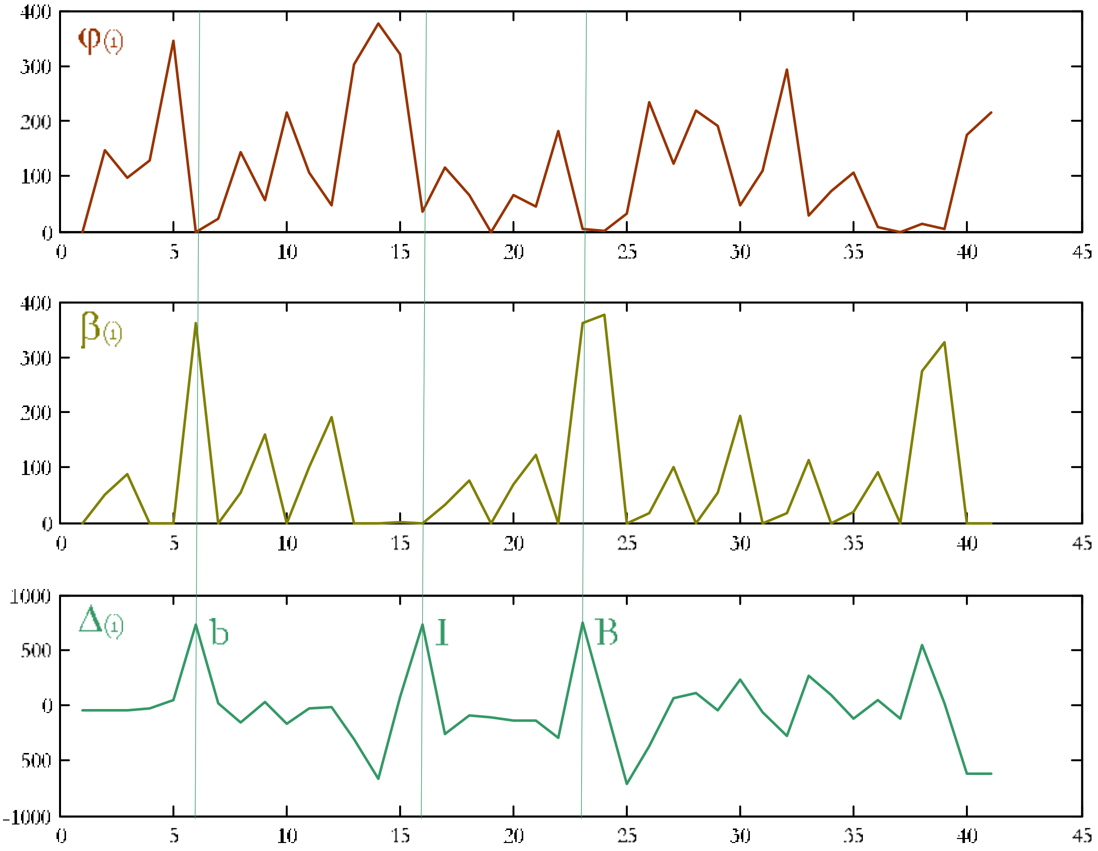


Figure VII.1 Detection of the cuts on the 6th, 16th and 23rd frame in the test sequence

VII.2.2. SHOT DETECTION EVALUATION PROCEDURE

In order to compare the efficiency, robustness and preciseness of the shot detection algorithms presented here, a unified procedure for result evaluation is applied. It has been adopted by many researchers in the publications and surveys on temporal video analysis [GARGI, BOREC].

The applied statistical performance evaluation is "based on the number of missed detections (MD's) and false alarms (FA's), expressed as recall and precision" [GARGI]:

$$\text{Recall} = \frac{\text{Detects}}{\text{Detects} + \text{MD's}}, \text{Precision} = \frac{\text{Detects}}{\text{Detects} + \text{FA's}} \tag{VII.1}$$

Recall is defined as the percentage of desired items that are retrieved. Precision is defined as the percentage of retrieved items that are desired items. Recall and precision are commonly used in the field of information retrieval. It is difficult to make comparisons between algorithms based on recall and precision values. For example, an automated video indexing system that uses a human operator to screen the results requires a high recall. A system that summarizes video by selecting a key-frame for each minute of video places higher emphasis on precision. In any application a trade-off must be made between recall and precision. It may or may not be acceptable to retrieve one extra shot boundary that would otherwise be missed at the expense of retrieving 100 non-boundaries incorrectly.

As said before, manually detected positions of the shot boundaries were taken as the ground truth. There were three main categories of video material analysed:

- NEWS; long monotonous sequences with mainly abrupt changes,
- SOAP OPERA; average shot length with some gradual changes and editing effects
- COMMERCIALS; short shots with a lot of gradual changes and editing effects

The initial cut detection procedure showed good results for abrupt changes while the gradual changes had intolerable number of both misses and false positives. Form 127 transitions, 17 were gradual, and the algorithm detected only 9 of them. Therefore, the frame difference metric needed improvement. The results are presented in Table VII.1.

	DETECT	MISSED	FALSE	RECALL	PRECISION
NEWS	87	2	6	98%	94%
SOAP	92	2	9	98%	91%
COMMERCIALS	127	9	16	94%	88%

Table VII.1 Cut detection statistics

VII.2.3. RANDOM DISTANCE DIFFERENCE METRICS

Comparing to the cut detection, evaluation of the gradual changes detection is always a delicate issue due to the variety of potential transition types and digital editing effects. A simplified classification of the transitions to cuts and gradual changes is applied here. By unifying the gradual types into one class, a complexity of the gradual transition categorisation is avoided. Any type of the shot transition that is longer than 3 frames is considered as gradual. First, a dead end direction towards using MPEG motion vectors as a feature to detect shot boundaries is described in more detail.

VII.2.3.1. MPEG motion vectors as a detection feature

The results of the algorithm that analyses motion vector fields turned out to be very poor, since less than 5% of MacroBlocks with defined motion vectors was obtained in frame areas neighbouring the shot boundary. If the union of the MacroBlock sets having forward (**Φ**) or backward (**B**) prediction is denoted Γ:

$$\Gamma(i) = \Phi(i) \bigcup B(i) \tag{VII.2}$$

then the fraction of defined motion vectors in a frame can be defined as:

$$\gamma(i) = \frac{\#\Gamma(i)}{H \cdot W} \cdot 256 \tag{VII.3}$$

where H and W are the frame height and width in pixels respectively. In Figure VII.2 a defined motion vector fraction is given in a short commercial clip. It is obvious that the number of defined motion vectors falls as soon as a transition occurs.

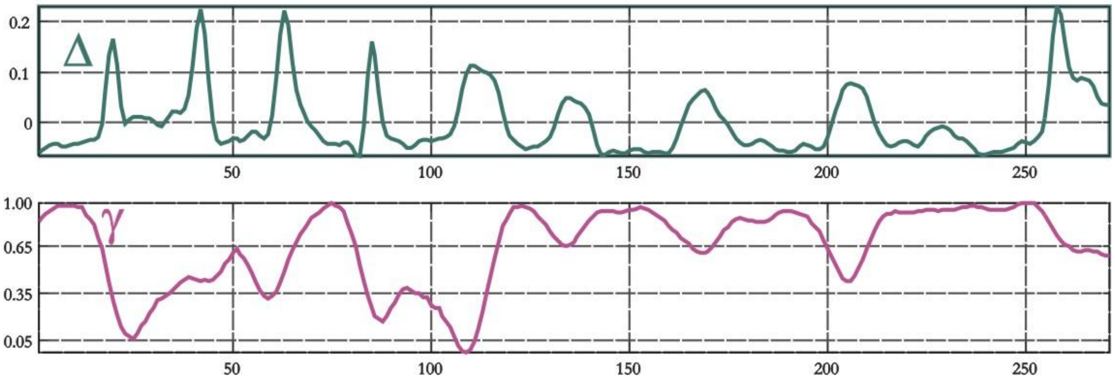


Figure VII.2 Obvious lack of defined motion vectors during transitions

If we consider the granularity of the MacroBlocks in case of different MPEG resolutions and bitrates, things doesn't change much, as shown in Table VII.2:

Mbps/Res	0.7/CIF	1.5/CIF	3/PAL	6/PAL
$\overline{\gamma}$	0.68	0.70	0.84	0.86
$\overline{\gamma_T}$	0.08	0.08	0.11	0.10

Table VII.2 Inpact of the higher bitrates/resolution on MV definition is minor

Where, if the set of the shot boundaries is denoted as Λ, $\overline{\gamma}$ is the average defined MV fraction for a whole clip and $\overline{\gamma_T}$ is the local average, as defined in the following formulae:

$$\overline{\gamma} = \frac{1}{N}\sum_{i=1}^{N}\gamma(i), \quad \text{and} \quad \overline{\gamma_T} = \frac{1}{10 \cdot \#\Lambda}\sum_{\forall i \in \Lambda}\sum_{\varepsilon=-5}^{5}\gamma(i+\varepsilon) \tag{VII.4}$$

VII.2.3.2. Metric Evaluation

Focusing on the metric extracted using random distance method that exploits information on the inter-frame referencing, let us analyse Figure VII.3. The figure shows an example of the raw frame difference metrics $\Delta_D(i)$, defined in the previous chapter.

In the generated sample video clip, there are three types of the shot changes: cut on the 48th frame, wipe from the 82nd to the 121st frame and dissolve from the 160th to the 183rd frame. The graph shows unclear detection of those three changes, regardless of the change type because of the strong additional noise and weak peaks for longer gradual transitions. This method showed additional detection difficulties on sequences with high motion during the shot changes.

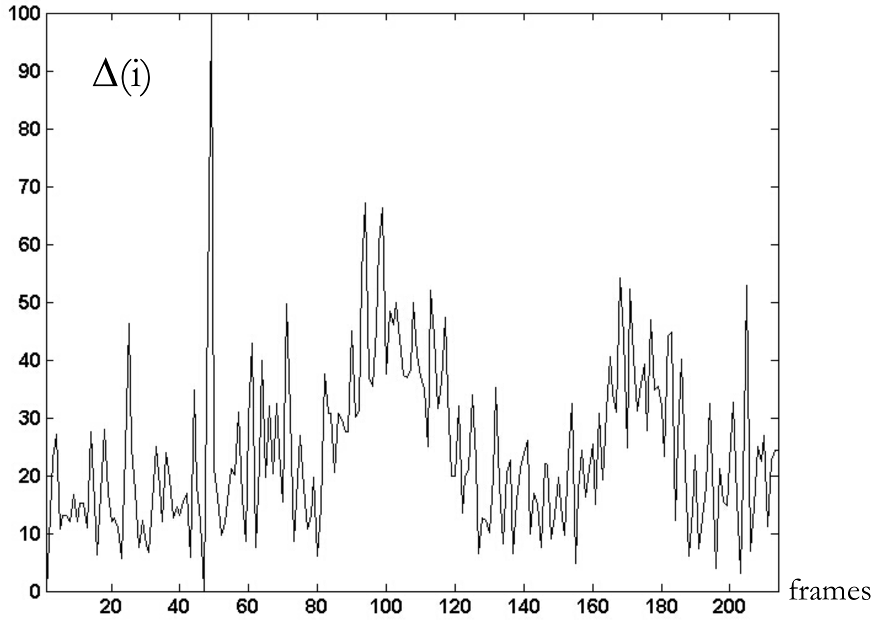


Figure VII.3 Difference metrics *Δ* for three types of gradual changes: cut, wipe and dissolve in the sequence news136.mpg (frames 100-320)

After the metric noise reduction and the detection procedure described in Chapter IV a set of shot boundaries is extracted. The same approach as before was used for the statistical evaluation of this method. The results are shown in Table VII.3.

	DETECT	MISSED	FALSE	RECALL	PRECISION
NEWS	88	1	5	98%	95%
SOAP	92	2	9	98%	91%
COMMERCIALS	130	6	10	96%	92%

Table VII.3 Statistics of the gradual changes detection

VII.2.4. RESULTS FOR THE GENERALIZED DIFFERENCE METRIC

Evaluation of the final metric is divided into two steps because of its utilisation in both shot boundary location and key-frame definition task. Throughout the evaluation process a commercial clip ulosci.mpg is analysed as an example. The stages of the temporal analysis of the sample clip are given in the Figure VII.4.

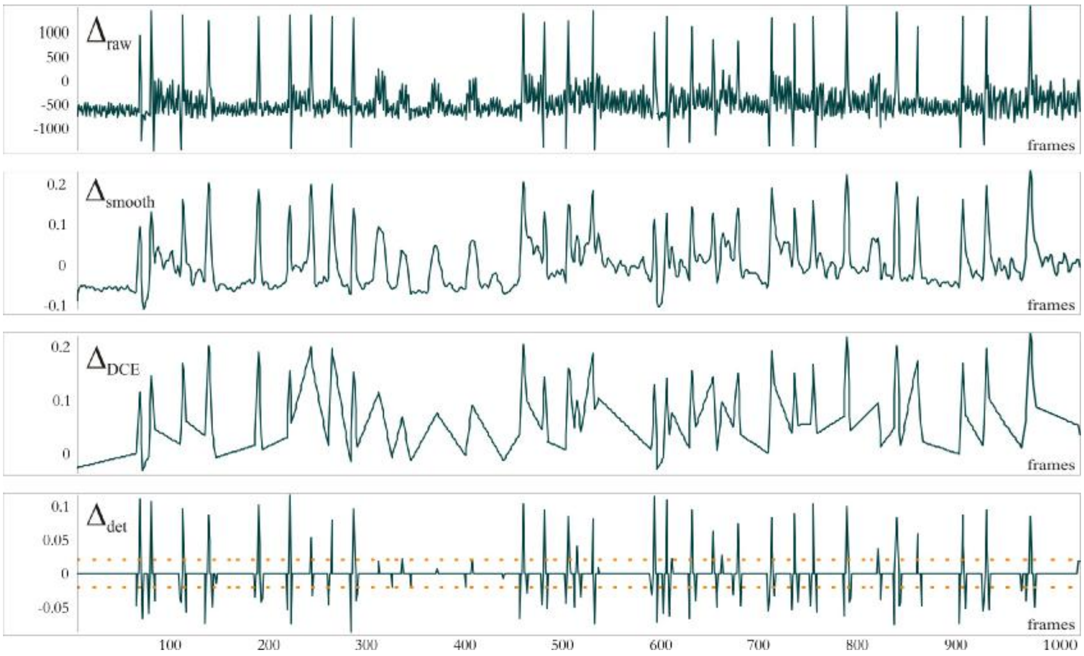


Figure VII.4 Stages in the temporal analysis for the sample mpeg file ulosci.mpg

VII.2.4.1. Shot detection evaluation

As described before, the evaluation of the generalized frame difference metric is based on the manually labelled ground truth of approximately 7 hours of MPEG video material. The results are presented in Table VII.4.

	DETECT	MISSED	FALSE	RECALL	PRECISION
NEWS	267	0	6	100%	99%
SOAP	276	6	27	98%	91%
COMMERCIALS	402	6	18	99%	97%

Table VII.4 Shot changes detection results

The boundaries are determined by analysing the difference metric Δ as given in the following formula:

$$\Delta_{\text{detection}} = \frac{\partial^2 \Delta_{\text{DCE}}(\text{i})}{\partial \text{i}^2} \tag{VII.5}$$

The set of shot boundary locations Λ is determined by thresholding the detection curve with the constant threshold Ψ:

$$\Lambda(\text{i}) = \{\text{i} \mid \Delta_{\text{detection}}(\text{i}) \geq \Psi, \Psi = \text{E}_{\Delta} + 2 * \sigma_{\Delta}\} \tag{VII.6}$$

where E_{Δ} and σ_{Δ} are the mean and the standard deviation of the peak metric $\Delta_{\text{detection}}$:

$$E_{\Delta} = \frac{1}{M}\sum_{j=1}^{M}\Delta_{\text{detection}}(j), \quad \sigma_{\Delta} = \frac{1}{M}\sum_{j=1}^{M}\left(\Delta_{\text{detection}}(j) - E_{\Delta}\right)^{2} \qquad \text{(VII.7)}$$

In order to evaluate the robustness of the algorithm to changes in the MPEG compression rates and resolution, a next experiment was conducted.

A chosen set of representative clips is transcoded in multiple bitrates and resolutions and the results are given in Table VII.5. Value pairs presented in the table are recall/precision values.

Res.\Bitrate [Mbps]	0.7	1.0	1.5	2.0	3.0	4.0	6.0	8.0
QCIF [176x144]	91/72	92/94	98/95	98/94	x	x	x	x
CIF [352x288]	94/99	99/100	100/97	97/88	91/85	80/77	x	x
PAL [720x576]	x	95/96	99/99	99/99	90/92	80/81	77/80	78/75
HDTV [1280x720]	x	x	91/84	94/87	91/87	81/74	74/68	66/71

Table VII.5 Robustness on bitrate/resolution

The graphical presentation of these results is given in Figure VII.5. The values presented in the figure are the sum of the recall and precision. Some of the resolution/bitrate pairs are avoided because of its irrelevancy.

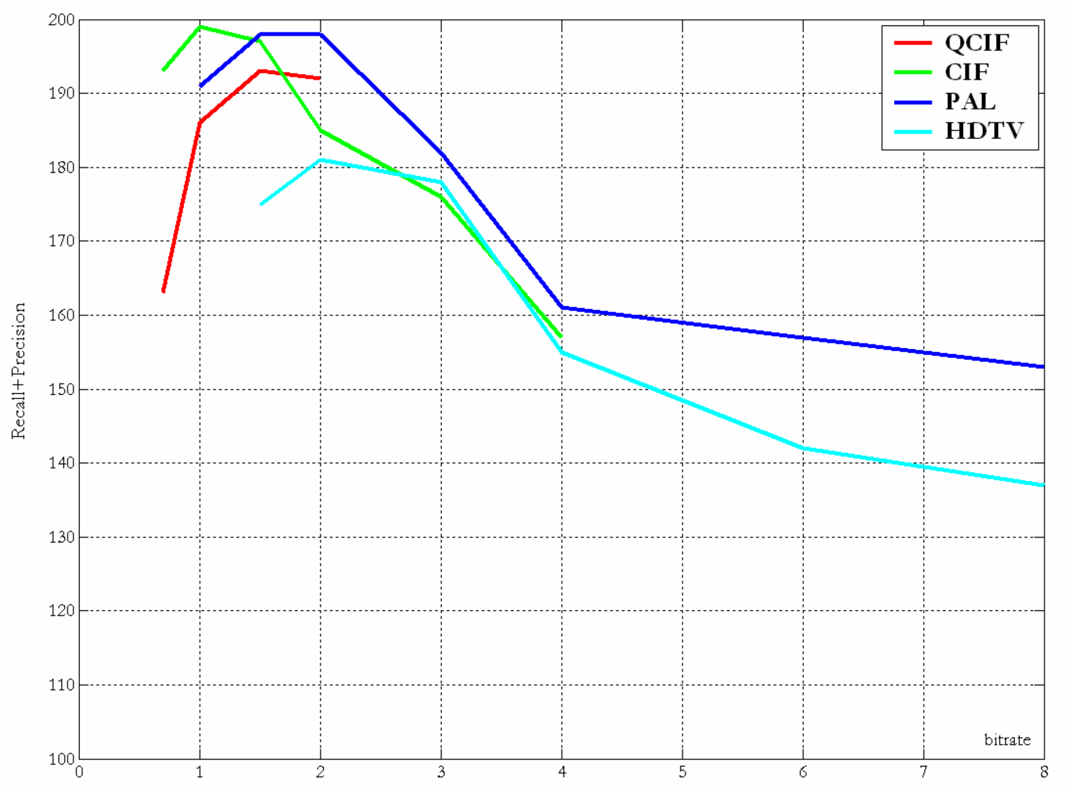


Figure VII.5 Robustness to the MPEG resolution/bitrate change

VII.2.4.2. Metric Simplification

After Gaussian smoothing of the raw difference metric curve simplification algorithm makes a scale-space of the metric curves on difference levels of detail. Two stages of the simplification algorithm are presented in Figure VII.6.

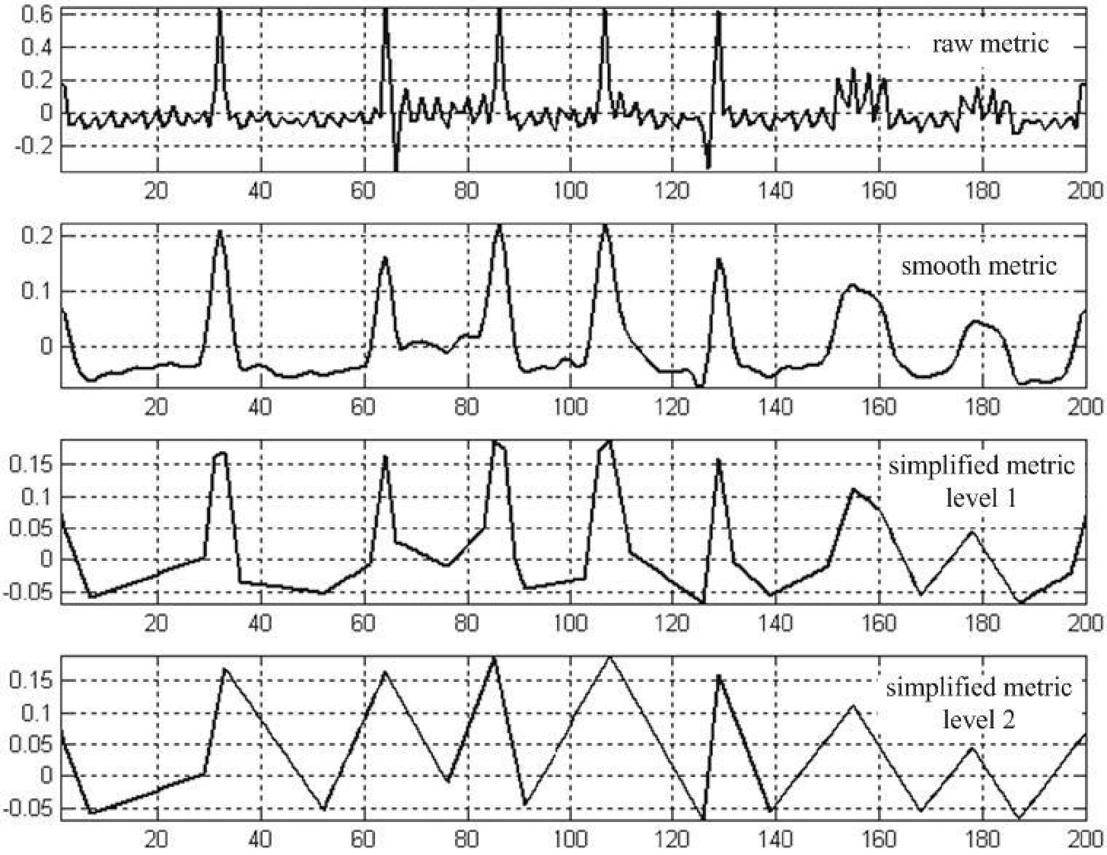


Figure VII.6 DCE algorithm results

During the curve simplification DCE algorithm deletes less important changes one by one without dislocating the vertices of the main difference metrics. Major values like frame difference and location of the peak in the function are stable.

There are three ways of determining the desired stage of the simplification process: a-priori defined number of key-frames, automatic analysis of the cost function and interactive.

If the user needs a predefined number of key-frames to make a visual summary, than the algorithm finalises the process when the number of key-points has been reached. Number of key-frames cannot be determined exactly, but in the final stages, the number of key-points is approximately double of the number of key-frames. This is strictly speaking an empirical conclusion, but the results show that by following this rule, final number of key-frames varies ±2%!

Another way of determining the final granularity of the curve in the simplification process is to track values of the cost function K:

$$K(s_1,s_2) = \left|\beta(s_1,s_2)\cdot\left(l_1+l_2\right)\cdot P_{\Delta(ABC)}\right| \qquad \text{(VII.8)}$$

where the variables involved in the equation are given as (see Figure IV.10):

$$\delta_i = \Delta(i+1)-\Delta(i)\ ,\ l_i = \sqrt{\tau_i^2+\delta_i^2} \qquad \text{(VII.9)}$$

$$\beta(s_i,s_{i+1}) = \mathrm{acrtg}\left(\delta_i/\tau_i\right)-\mathrm{acrtg}\left(\delta_{i+1}/\tau_{i+1}\right) \qquad \text{(VII.10)}$$

$$P_{\Delta ABC} = \frac{1}{2}\left(\delta_i\tau_i+\delta_{i+1}\tau_{i+1}\right) \qquad \text{(VII.11)}$$

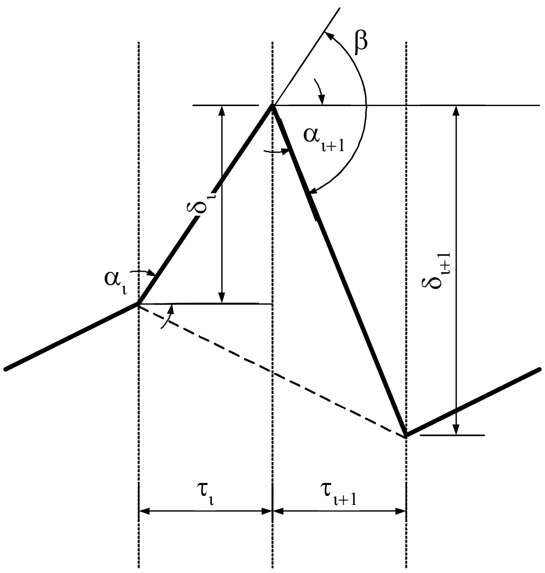


Figure VII.7 DCE linearization of two adjacent line segments

In the Figure VII.8 a cumulative cost function σ_K is given for 6 stages in the simplification process. Cumulative cost function is defines as follows:

$$\sigma_K\left(j\right) = \sum_{i=1}^{j} K_{removed}\left(i\right) \qquad \text{(VII.12)}$$

In order to determine the optimal threshold an empirical evaluation has been conducted. By knowing the ground truth, the final simplification limit was set and the threshold for σ_K was calculated at the final stage of the process. The determined value for the experimental dataset involved was σ_{KTh}=1.73.

The interactive setting of the final simplification stage needs a semi-automatic interface. However, in case of the real world application, editor can set the values of the threshold σ_{KTh} on a slider and in that way make a summary more or less detailed.

This refinement can run in real time because of the computational simplicity of the calculations involved in the final stages of simplification.

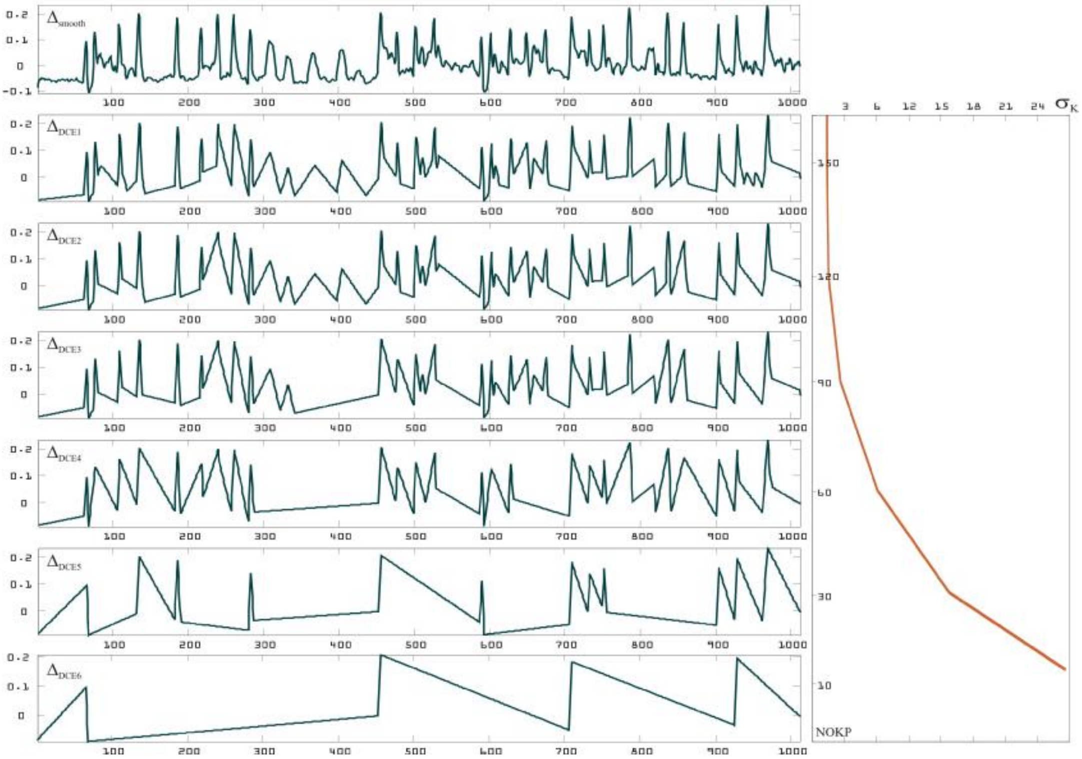


Figure VII.8 Evident increase in the cost function appears in the area when the simplification procedure starts removing important vertices from the metric, sequence ulosci.mpg

VII.3. MOTION FLOW AND CAMERA WORK ANALYSIS

Since the motion flow extracted directly from MPEG motion vectors has rough granularity of one vector per 16x16 pixel region, it was impossible to determine local motion of the objects present in the scene. Therefore, just the global motion characteristics were extracted: camera pan, zoom in/out and rotation, as presented in Chapter IV.

Rough granularity with rather good global motion description of the extracted motion flow is depicted in Figure VII.9. The only unsolved problem is with shots having big homogeneous regions covering the most of the screen, like black breaks, some very dark scenes or just special effects. During that sequence type, a motion estimator has almost a random choice of the best prediction region to minimise the prediction error, and therefore, the motion vectors involved are completely random in both direction and intensity, as depicted in the first frame in Figure VII.9.

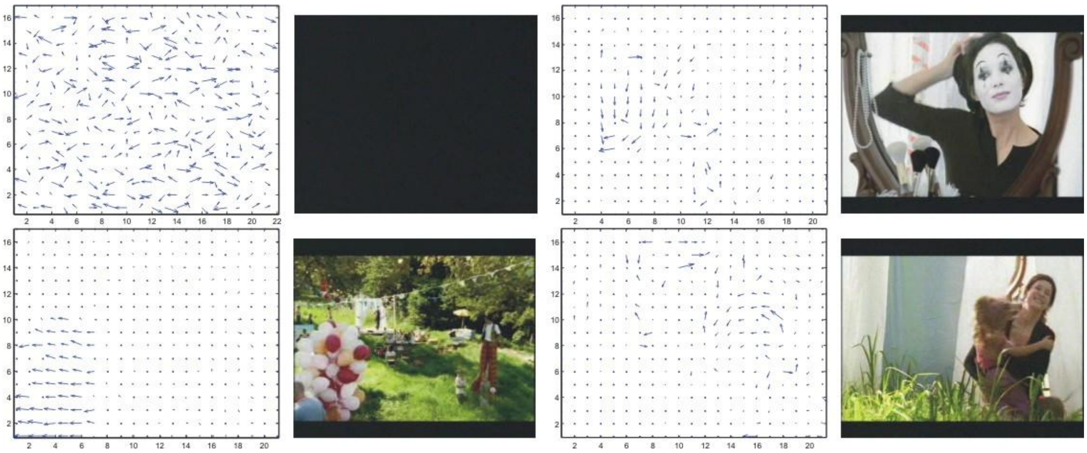


Figure VII.9 Motion Flow extraction

Again, the ground truth for the camera categorisation is labelled manually. It consisted of categories: pan, tilt, zoom in, zoom out, rotation. Results are compared with similar research work that based its camera analysis on MPEG motion vectors [MILAN]. Results are given in Table VII.6 in the form our result/comparison for the categories available. Other publication didn't give any numerical comparison of the achieved camera classification quality.

	Pan	Tilt	Zoom in	Zoom out	Rotation
FA	5.64/8.89	7.21/7.48	5.64/-	7.25/-	21.56/-
MD	6.35/5.75	6.25/0.00	15.02/18.87	12.35/-	26.22/-

Table VII.6 Evaluation of the camera work categorisation

VII.4. KEY-FRAME EXTRACTION

Objective evaluation of how representative is the given set of key-frames is a very difficult task because of the subjective impression one can have about the importance of the particular events to the overall content. After few experiments with different abstraction rate and different video content, the conclusion is that the algorithm shows subjectively excellent results for news and soaps, while the content of the commercials is presented adequately if the major visual changes are transitions and not editing effects. Nevertheless, since the visual summary has to give more key-frames in the case of frequent content change even within one single shot, this algorithm gives a good visual summary of the visual events present in the sequence. The three step metric analysis of a representative part in the sample clip ulosci.mpg is presented in Figure VII.10, where the dotted lines show the positions of the key-frames.

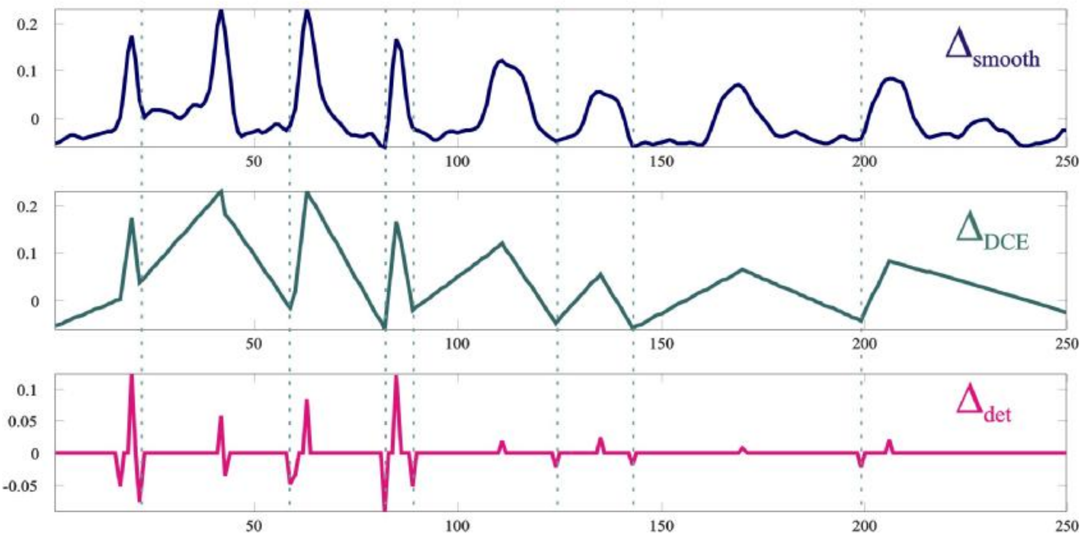


Figure VII.10 Dotted lines show the key-frame positions

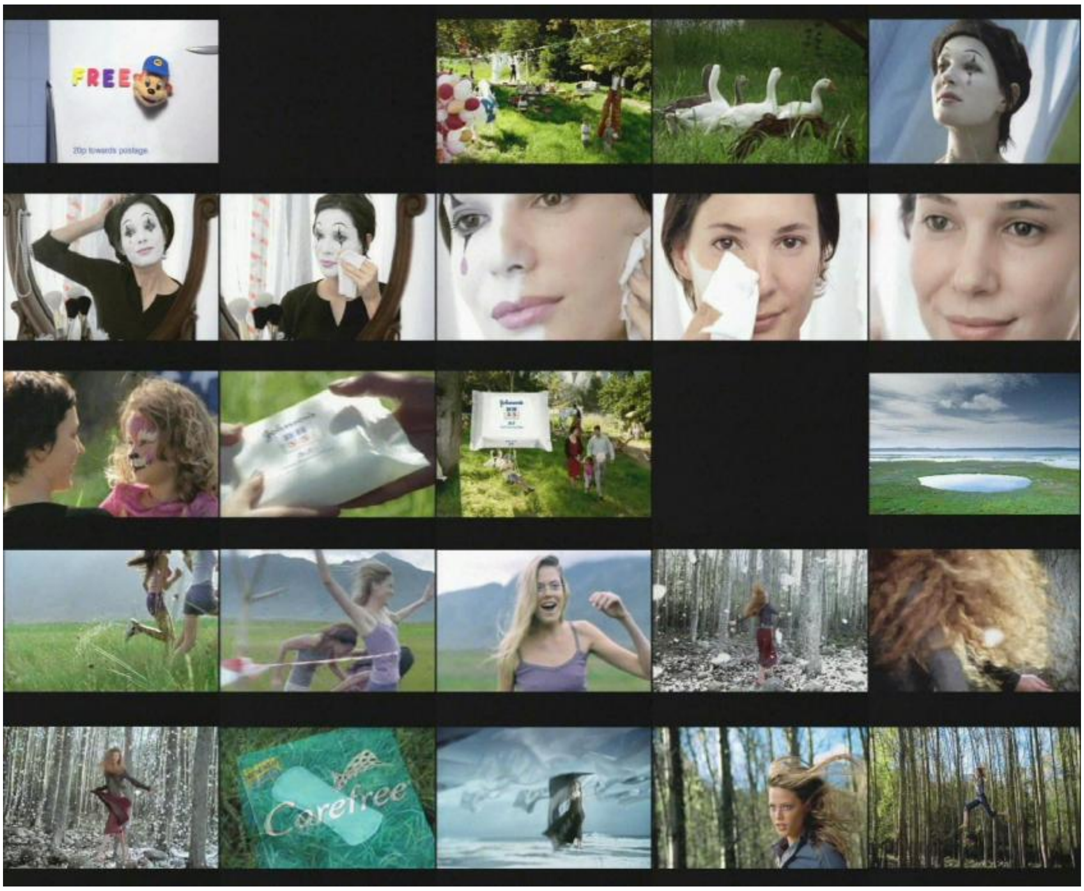


Figure VII.11 Summary of the commercial video clip

An example of the video summary generated from the extracted set of the key-frames is given in Figure VII.11. Shot and scene analysis is easily applied to it: black key frames reveal the breaks between the commercials; scenes could be differentiated by simple colour analysis and local features could be extracted using common computer vision methods, like shape and texture descriptors.

VII.5. HIERARCHICAL COLOUR HISTOGRAM QUANTISATION

Figure VII.12 depicts the hierarchical quantisation of the hue component of the colour histogram, as described in Chapter IV.

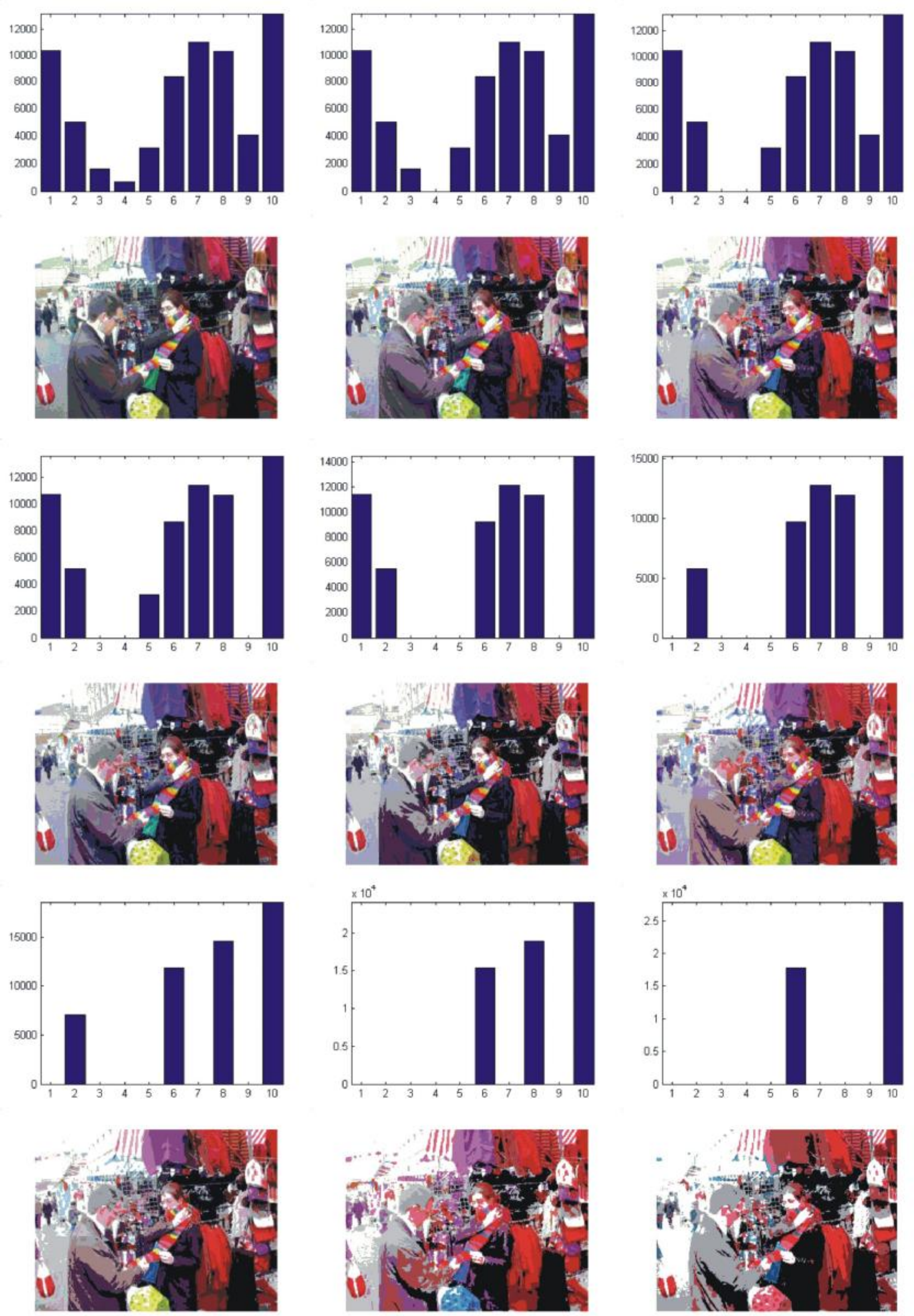


Figure VII.12 Scalable Quantisation of the colour historgam descriptor

Even with only two components left, this histogram quantisation algorithm saves perceptual features needed to maintain the visual similarity. Moreover, the simplification procedure is implemented in the descriptor domain, so that the computational cost is minimised keeping the perceptual control of the process.

Due to its scalability this hierarchical scheme offers highly efficient image and video capabilities. Since this features hasn't been involved in the video representation model, the final results of the retrieval quality are not available.

VII.6. GENRE CLASSIFICATION USING K-MEANS ALGORITHM

Once the set of low-level descriptors have been extracted, the database is clustered by applying a k-means algorithm on the video descriptors. For the 6-dimensional feature vector defined in Chapter V the EMD distance between points in the 6-dimensional space was used.

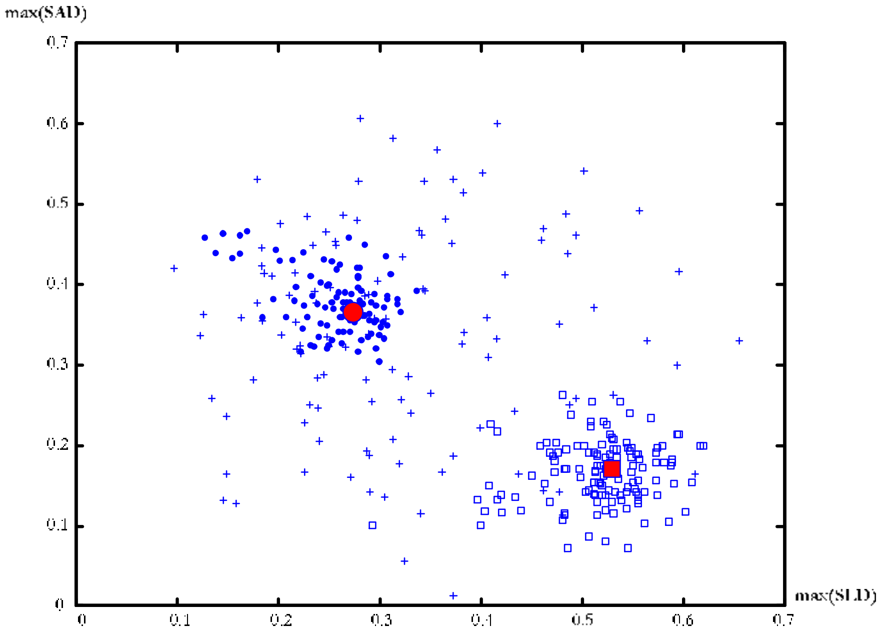


Figure VII.13 Video Filtering into three sub-classes: news, commercials and others.

Figure VII.13 shows the partition of a portion of the experimental dataset into the three sub-classes: news, commercials and soap clips. In this image each item (square, circle or cross) represents the maximum component in 3-bin feature distribution of video clip's SLD and SAD. In this representation squares correspond to news, circles correspond to commercials and the crosses represent other video clips in the database. As seen in the figure, designed representation model separates news and commercials classes efficiently. However, the soap cluster has been dispersed so that the soap class was avoided in the final classification and automatic annotation described in the next section.

VII.7. REPRESENTATION MODEL FOR AUTOMATIC ANNOTATION

In case of the automatic annotation system the representation model consists of the shot activity descriptor as: i) normalised 3-bin distribution of the percentage of the video clips with high, mid and low shot activity and ii) 3-bin SLD descriptor defined in Chapter V. The final video representation model consists of a 6-dimensional vector containing three values for the length distribution (long, mid and short) and three values for the shot activity (high, mid and low).

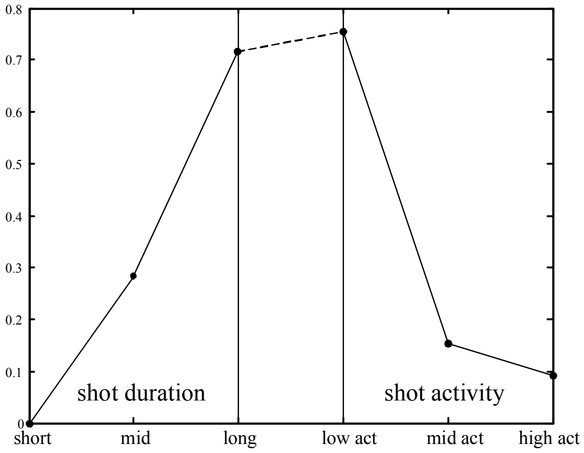


Figure VII.14 Example of the 6-dimensional temporal descriptor of the video clip news025.mpg.

Figure VII.14 shows an example of the 6-dimensional feature vector obtained for a news video clip with only anchorperson present. In this particular example the shots are long and the activity is low.

Universe	Fuzzy set	b_1	b_2	b_3
Shot length distributions	$\tilde{A}$ =short	0.2	0.35	0.5
	$\tilde{B}$ =mid	0.2	0.35	0.5
	$\tilde{C}$ =long	0.1	0.25	0.4
Shot Activity	$\tilde{A}$ =low_act	0.3	0.5	0.7
	$\tilde{B}$ =mid_act	0.1	0.25	0.4
	$\tilde{C}$ =high_act	0.1	0.25	0.4

Table VII.7 Knowledge representation boundaries for the temporal descriptors.

The knowledge representation of the fuzzy system used in the experiments is given in the Table VII.7. Since news and commercial programmes are produced using a rather unique editing technique, the two temporal features described previously appear to be well suited to represent this sort of video clip in the rule mining process.

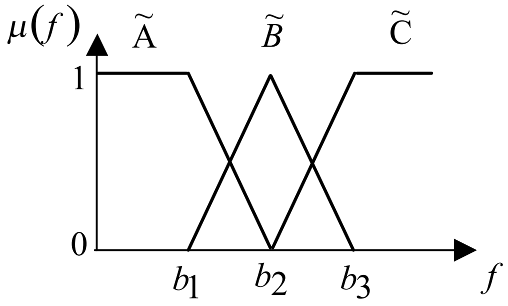


Figure VII.15 Definition of the fuzzy variables for the knowledge representation.

For each feature a set of three fuzzy variables is generated, as shown in Figure VII.15. In this representation the shot length distribution is given by the percentage of shots in the video clip having short, mid and long duration. In addition, representation of the shot activity is given as the percentage of the clip duration having low, mid or high visual activity. The normalised fuzzy boundaries were determined empirically, by "*manually optimizing*" the differences between the three video categories in the feature space.

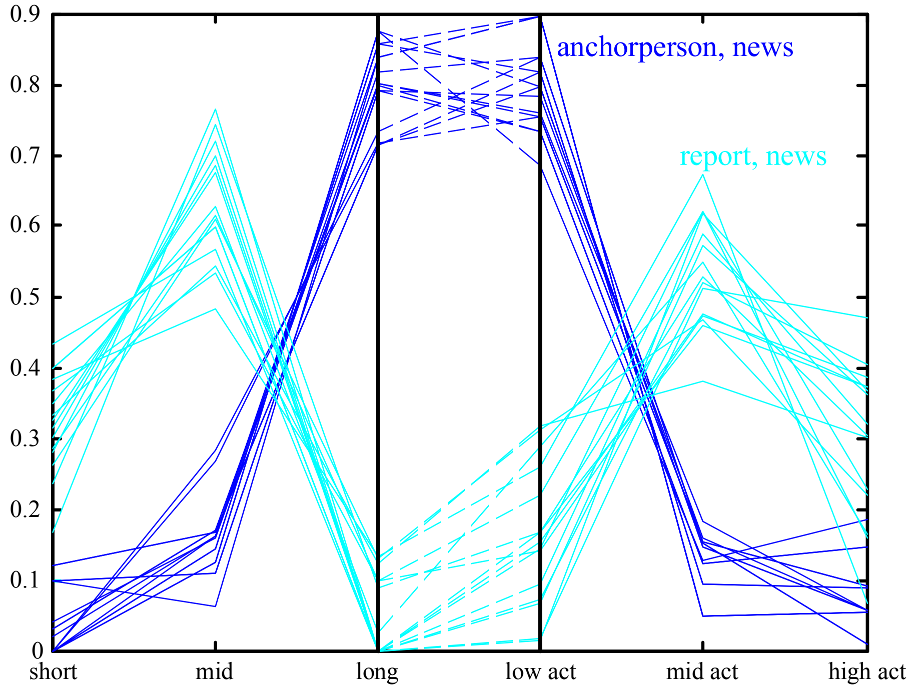


Figure VII.16 Temporal descriptors of the training dataset.

Applying rule mining based on the knowledge representation [DORADO], a rule knowledge base was created. Figure VII.16 shows temporal descriptors of video clips with the corresponding pre-annotated labels, e.g. anchorperson and report for news clips. At the left side of this figure the shot duration is plotted, while at the right side the shot activity is shown.

After fuzzification feature-related words were added to the log of transactions and the rule mining generated the rules given in Table VII.8.

SUPPORT	<CONDITION>	<ACTION>
0.1	SLD_shorts is low AND SLD_longs is high	Labeled as anchorperson
	SLD_longs is high AND SA_low is high	Labeled as anchorperson
	SLD_longs is high AND SA_mid is low	Labeled as anchorperson
0.08	SLD_longs is low AND SA_mid is high	Labeled as report
	SLD_midis is high AND SA_low is low	Labeled as report
	SLD_short is mid AND SA_mid is high	Labeled as report

Table VII.8 Rules generated using representation given above with support values 1.0 and 0.08

These rules appear to be rather intuitive resembling human reasoning and knowledge: clips containing anchorperson have many long shots with low shot activity, while report clips are characterised by middle length shots having medium visual activity. This shows the value of the representation model that enabled the meaningful automatic annotation of the videos based on a learning dataset.

VII.8. AUTOMATIC ANNOTATION EVALUATION

Using the rule-knowledge base given in Table VII.8, a set of 80 video clips was annotated both automatically and manually. The objective of the manual annotation was to generate ground truths for the evaluation of the automatic annotation process. An example of the particular feature values extracted from three random video sequences belonging to the "news" cluster is given in Table VII.9. Next to each feature value, fuzzy membership values are given as A, B and C. At the bottom of the table the membership values m_f^k for the keywords *k* from the lexicon are given. These values are obtained as the output of the fuzzy inference module of the annotation unit as described in [DORADO]. In order to assess the accuracy of the annotation procedure, a statistical performance evaluation based on the amount of missed detections (MD's) and false alarms (FA's) for each keyword from the lexicon was conducted. The values for quality of the annotations are defined as recall and precision, just as in the shot detection evaluation:

$$\text{Recall} = \frac{D}{D+MD}, \text{Precision} = \frac{D}{D+FA} \qquad \text{(VII.13)}$$

where the *D* is the sum of memberships m_f for the corresponding keyword *k*, *MD* is the sum of the distances to the full true membership $m_f=1$ and *FA* is a sum of false memberships:

$$D = \sum_{i=1}^{M} m_f^k(i), \quad k: \text{true}, MD = \sum_{i=1}^{M} 1 - m_f^k(i), \quad k: \text{true}, FA = \sum_{i=1}^{M} m_f^k(i), \quad k: \text{false} \qquad \text{(VII.14)}$$

	NEWS032	A	B	C	NEWS081	A	B	C	NEWS138	A	B	C
sld_shorts	0.000	1.000	0.000	0.000	0.200	1.000	0.000	0.000	0.333	0.111	0.889	0.000
sld_mids	0.382	0.000	0.788	0.212	0.143	1.000	0.000	0.000	0.544	0.000	0.000	1.000
sld_longs	0.618	0.000	0.000	1.000	0.657	0.000	0.000	1.000	0.122	0.852	0.148	0.000
sa_low	0.655	0.000	0.225	0.775	0.418	0.410	0.590	0.000	0.317	0.916	0.085	0.000
sa_mid	0.153	0.645	0.355	0.000	0.292	0.000	0.718	0.282	0.380	0.000	0.131	0.869
sa_high	0.191	0.390	0.610	0.000	0.289	0.000	0.738	0.262	0.302	0.000	0.652	0.348
ground truth	Anchorperson		mf		Anchorperson		mf		Report		mf	
fuzzy inferences	Anchorperson		1.000		Anchorperson		1.000		Anchorperson		0.000	
	Report		0.000		Report		0.212		Report		0.916	

Table VII.9 Temporal representation of three randomly selected news clips.

The obtained recall and precision for three representative clips are given in the Table VII.10.

KEYWORDS	DETECTS	MD	FA	RECALL	PRECISION
Anchorperson	46.65	1.36	2.48	0.97	0.95
Report	25.74	6.28	1.32	0.80	0.95

Table VII.10 Recall and Precision results of the annotation process.

VII.9. SYSTEM DEMONSTRATORS

During the three year PhD project the system has developed into several demonstrators. They were presented at IPOT 2001 and IPOT 2002 exhibitions, EPSRC and DTI seminars, various internal seminars at QMUL, etc. They presented the functionalities of the presented system at different stages. For real time demos and publications please visit http://www2.elec.qmul.ac.uk/~janko .

The layout images of the demos are given in three following figures Figure VII.17, Figure VII.18, Figure VII.19:

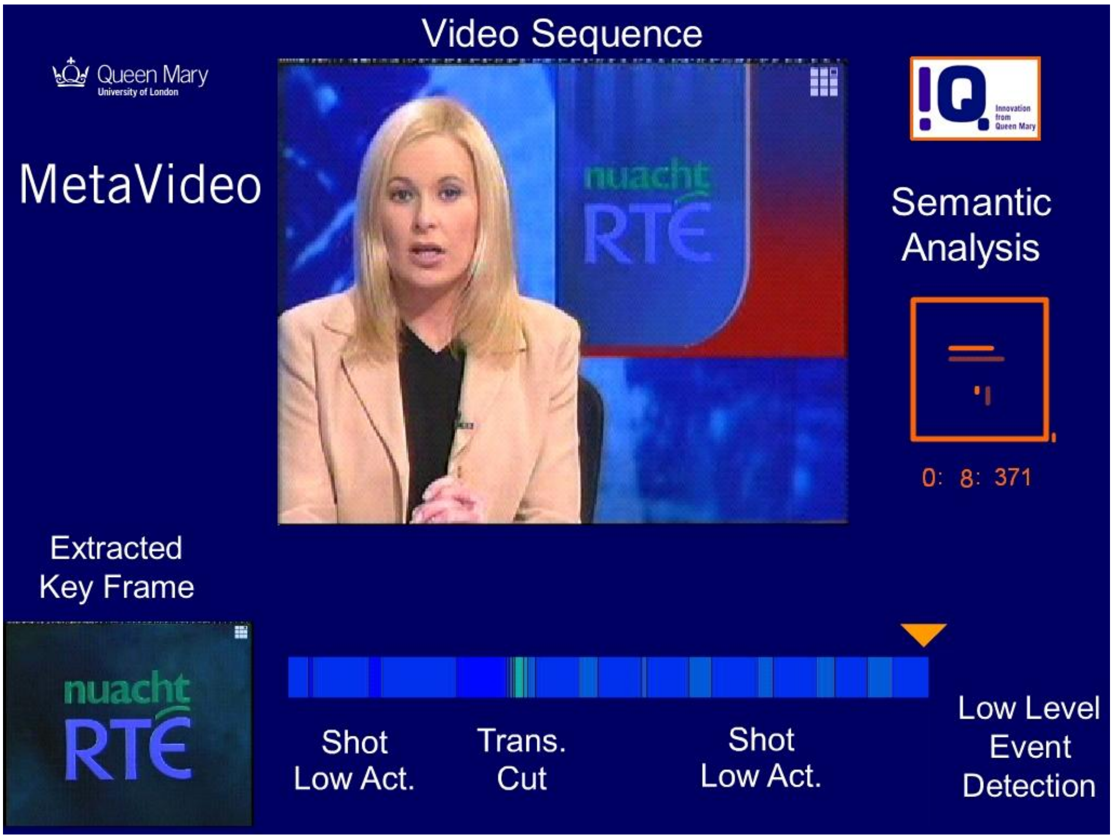
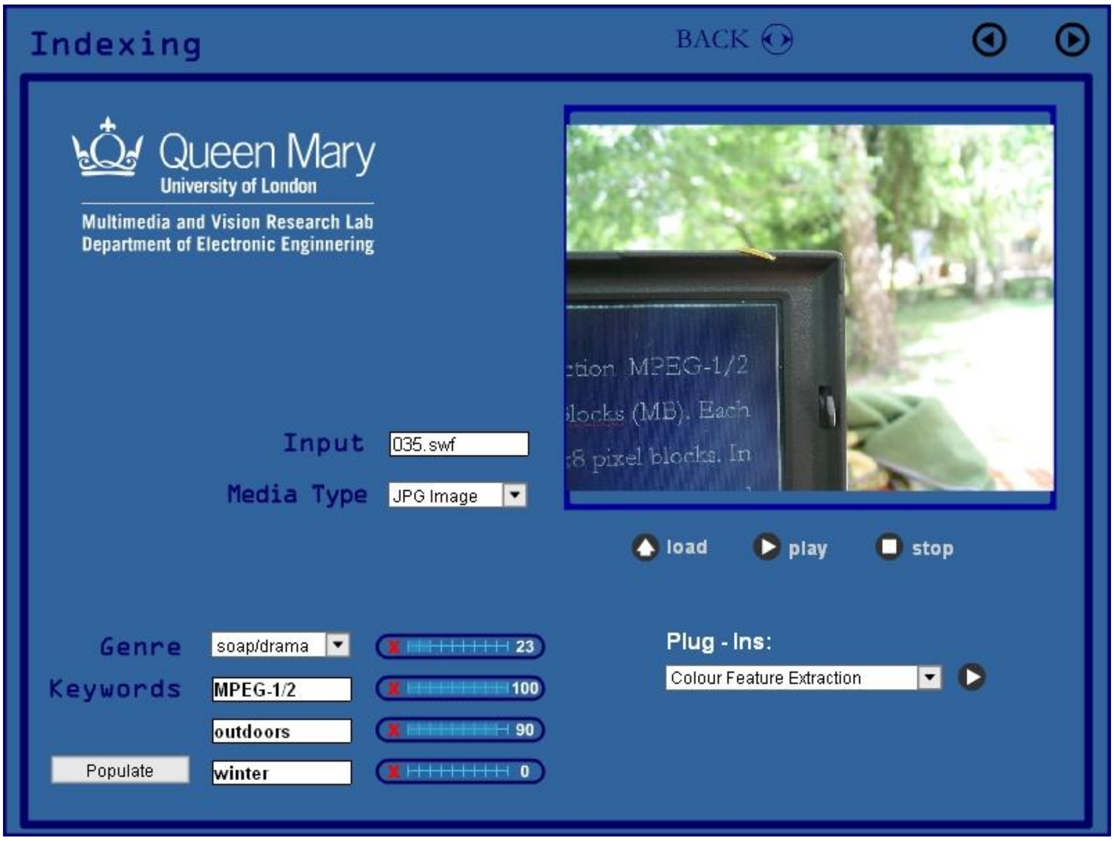


Figure VII.17 IPOT 2001 Demo


Indexing
BACK
Queen Mary University of London
Multimedia and Vision Research Lab
Department of Electronic Enginnering
load
play
stop
Populate

Input: 035.swf

Media Type: JPG Image

Genre: soap/drama 23

Keywords: MPEG-1/2 100

outdoors 90

winter 0

Plug - Ins: Colour Feature Extraction

Figure VII.18 DTI 2002 Demo

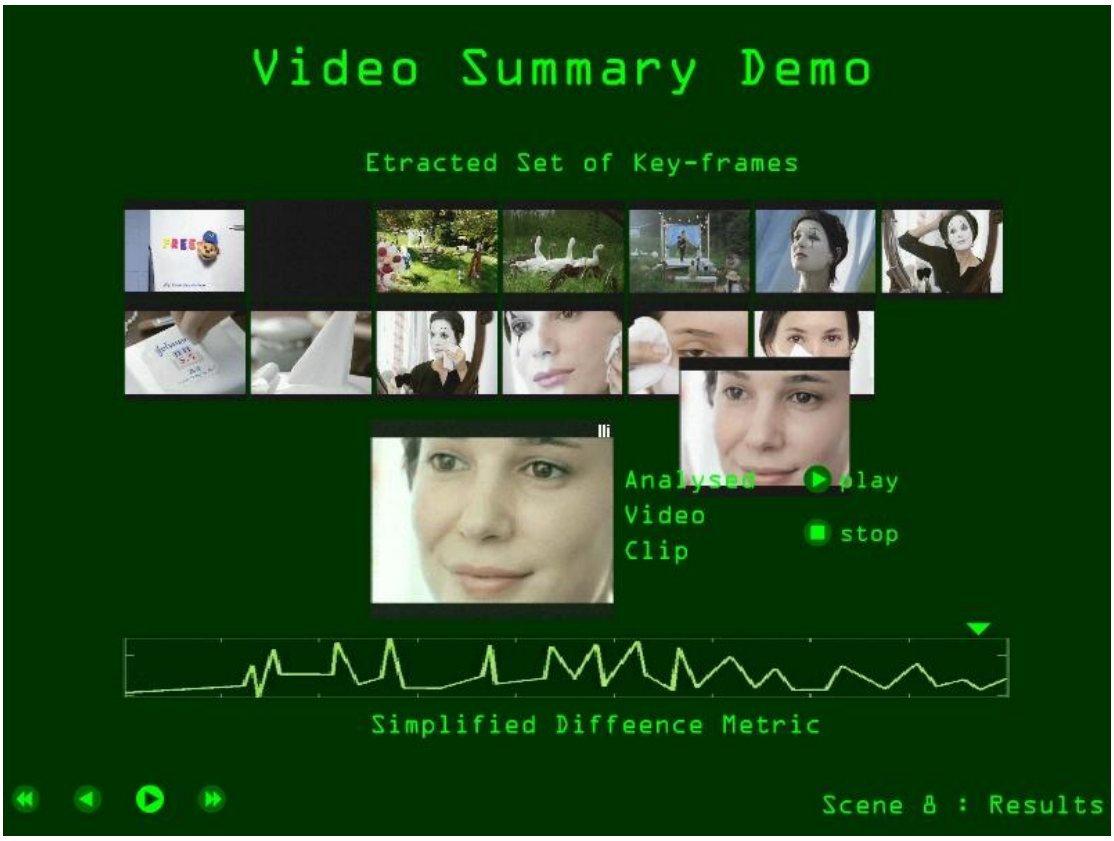


Figure VII.19 MPhil transfer Demo

VII.10. SUMMARY

This chapter brings the final results achieved during this research. It stats with evaluation of the temporal parsing, giving the details of both successful and less successful algorithms implemented. Followed by mid-level descriptors like motion flow and camera work, it presents the results of the exploitation of the video model in the automatic annotation system. The chapter concludes with the shortlist of the system demonstrators developed during the project.

VIII. DISCUSSION AND CONCLUSIONS

VIII.1. DISCUSSION

Recent development of highly efficient video compression technology combined with the rapid increase in desktop computer performance, and a decrease in the storage cost, have led to a proliferation of digital video media. Therefore, the crucial problem in the field of multimedia indexing and retrieval nowadays is intuitive handling of that vast data stored in a multimedia database.

Research presented here has focused on the problems of the content based video indexing and retrieval, targeting both robustness and efficiency of implemented algorithms on one hand, and the semantic capability of the generated video representations on the other. Wide spectrum of research activities fell within the scope of the work presented: temporal video analysis, shot boundary detection, key frame extraction, colour feature extraction and quantisation, video representation design and genre classification.

The main objectives of this research project were to achieve real time processing capabilities of the temporal analysis algorithms in order to enable efficient and reliable representation for later high level semantic analysis in a CBVIR system. This goal was achieved by utilising easily accessed information from MPEG-1/2 compressed domain and other compliant compressing standards like the H.26X. The key advantage of the video analysis in the compressed domain is in its inherent efficiency and robustness. Particularly in our case, the prediction information extracted in the motion estimation part of the temporal prediction process is easily extracted directly from the MPEG video stream. Behind this information stands intense pre-processing in the encoding stage that tries to minimise temporal redundancy present in the sequence of frames by predicting motion compensated pixel values. From this prediction's behaviour one can derive a frame similarity metric, essential for the process of temporal analysis.

The MPEG compressed domain information exploited in our work is type of prediction on a macro block level, i.e. MB type. Various other MPEG variables were tested for this purpose. Motion Vectors and DC sequence coefficients were evaluated but being unreliable and complex to extract while achieving similar or even inferior results, these features were discarded in further study.

In number of development stages, a one dimensional frame difference metric is generated from a straightforward statistics of MB type distribution. The major obstacle in the metric development was its continuity through neighbouring sub-groups of pictures (SGOP). This problem is solved by tracking the information on intra and interpolated prediction within a SGOP, in addition to the forward and backward prediction. Moreover, a Gaussian smoothing is applied to minimise the noise present in the final metric.

The second objective to efficiency was algorithm scalability. This requirement is essential for adaptive behaviour needed for the high-level semantic analysis. Adopting a geometric procedure called discrete contour evolution and adapting it to this particular application, a family of curves describing the temporal features of the video sequence is generated. Experiments conducted on various types of video clips showed that the simplification algorithm gradually removes noise and unimportant events from the metric while saving salient features. This scale space of frame difference metric curves can be used in various applications, especially knowing the fact that it is created in real-time: live editing, summarising, video surveillance, etc. Here, it is exploited for key-frame extraction, localising the most representative frames in a shot. Again, there is no expensive computation involved, so that the efficiency of the algorithm is maintained even in the key-frame extraction module.

Following a similar simplification technique, a HSV colour histogram hierarchical quantisation is developed. Unlike the most of colour quantisation methods, this method simplifies a colour histogram in the descriptor domain. Consequently, the simplification procedure is very efficient. Yet, the simplification procedure is driven by perceptual degradation of colours in the image, so that the basic criterion behind the simplification process is not the degradation of histogram itself.

Having a set of low-level descriptors, a video representation model is designed. The main guidelines in the design process were representation semantic quality while maintaining low complexity of the final model. Motivated by the Computational Media Aesthetics paradigm, representation model is generated following the rules and knowledge of the video editing and theory. Shot pace and the overall activity present in the scene are the backbone of the perceptual model developed. Shot pace is calculated on the fly with shot boundary detection, while the shot activity is extracted from the frame difference metric information. In addition, an adaptive capability is embedded in the model generation algorithm utilising scalable model behaviour.

In order to support contextual issues in the semantic retrieval, a genre classification algorithm is created. It is founded upon editing rules characteristic for a particular type of programme, e.g. news, commercials, soaps, etc. A video dataset is clustered into video categories using k-means algorithm, due to the fact that the number of clusters and at least on member of cluster is known from the learning dataset. Experimental results have shown that the automatic annotation system, being supported by contextual information gain from the genre classification module, achieves excellent results in unsupervised linking between video clip representations and a predefined keyword lexicon. This system is still evolving with the prospects to become an autonomous self-learning video indexing and retrieval engine, a system that has a wide application horizons and many research opportunities. Demonstrators built throughout this research project attracted a lot of attention of people from the media production business on various multimedia exhibitions and seminars in UK and worldwide.

VIII.2. FUTURE WORK

The need for further developments in the CBVIR area is obvious. The presented work brings up the problem of appropriate representations of video clips in current database. Whether the choice of one key-frame can bear the information load present in a shot; and if it can, are the current methods appropriate? Future research activities will try to answer this and other similar questions, critical for the progress of CBVIR towards semantic capabilities.

Retaining requirements for the algorithm efficiency and robustness, our future work will look into the compressed domain analysis of the spatial information for intelligent key-frame extraction. By rough unsupervised region segmentation and camera motion characterisation, spatial relationships of regions and camera work analysis could improve the choice of the optimal representative frame sub-set, ranging from one key-frame to multiple frame or panoramic shot representations.

Another direction of the future research will be towards adaptive video representation models based on the production knowledge and high-level semantic information, as well as interaction with user. Without an intelligent interaction and adaptation of the system to the contextual circumstances and user preferences, semantic retrieval will stay only a science fiction topic. Deeper involvement of the user in the process of video analysis, even on the lowest levels, is essential to the CBVIR progress.

Therefore, utilisation of user relevance feedback information through various interfaces and creation of appropriate adaptable representations are the major challenges of the current multimedia database management development.

VIII.3. CONCLUSIONS

The starting objectives of the project towards efficient low-level feature extraction for video indexing and retrieval were achieved entirely. Experimental results show high efficiency and robustness of the temporal analysis algorithm and its scalability to various applications. The main contribution of the temporal analysis is in its entirely compressed-domain based analysis and the specific transformation from a complex video stream to one dimensional metric describing activity of visual change present in the analysed sequence. In addition, a novel key-frame extraction algorithm is developed.

In the domain of video representation, a novel perspective is given. It follows the approach to video analysis adopted by filmmakers and film theoreticians, with the intention of generating more intuitive representations of videos in modern multimedia databases. A scalable video representation model designed to emulate the media editing rules, as another major contribution of this research, narrows the "semantic gap" by lifting low-level descriptors to a completely new level in a semantic signification chain. The representation model allows contextual classification of clips into genres and achieves high punctuality in the automatic video annotation.

However, ever demanding field of content based retrieval needs further developments. Therefore, our future work will be focused on further exploration of intelligent content based retrieval and bringing the role of the user to a new level in a semantic retrieval process.

By following these guidelines, the future of more intuitive handling of media is bright. If not fully understandable to future computers, digital media will be much more accessible by users, because the content of the motion pictures will be much closer to the machine, opening the whole new horizons of creativity and accessibility to humans. Whether the computer will ever be able to understand the smiles and tears of actors from Zuse's punched film tapes will stay a question for future generations.

IX. REFERENCES

[ADAMS1] B. Adams, C. Dorai, and S. Venkatesh, "Automated film rhythm extraction for scene analysis", In IEEE International Conference on Multimedia and Expo, Tokyo, Japan, August 2001.

[ADAMS2] B. Adams, C. Dorai, and S. Venkatesh, "Role of shot length in characterizing tempo and dramatic story sections in motion pictures", In IEEE Pacific Rim Conference on Multimedia 2000, pp. 54-57, Sydney, Australia, December 2000.

[ADAMS3] B. Adams, C. Dorai, and S. Venkatesh, "Study of shot length and motion as contributing factors to movie tempo", In 8th ACM International Conference on Multimedia, pages 353-355, Los Angeles, California, November 2000.

[ADAMS4] B. Adams, C. Dorai, and S. Venkatesh, "Towards automatic extraction of expressive elements from motion pictures: Tempo", In IEEE International Conference on Multimedia and Expo, volume II, pages 641-645, New York City, USA, July 2000.

[AHMED] N. Ahmed, T.Natrajan and K.R.Rao, "Discrete Cosine Transform", IEEE Transactions on Computers, Vol. C-23, No.1, pp. 90-93, December 1984.

[AIGRAIN] P. Aigrain, P. Joly, "The automatic real-time analysis of film editing and transition effects and its applications", Computers and Graphics 18(1) (1994) 93-103.

[AKUTSU] A. Akutsu, Y. Tonomura, "Video tomography: An efficient method for camerawork extraction and motion analysis", ACM Multimedia 94, 1994, 349-356.

[AKUTSU1] A. Akutsu, Y. Tonomura, H. Hashimoto, Y. Ohba, "Video indexing using motion vectors", Proc. SPIE: Visual Commun. Image Process. '92 1818, 1992, 1522-1530.

[ARIJON] D. Arijon. Grammar of the film language. Silman-James Press, 1976.

[ARMAN] F. Arman, A. Hsu, M-Y Chiu, "Image processing on compressed data for large video databases", in: Proc. First ACM Intern. Conference on Multimedia, 1993, pp. 267-272.

[ARMAN1] F. Arman, R. Depommier, A. Hsu, M.-Y. Chiu, "Content-Based Browsing of Video Sequences", ACM Multimedia 1994: 97-103

[AYER] S. Ayer and H. S. Sawhney, "Layered Representation of Motion Video using Robust Maximum-Likelihood Estimation of Mixture Models and MDL Encoding", Technical report, IBM Almaden Research Center, San Jose, CA 95120, December 1994.

[BACH] J. R. Bach et al., "Virage Image Search Engine: An Open Framework for Image Management", Proceeding of Conference on Storage and Retrieval for Image and Video Databases IV (IS&T/SPIE-1996), San Jose, California, 1996.

[BESCOS] Bescos J., Martinez J.M., Cabrera J., Menendez J.M., Cisneros G., "Gradual shot transition detection based on multidimensional clustering", 4th IEEE Southwest Symposium on Image Analysis and Interpretation. IEEE Comput. Soc. 2000, pp.53-7. Los Alamitos, CA, USA

[BORDW] D. Bordwell and K. Thompson. Film Art, 5th Ed. McGraw-Hill, 1997.

[BOREC] J. Boreczky, L.A. Rowe, "Comparison of video shot boundary detection techniques", in: Proc. IS&T/SPIE Intern. Symposium Electronic Imaging, San Jose, 1996.

[BOREC1] J. Boreczky, L.D. Wilcox, "A hidden Markov model framework for video segmentation using audio and image features", in: Proc. Int. Conf. Acoustics, Speech, and Signal Proc., 6, Seattle, 1998, pp. 3741-3744.

[BOUTH] P. Bouthemy and R. Fablet, "Motion Characterization from Temporal Co-occurences of Local Motion-based Measures for Video Indexing," 14th Int. Conf. on Pattern Recognition, ICPR '98, Brisbane, 1998.

[BRUNET] P. Brunette, D. Wills "Screen/play. Derrida and film theory", Princeton Univ. Press, Princeton, NJ, USA, 1989

[BUCKLEY] C. Buckley, G. Salton, Optimization of Relevance Feedback Weights, Proc. 18 Annual Intl ACM SIGIR Conf., Seattle, USA, 1995, pp. 351-357.

[CALIC] J. Calic and E. Izquierdo, "A Multiresolution Technique For Video Indexing And Retrieval", Proceedings of the IEEE International Conference on Image Processing 2002, Rochester, NY, USA, September 2002.

[CALIC] J. Calic and E. Izquierdo, "Deconstructing Bridges", Submitted to IEEE Multimedia, Special issue: Computational Media Aesthetics - Bridging the Semantic Gap, April 2003.

[CALIC] J. Calic and E. Izquierdo, "Efficient Key-Frame Extraction and Video Analysis", Proceedings of the IEEE International Conference on Information Technology ITCC 2002, Las Vegas, NV, USA, April 2002.

[CALIC] J. Calic and E. Izquierdo, "Temporal Segmentation of MPEG video streams", Special issue on Image Analysis for Multimedia Interactive Services, EURASIP Journal on Applied Signal Processing, June 2002

[CALIC] J. Calic and E. Izquierdo, "Temporal Video Segmentation for Real-Time Key Frame Extraction", Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Orlando, FL, USA, May 2002.

[CALIC] J. Calic and E. Izquierdo, "Towards Real-Time Shot Detection in the MPEG Compressed Domain", Proceedings of the Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS'2001, Tampere, Finland, May 2001.

[CASTELLI] V. Castelli, L.D. Bergman Ed., "Image Databases-Search and Retrieval of Digital Imagery", John Wiley and Sons, New York, 2002.

[CHANG] S. Chang, W. Chen, H. J. Meng, H. Sundaram, and D. Zhong, VideoQ: "An automated content based video search system using visual cues", in Proceedings of the Fifth ACM Multimedia Conference, Seattle, November 1997.

[CONKLIN] J. Conklin, "HyperText: An introduction and survey," IEEE Comput. Mag., pp. 17-41, Sept. 1987.

[COX] I.J. Cox, M.L.Miller, T. P. Minka, T. Papathomas and P. N. Yianilos, "The Bayesian Image Retrieval System, PicHunter: Theory, Implementation and Psychophysical Experiments", IEEE Trans on Image Processing, Vol. 9, p.p. 20-37, Jan 2000.

[DAVIS] M. Davis, "Knowledge Representation for Video", Proc. Of 12th National Conference on Artificial Intelligence (AAAI-94), Seattle, USA, AAAI Press, pp. 120-127, 1994.

[DAVIS1] M. Davis, "Media Streams: An Iconic Visual Language for Video Representation", Readings in Human-Computer Interaction: Toward the Year 2000, ed. Ronald M. Baecker, Jonathan Grudin, William A. S. Buxton, and Saul Greenberg. 854-866. 2nd ed., San Francisco: Morgan Kaufmann Publishers, Inc., 1995.

[DAWOO] Dawood, A.M., Ghanbari, M. ,"Clear scene cut detection directly from MPEG bit streams", Image Processing And Its Applications, 1999. Seventh International Conference on, Volume: 1 , 13-15 July 1999, p.p. 285 – 289, IEE, London, UK

[DEGALL] D. LeGall, J.L. Mitchell, W.B. Pennbaker, Fogg C.E., "MPEG video compression standard", Chapman & Hall, New York, USA, 1996.

[DELBIM] A. Del Bimbo, "Expressive Semantics for Automatic Annotation and Retrieval of Video Streams", Proceedings of International Conference On Multimedia and Expo (ICME-2000), New York, NY, July 2000.

[DERRIDA] J. Derrida, "Of Grammatology", Johns Hopkins University Press, Baltimore and London, USA, 1976

[DIMIT] N. Dimitrova and F. Golshani, "Motion recovery for video content classification," ACM Trans. Inform. Syst., vol. 13, no. 4, pp. 408-439, Oct. 1995.

[DOBIE] M. Dobie and P. H. Lewis, "Object tracking in multimedia systems," in Proc. 4th Int. Conf. Image Processing Applications, The Netherlands, Apr. 1992, pp. 41-44.

[DONGGE] L. Dongge, I.K.Sethi, "MDC: a software tool for developing MPEG applications", Proceedings IEEE International Conference on Multimedia Computing and Systems. IEEE Comput. Soc. Part vol.1, 1999, pp.445-50 vol.1. Los Alamitos, CA, USA.

[DORADO] A. Dorado, J. Calic, E. Izquierdo, "A Rule-Based Video Annotation System", IEEE Transactions on Circuits and Systems in Video Technology, to be published April 2004

[DUGAD] Dugad R, Ratakonda K, Ahuja N, "Robust video shot change detection", Proc. of 1998 IEEE Second Workshop on Multimedia Signal Processing (Cat. No.98EX175). IEEE. 1998, pp.376-81. Piscataway, NJ, USA.

[FABLE] R. Fablet and P. Bouthemy, "Motion-Based Feature Extraction and Ascendant Hierarchical Classification for Video Indexing and Retrieval," 3rd Int. Conf. on visual Information Systems, VISual'99, Amsterdam, 1999.

[FERMAN] A. Ferman, A. Tekalp, "Efficient filtering and clustering for temporal video segmentation and visual summarization", Journal of Visual Communication and Image Representation 9(4) (1998) 3368-351.

[FERNA] Fernando, W.A.C., Canagarajah, C.N., Bull, D.R., "A unified approach to scene change detection in uncompressed and compressed video", Consumer Electronics, IEEE Transactions on , Volume: 46 , Issue: 3, p.p. 769 – 779, Aug. 2000.

[FISCHER] S. Fischer, R. Lienhart, and W. Effelsberg, "Automatic Recognition of Film Genres," in The 3rd ACM Int. Multimedia Conference and Exhibition, 1995.

[FLICK] M. Flickner et al., "Query by Image and Video Content: The QBIC System", Computer, Vol. 28, No. 9, pp. 23-32, Sep. 1995.

[FORGY] Forgy, E. 1965. Cluster analysis of multivariate data: Efficiency versus interpretability of classification. Biometrics, 21, 768-780.

[GERSHO] A. Gersho and R. M. Gray, "Vector quantization and signal compression", Kluwer international series in engineering and computer science. Kluwer Academic Publishers, 1992.

[GHARGI] U. Gargi, R. Kasturi, S. Antani, "Performance characterization and comparison of video indexing algorithms", in: Proc. Conf. Computer Vision and Pattern Recognition (CVPR), 1998.

[GHARGI1] U. Gargi, S. Strayer, "Performance Characterisation of Video-Shot-Change Detection Methods", IEEE Trans. on Circuits and Systems for Video Technology, Vol.10, No.1, February 2000

[GONG] Y. Gong, "Intelligent Image Databases Towards Advanced Image Retrieval", Kluwer Academic Publishers, 1998.

[GONZA1] R. Gonzalez C., R. E. Woods, "Digital Image Processing" Addison-Wesley, 1992.

[GONZA2] R. Gonzalez, "Hypermedia data modelling, coding, and semiotics" [Journal Paper] Proceedings of the IEEE, vol.85, no.7, July 1997, pp.1111-40. Publisher: IEEE, USA.

[GU] J. Gu and E. J. Neuhold, "A data model for multimedia information retrieval," in Proc. 1st Int. Conf. Multimedia Modeling, Singapore, Nov. 9-12, 1993, pp. 113-127.

[HAERING] N.C. Haering, R.J. Qian, and M.I. Sezan, "A Semantic Event Detection Approach and Its Application to Detecting Hunts in Wildlife Video," IEEE Trans. on Circuits and Systems for Video Technology, 1999.

[HALHED] B. Halhed, “Videoconferencing Codecs: Navigating the MAZE”, Business Communication Review, Vol. 21, No. 1, pp. 35-40, 1991.

[HAMPA] A. Hampapur, R. Jain, T. E. Weymouth, “Production model based digital video segmentation”, Multimedia Tools and Applications 1(1) (1995) 9-46.

[HANJA] A. Hanjalic and R.L. Langendijk, “A New Key-Frame Allocation Method for Representing Stored Video Streams”, Proc. of 1st Int. Workshop on Image Databases and Multimedia Search, 1996.

[HARDT] O. Hardt, “PSYC 325: COGNITIVE PSYCHOLOGY VISUAL PERCEPTION II”, Lecture Notes in Cognitive Psychology, Psychology Dept., The University of Arizona, USA, June 2003, http://www.u.arizona.edu/~hardt/

[HARTI1] Hartigan, J. 1975. Clustering Algorithms. John Wiley & Sons, New York, NY.

[HARTI2] Hartigan, J. and Wong, M. 1979. Algorithm AS136: A k-means clustering algorithm. Applied Statistics, 28, 100-108.

[HIRZAL] N. Hirzalla and A. Karmouch, “Detecting cuts by understanding camera operation for video indexing”, J. Visual Lang. Comput. 6, 1995, 385-404.

[HITCH] F. L. Hitchcock. The distribution of a product from several sources to numerous localities. J. Math. Phys., 20:224-230, 1941.

[HOFFM] D. Hoffman, “Visual Intelligence: How we create what we see”, New York, W.W. Norton & Co., 1998.

[HUANG] J. Huang, Z. Liu, and Y. Wang, “Integration of audio and visual information for contend-based video segmentation”, In Proc. IEEE International Conference on Image Processing. IEEE, 1998.

[IDE] I. Ide, R. Hamada, H. Tanaka, and S. Sakai, "News Video Classification based on Semantic Attributes of Captions," in Proc. 6th ACM International Conference, 1998.

[INFOME] http://www.infomedia.lu

[ISO1] ISO/IEC 11172-2:1993, “Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s -- Part 2: Video”, ISO/IEC JTC 1

[ISO2] ISO/IEC 13818-2:2000, “Information technology - Generic coding of moving pictures and associated audio information: Video”, ISO/IEC JTC 1

[JANSCH] R. S. Jasinschi, J. Louie, “Automatic TV Program Genre Classification Based on Audio Patterns”, EUROMICRO 2001: 370-375

[JOYCE] D. W. Joyce, P. H. Lewis, R. H. Tansley, M. R. Dobie, and W. Hall, “Semiotics and Agents for Integrating and Navigating through Multimedia Representations of Concepts”, Proceedings of the Conference on Storage and Retrieval for Media Databases (IS&T/SPIE-2000), pp. 120-131, San Jose, California, Jan. 2000.

[KASTURI] R. Kasturi, R. Jain, “Dynamic vision, in Computer Vision: Principles”, R. Kasturi and R. Jain, (eds.), pp. 469-480, IEEE Computer Society Press, Washington DC, 1991.

[KIKUK] T. Kikukawa, S. Kawafuchi, “Development of an automatic summary editing system for the audio-visual resources”, Transactions on Electronics and Information J75-A (1992) 204-212.

[KOBLA] V. Kobla, D.S. Doermann, Lin K. -I., Faloutsos C., “Compressed domain video indexing techniques using DCT and motion vector information in MPEG video”, Proceedings of SPIE V, Volume 3022, pp. 200-211, February 1997.

[KOPRI] I. Koprinska, S. Carrato, "Detecting and classifying video shot boundaries in MPEG compressed sequences", in: Proc. IX Eur. Sig. Proc. Conf.(EUSIPCO), Rhodes, 1998, pp. 1729-1732.

[KULESH] R. Levaco ed., "Kuleshov on Film: Writings by Lev Kuleshov", Univ. of California Press, Berkeley, USA, 1974

[LATEC] L. Latecki, R. Lakimper, "Convexity rule for shape decomposition based on discrete contour evolution" Computer Vision & Image Understanding, vol.73, no.3, March 1999, pp.441-54. Academic Press, USA

[LEEIP] C. Lee, D. Ip, "A robust approach for camera break detection in color video sequences", in: Proc. IAPR Workshop Machine Vision Appl., Kawasaki, Japan, 1994, pp. 502-505.

[LEE] S. Lee, Young-Min Kim, Sung Woo Choi, "Fast scene change detection using direct feature extraction from MPEG compressed videos", IEEE Transactions on Multimedia, vol.2, no.4, Dec. 2000, pp.240-54. Publisher: IEEE, USA

[LINDLEY] C. Lindley. A computational semiotic framework for interactive cinematic virtual worlds. In Workshop on Computational Semiotics for New Media, Guildford, Surrey, UK, 2000.

[MANOV] L. Manovich, "Metamediji", Centre for Contemporary Arts, 2001., Belgrade, Serbia and Montenegro, (In Serbian)

[MANOV2] L. Manovich, "The Language of New Media", The MIT Press, Boston, USA, 2001.

[MATLIN] M. Matlin and H. J. Foley, "Sensation and Perception". Needham Heights, MA: Simon & Shuster, Inc, 1991.

[MCCLE] McClelland, J. L. and Rogers, T. T. (2003). The Parallel Distributed Processing Approach to Semantic Cognition. Nature Reviews Neuroscience, Vo. 4, April 2003,

[MENG] J. Meng, Y. Juan, S.-F. Chang, Scene change detection in a MPEG compressed video sequence, in: Proc. IS&T/SPIE Int. Symp. Electronic Imaging 2417, San Jose, 1995, pp. 14-25.

[METZ] C. Metz, "Film Language: A Semiotics of the Cinema", Translated by Michael Taylor. xviii, 268 p. 1974, The University of Chicago Press, Chicago, USA

[MILAN] R. Milanese, F. Deguillaume, A. Jacot-Descombes, "Video segmentation and camera motion characterization using compressed data", Proceedings of SPIE Volume: 3229, Multimedia Storage and Archiving Systems II, Editor(s): C.-C. Jay Kuo, Shih-Fu Chang, Venkat N. Gudivada, SPIE, USA

[MINKA] T. Minka, "An image database browser that learns from user interaction", MEng Thesis, MIT, 1996

[MINKA1] T.P. Minka, R. Picard, "Interactive learning using a "society of models" ", Technical report 349, MIT Media Lab Perceptual computing Section, 1995.

[MITRY] J. Mitry, "The Aesthetics and Psychology of the Cinema", The Athlone Press, London, 1998.

[MITRY2] J. Mitry, "Semiotics and the Analysis of Film ", Indiana University Press; ISBN: 025333733X, USA

[MIYAH] M. Miyahara and Y. Yoshida, "Mathematical transfomr of (R,G,B) color data to Munsell (H,V,C) color data," in SPIE Visual Communications and Image Processing, vol. 1001, pp. 650-657, 1998.

[NAGAS] A. Nagasaka, Y. Tanaka, "Automatic video indexing and full-video search for object appearances", in Visual Database Systems II (E. Knuth and L.M. Wegner, eds.), pp. 113-127, Elsevier, 1995.

[NAKA] Y. Nakajima, "A video browsing using fast scene cut detection for an efficient networked video database access," IEICE Trans. Inform. and Syst., vol. E77-D, no. 12, Dec. 1994.

[NAM] J. Nam, M. Alghoniemy, A. Tewfik. Audio visual content based violent scene characterization. In Proc. IEEE International Conference on Image Processing. IEEE, 1998.

[NAHPA] M. Naphade, T. Kristjansson, B. Frey, T. S. Huang, "Probabilistic Multimedia Objects Multijects: A novel Approach to Indexing and Retrieval in Multimedia Systems", Proc. IEEE International Conference on Image Processing, Volume 3, pages 536-540, Oct 1998, Chicago, IL

[NIBLA] W. Niblack, R. Barber, W. E. M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin, "The QBIC project: Querying image by content using color, texture, and shape," in SPIE Proceedings, A34vol. 1908, pp. 173-187, 1993.

[PAPPAS] T. Pappas, "An adaptive clustering algorithm for image segmentation", IEEE Trans. on Signal Processing 40 (1992) 901-914.

[PASS] G. Pass, R. Zabih, "Comparing images using joint histograms", Multimedia Systems (1999), vol.7, no.3, May 1999, pp.234-40. Springer-Verlag

[PATEL] N. Patel, I.K. Sethi, "Video shot detection and characterization for video databases", Pattern Recognition 30 (1997) 583-592.

[PATEL1] N. Patel and I.K. Sethi, "Video Segmentation for Video Data Management" in The Handbook of Multimedia Information Management, W.I. Grosky, R. Jain, and R. Mehrotra (Eds.), Prentice-Hall, 1997 .

[PAVLO] V. Pavlovic. Multimodal tracking and classification of audio visual features. In Proc. IEEE International Conference on Image Processing. IEEE, 1998.

[PEI] S. Pei, Yu-Zuong Chou, "Efficient MPEG compressed video analysis using macroblock type information", IEEE Transactions on Multimedia, vol.1, no.4, Dec. 1999, pp.321-33. Publisher: IEEE, USA.

[PENTL] Pentland, Picard, Sclaroff, PhotoBook:CB manipulation of image databases, In B. Fruht: Multimedia Tools and Applications, Kluwer, 1996.

[PICARD] R. W. Picard and T. P. Minka, "Vision texture for annotation", ACM/Springer Verlag Journal of Multimedia Systems, pp. 3-14, Vol. 3, 1995

[RAO] A. Rao, G. L. Lohse, "Towards a texture naming system: Identifying relevant dimensions of texture," in IEEE Proceedings of Visualization '93, pp. 220-228, October 1993.

[RASHE] Z. Rasheed, M. Shah, "Movie genre classification by exploiting audio-visual features of previews", Proc. 16th International Conference on Pattern Recognition, 2002. Comput. Vision Lab, Central Florida Univ., Orlando, FL

[RICH] E. Rich, K. Knight, "Artificial Intelligence", McGrawHill, Inc. New York, 1991

[ROACH1] M. Roach, "Video Genre Classification", PhD Thesis, University of Wales Swansea, 2002.

[ROACH2] M. Roach, J. Mason, L-Q. Xu "Video genre verification using both acoustic and visual modes", Int. Workshop on Multimedia Signal Processing, 2002

[RUI] Y. Rui, T. S. Huang, and S. Mehrotra, "Relevance Feedback Techniques in Interactive Content-Based Image Retrieval", Proceedings of the Conference on Storage and Retrieval of Image and Video Databases VI, (IS&T/SPIE-1998), San Jose, California, Jan. 1998.

[RUI1] Rui, Huang, Mehorta, CBIR with relevance feedback in ARS, ICIP 1997.

[RUMEL] Rumelhart, D. E., McClelland, J. L. & the PDP Research Group. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations, MIT Press, Cambridge, Massachusetts, 1986.

[SANTINI] S. Santini, "Exploratory image databases. Content-based retrieval", Academic Press, San Diego, USA, 2001.

[SARAC] C. Saraceno and R. Leonard!. Identification of story units in audio visual sequences by joint audio and video processing. In International Conference on Image Processing. IEEE, 1998.

[SAUSS] F. de Saussure, "Course in General Linguistics", (trans. Roy Harris). London, Duckworth, ([1916] 1983)

[SHAFER] R. Schäfer and T.Sikora, "Digital Video Coding Standards and Their Role in Video Communications", Proceedings of the IEEE Vol. 83, pp. 907-923, 1995.

[SCHNASE] J. Schnase, J. J. Leggett, D. L. Hicks, and R. L. Szabo, "Semantic data modeling of hypermedia associations," ACM Trans. Inform. Syst., vol. 11, no. 1, pp. 27-50, 1993.

[SCHOLSS] G. Scholss, M. J. Wynblatt, "Providing definition and temporal structure for multimedia data," Multimedia Syst., vol. 3, pp. 264-277, 1995.

[SETHI] I. Sethi, N.V. Patel, A statistical approach to scene change detection, in: Proc. IS&T/SPIE Conf. Storage and Retrieval for Image and Video Databases III 2420, San Jose, 1995, pp. 2-11.

[SHAHR] B. Shahraray, "Scene change detection and content-based sampling of video sequences", in Proc. IS&T/SPIE 2419, pp. 2-13, 1995.

[SHEN] K. Shen, E. Delp, "A fast algorithm for video parsing using MPEG compressed sequences", in: Proc. Intern. Conf. Image Processing (ICIP'96), Lausanne, 1996.

[SMITH] J. Smith, S.-F. Chang, "Automated image retrieval using color and texture," Tech. Rep. 414-95-20, Columbia University, July 1995.

[SMITH1] J. Smith, S.-F. Chang, "Quad-tree segmentation for texture-based image query," in ACM Multimedia 94, pp. 279-286, 1994.

[SMITH2] J. R. Smith and S.-F. Chang, "VisualSEEk: A Fully Automated Content-Based Image Query System", Proceedings of the ACM Conference Multimedia, New York, 1996.

[SMOUL] A. Smoulders, M. Worring, S. Santini, and A. Gupta, "Content based image retrieval at the end of the early years", pami, 22(12): 1349-1380,2000.

[SMOLIAR] S. W. Smoliar, J. D. Baker, T. Nakayama, and L. Wilcox, "Multimedia Search: An Authoring Perspective", Proceedings of the First International Workshop on Image Databases and Multimedia Search (IAPR-1996), pp. 1-8, Amsterdam, The Netherlands, Aug. 1996.

[SOBCHA] T. Sobchack and V Sobchack. An introduction to film. Scot, Foresman and Company, 1987.

[SNOEK] C. Snoek, M. Worring, "A Review on Multimodal Video Indexing", IEEE International Conference on Multimedia and Expo, volume 2, pp. 21-24, Lausanne, Switzerland, 2002.

[SRINI] M. Srinivasan, S. Venkatesh, and R. Hosie, "Qualitative Estimation of Camera Motion Parameters from Video Sequences, Pattern Recognition, vol.30, no.4, April 1997, pp.593-606. Publisher: Elsevier, UK

[STEELS] L. Steels, "Language Games for Emergent Semantics", ", in "Emergent semantics" edited by Staab, S., IEEE Intelligent Systems, Volume: 17 Issue: 1, Jan.-Feb.

[STICKER] M. Stricker and M. Orengo, "Similarity of color images," in Proceedings of Storage and Retrieval for Image and Video Databases III, vol. 2420, pp. 381-392, 1995.

[SWAIN] M. Swain, D. H. Ballard, "Color indexing" International Journal of Computer Vision, vol. 7, no. 1, pp. 11-32, 1991.

[SWANB] D. Swanberg, C.-F. Shu, and R. Jain, "Knowledge guided parsing in video databases," in Proc. IS&T/SPIE Symp. Electronic Imaging: Science and Technology, San Jose, CA, Feb. 1993.

[SZUMMER] M. Szummer and R. Picard, “Indoor-Outdoor Image Classification”, IEEE International Workshop in Content-Based Access to Image and Video Databases, in conjunction with ICCV'98, Bombay, India, Jan. 1998.

[TAMURA] H. Tamura, S. Mori, and T. Yamawaki, “Textural features corresponding to visual perception,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 8, pp. 460-473, June 1978.

[TASKI] C. Taskiran, E. Delp, “Video scene change detection using the generalized sequence trace”, in: Proc. IEEE Int. Conf. Acoustics, Speech & Signal Processing, Seattle, 1998, pp. 2961-2964.

[TONG] Tong S. and Chang E. “Support vector machine active learning for image retrieval”, Proceedings of the ninth ACM international conference on Multimedia, p.p. 107 – 118, 2001, ACM Multimedia, Ottawa, Canada.

[TONOM] Y. Tonomura, A. Akutsu, Y. Taniguchi, and G. Suzuki, “Structured video computing,” IEEE Multimedia Mag., pp. 34-43, Fall 1994.

[TANSLEY] R. Tansley, “The Multimedia Thesaurus: Adding A Semantic Layer to Multimedia Information”, Ph. D. Thesis, Computer Science, University of Southampton, UK, Aug. 2000.

[TROUNG] B. Tu Truong, C. Dorai, S. Venkatesh, “Automatic Genre Identification for Content-Based Video Categorization”, 15th International Conference on Pattern Recognition, September, 2000

[TVENCY] “Encyclopedia of television”, editor, Horace Newcomb, Museum of Broadcast Communications, Chicago. London. Fitzroy Dearborn. 1997 http://www.museum.tv/archives/etv/Encyclopediatv.htm

[TVTHEO] “Critical Dictionary of Film and Television Theory”, R. Pearson, P. Simpson, Routledge, an imprint of Taylor & Francis Books Ltd, November, 2000

[VAILAYA] A. Vailaya, A. Jain, and H.J. Zhang, “On Image Classification: City vs. Landscape”, IEEE Workshop on Content-Based Access of Image and Video Libraries, Santa Barbara, CA, June 1998.

[VASCO] VASCONCELOS, IEEE Trans. on Imag. Proc, Jan 2000…

[WANG] J. Wang and E. H. Adelson, “Spatio-temporal segmentation of video data”, in Image and Video Processing II, Proceedings of the SPIE, San Jose, CA, February 1994, Vol. 2182.

[WEISS] R. Weiss, A. Duda, and D. K. Gifford, “Composition and search with a video algebra,” IEEE Multimedia Mag., pp. 12-25, Spring 1995.

[WU] Wu Y., Tian Q. and Huang T.S., “Discriminant EM algorithm with application to image retrieval”, IEEE Conf. Computer Vision and Pattern Recognition, South Carolina, 2000.

[XIONG] W. Xiong, J. C.-M. Lee, “Efficient scene change detection and camera motion annotation for video classification”, Computer Vision and Image Understanding 71(2) (1998) 166-181.

[XIONG1] W. Xiong, J. C.-M. Lee, M.C. Ip, “Net comparison: a fast and effective method for classifying image sequences”, in: Proc. SPIE Conf. Storage and Retrieval for Image and Video Databases III 2420, San Jose, CA, 1995, pp. 318-328.

[XM] Sources for MPEG-7 XM Software, MPEG-7 eXperimentation Model (XM), http://www.lis.ei.tum.de/research/bv/topics/mmdb/e_mpeg7.html

[YEO] B. Yeo, B. Liu, “Rapid scene analysis on compressed video”, IEEE Transactions on Circuits & Systems for Video Technology, vol.5, no.6, Dec. 1995, pp.533-44. Publisher: IEEE, USA.

[YOW] D. Yow, B-L Yeo, M. Yeung, and B. Liu, "Analysis and presentation of soccer highlights from digital video," in Proc. Asian Conf. on Computer Vision, 1995.

[YU] H. Yu, G. Bozdagi, S. Harrington, "Feature-based hierarchical video segmentation", in: Proc. Int. Conf. on Image Processing (ICIP'97), Santa Barbara, 1997, pp. 498-501.

[YUSOFF] Y. Yusoff, W. Christmas, J. Kittler, "Video shot cut detection using adaptive thresholding", Proceedings of the 11th British Machine Vision Conference, Univ. Bristol. Part vol.1, 2000, pp. 362-71 vol.1, Bristol, UK.

[ZABIH] R. Zabih, J. Miler, K. Mai, "A feature-based algorithm for detecting and classifying production effects", Multimedia Systems 7 (1999) 119-128.

[ZABIH1] R. Zabih, J. Miller and K. Mai, "A feature-based algorithm for detecting and classifying scene breaks", Proc. ACM Multimedia '95, pp.189-200, 1995.

[ZAKHOR] A. Zakhor and F. Lari, "Edge-based 3-D camera motion estimation with application to video coding," IEEE Trans. Image Processing, vol. 2, pp. 481-498, Oct. 1993.

[ZETTL] H. Zettl, "Sight, Sound, Motion: Applied Media Aesthetics", 3rd Edition, Wadsworth Pub Company, 1999.

[ZHANG] H. Zhang, A. Kankanhalli and W. Smoliar, "Automatic partitioning of full-motion video", Multimedia Systems, vol.1, no.1, pp. 10-28, 1993.

[ZHANG1] H. Zhang, "Content-based Video Browsing and Retrieval", from "Handbook of Multimedia Computing" editor-in-chief Furht B., CRC Press, Boca Raton, Florida, USA, 1999.

[ZHANG2] H. Zhang, C. Y. Low, and S.W. Smoliar, "Video parsing and browsing using compressed data", Multimedia Tools Appl. 1, 1995, 91-113.

[ZHANG3] H. Zhang, S. W. Smoliar, and J. H. Wu, "Content-based video browsing tools", in Proceedings SPIE Conference on Multimedia Computing and Networking, San Jose, CA, February 1995.

X. ANNEXES

X.1. LIST OF AUTHOR'S PUBLICATIONS

J. Calic and E. Izquierdo, "Towards Real-Time Shot Detection in the MPEG Compressed Domain", Proceedings of the Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS'2001, Tampere, Finland, May 2001.

J. Calic and E. Izquierdo, "Temporal Segmentation of MPEG video streams", Special issue on Image Analysis for Multimedia Interactive Services, EURASIP Journal on Applied Signal Processing, June 2002.

J. Calic and E. Izquierdo, "Temporal Video Segmentation for Real-Time Key Frame Extraction", Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Orlando, FL, USA, May 2002.

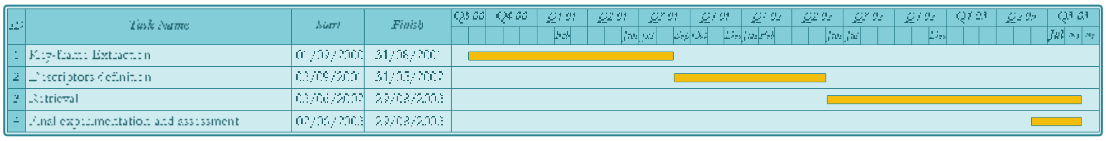
J. Calic and E. Izquierdo, "Efficient Key-Frame Extraction and Video Analysis", Proceedings of the IEEE International Conference on Information Technology ITCC 2002, Las Vegas, NV, USA, April 2002.

J. Calic and E. Izquierdo, "A Multiresolution Technique For Video Indexing And Retrieval", Proceedings of the IEEE International Conference on Image Processing 2002, Rochester, NY, USA, September 2002.

A. Dorado, J. Calic, E. Izquierdo, "A Rule-Based Video Annotation System", Circuits and Systems for Video Technology, IEEE Transactions on , Volume: 14 , Issue: 5 , p.p. 622 – 633, May 2004.

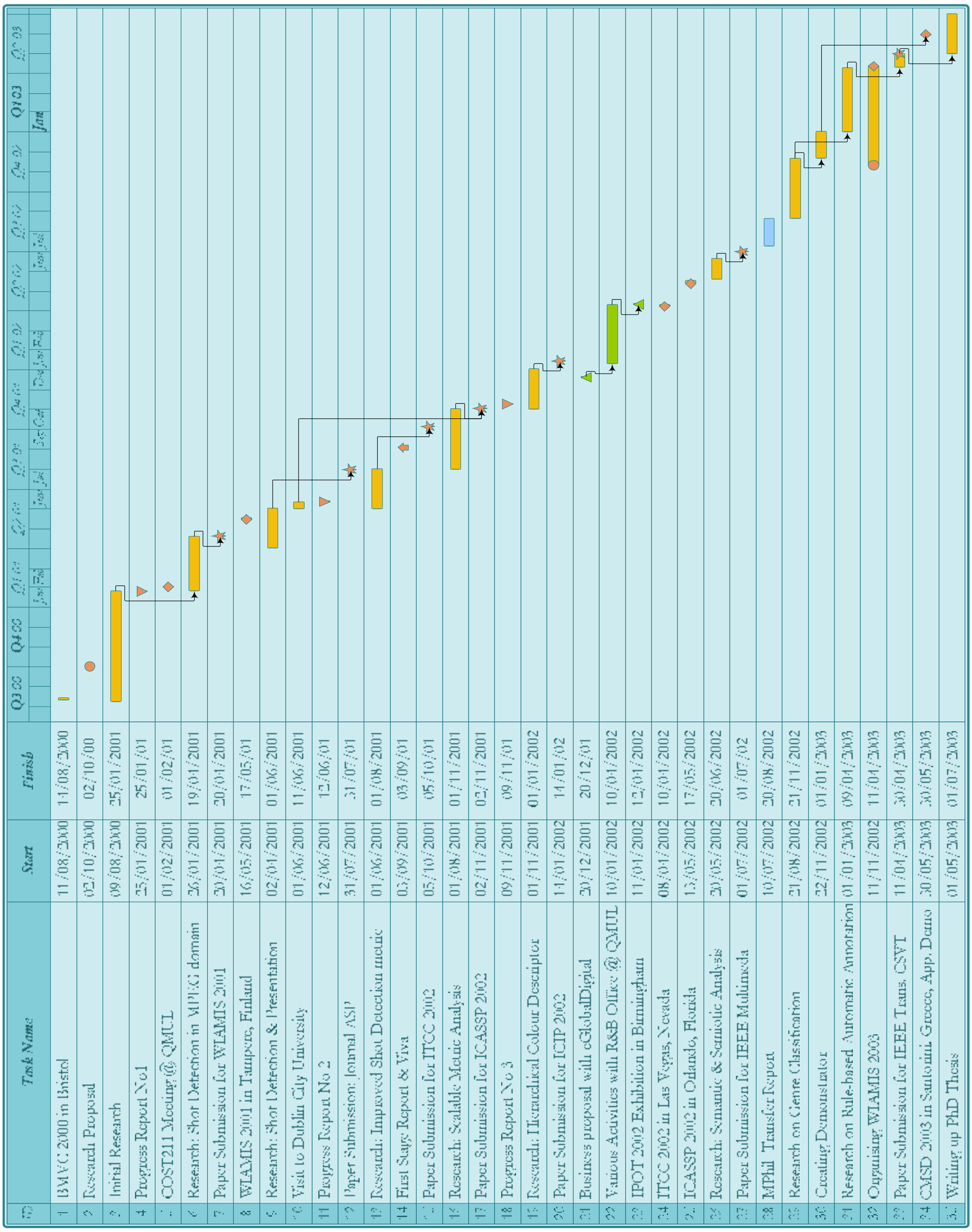
X.2. GANTT CHARTS

Proposed GANTT Chart



ID	Task Name	Start	Finish
1	Key-Frame Extraction	01/09/2000	31/08/2001
2	Descriptors definition	03/09/2001	31/05/2002
3	Retrieval	03/06/2002	29/08/2003
4	Final experimentation and assessment	02/06/2003	29/08/2003

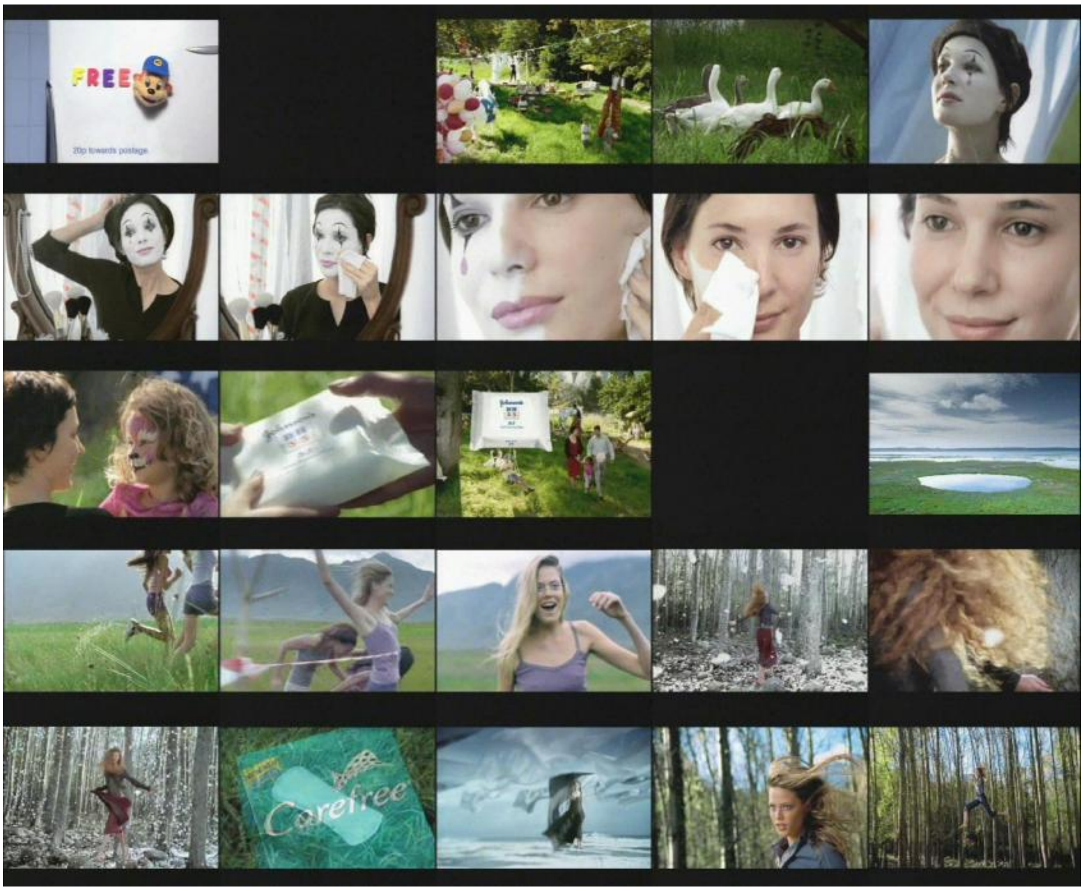
Project GANTT Chart

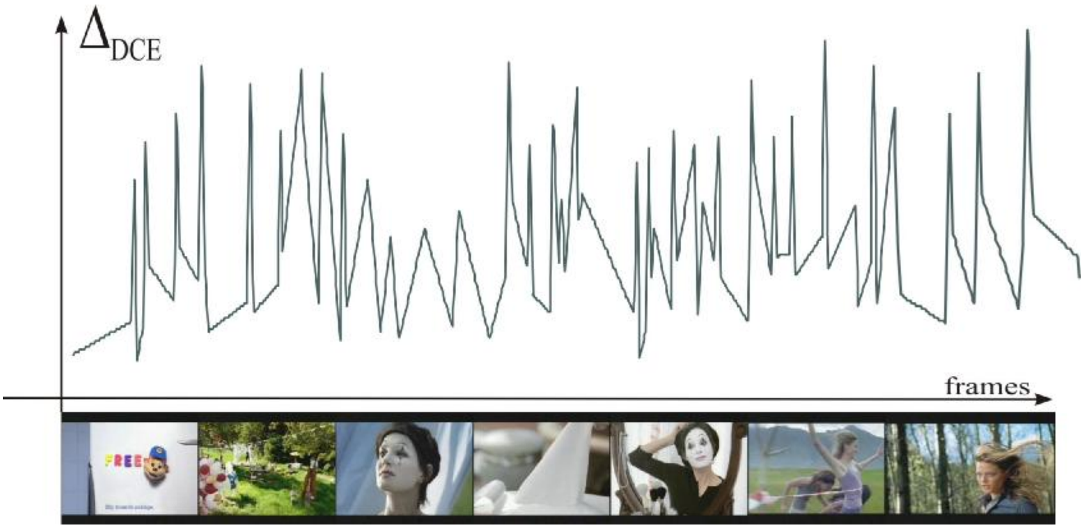


ID	Task Name	Start	Finish
1	BMVC 2000 in Bristol	11/08/2000	11/08/2000
2	Research Proposal	02/10/2000	02/10/00
3	Initial Research	09/08/2000	25/01/2001
4	Progress Report No1	25/01/2001	25/01/01
5	COST211 Meeting @ QMUL	01/02/2001	01/02/01
6	Research: Shot Detection in MPEG domain	26/01/2001	19/04/2001
7	Paper Submission for WIAMIS 2001	20/04/2001	20/04/2001
8	WIAMIS 2001 in Tampere, Finland	16/05/2001	17/05/01
9	Research: Shot Detection & Presentation	02/04/2001	01/06/2001
10	Visit to Dublin City University	01/06/2001	11/06/2001
11	Progress Report No 2	12/06/2001	12/06/01
12	Paper Submission: Journal ASP	31/07/2001	31/07/01
13	Research: Improved Shot Detection metric	01/06/2001	01/08/2001
14	First Stage Report & Viva	03/09/2001	03/09/01
15	Paper Submission for ITCC 2002	05/10/2001	05/10/01
16	Research: Scalable Metric Analysis	01/08/2001	01/11/2001
17	Paper Submission for ICASSP 2002	02/11/2001	02/11/2001
18	Progress Report No 3	09/11/2001	09/11/01
19	Research: Hierarchical Colour Descriptor	01/11/2001	01/01/2002
20	Paper Submission for ICIP 2002	14/01/2002	14/01/02
21	Business proposal with eGlobalDigital	20/12/2001	20/12/01
22	Various Activities with R&B Office @ QMUL	10/01/2002	10/04/2002
23	IPOT 2002 Exhibition in Birmingham	11/04/2002	12/04/2002
24	ITCC 2002 in Las Vegas, Nevada	08/04/2002	10/04/2002
25	ICASSP 2002 in Orlando, Florida	15/05/2002	17/05/2002
26	Research: Semantic & Semiotic Analysis	20/05/2002	20/06/2002
27	Paper Submission for IEEE Multimeda	01/07/2002	01/07/02
28	MPhil Transfer Report	10/07/2002	20/08/2002
29	Research on Genre Classification	21/08/2002	21/11/2002
30	Creating Demonstrator	22/11/2002	01/01/2003
31	Research on Rule-based Automatic Annotation	01/01/2003	09/04/2003
32	Organising WIAMIS 2003	11/11/2002	11/04/2003
33	Paper Submission for IEEE Trans. CSVT	11/04/2003	30/04/2003
34	CMSD 2003 in Santorini, Greece, App. Demo	30/05/2003	30/05/2003
35	Writing up PhD Thesis	01/05/2003	01/07/2003

X.3. SUMMARY OF USED VIDEO SEQUENCES

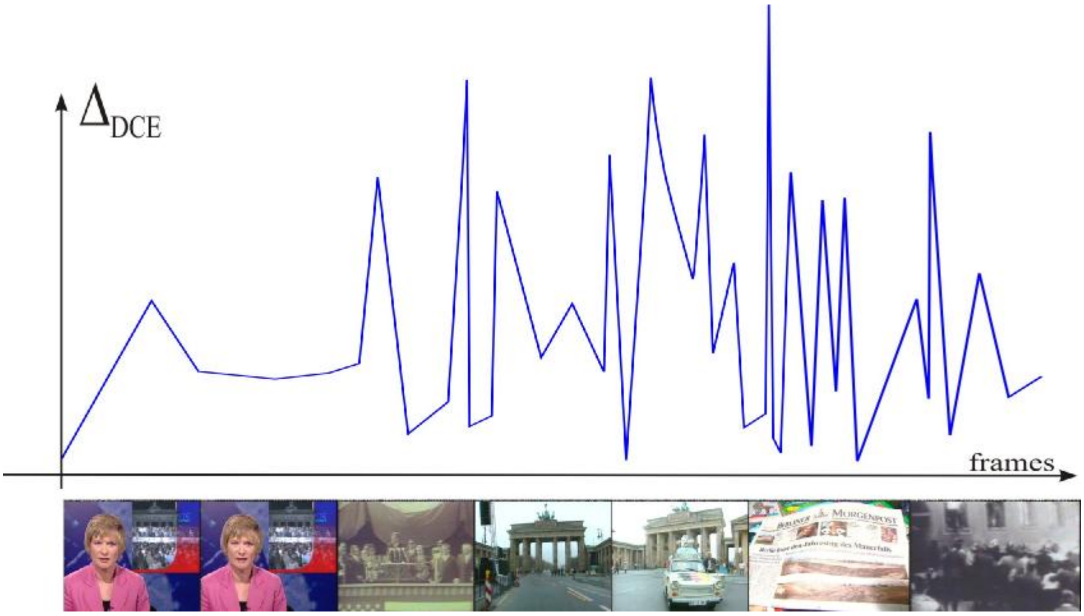
ULOSCI.MPG





NEWS136.MPG

RTÉ news Report SINEAD CROWLEY
RTÉ news Report SINEAD CROWLEY
BERLINER MORGENPOST
Berlin feiert den Jahrestag des Mauerfalls
Als das Mon ter noch lebte
Jahre ins Gefängnis
Riesiger Feuerz über Deutschlan
vor 10 Jahren Glücks-Nacht Berlin
Bild
Egon Krenz
Krenz ab ins Gefängnis


Δ_{DCE}
frames