

DUBLIN CITY UNIVERSITY
SCHOOL OF ELECTRONIC ENGINEERING

AN OBJECT-BASED APPROACH TO
RETRIEVAL OF IMAGE AND VIDEO
CONTENT

by
Sorin Sav


A thesis submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy

Supervisor: Dr. Noel E. O'Connor

JULY 2006

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of PhD in Electronic Engineering is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: 
Date: 20/09/2006

ID No.: 50161229

Abstract

Promising new directions have been opened up for content-based visual retrieval in recent years. Object-based retrieval which allows users to manipulate video objects as part of their searching and browsing interaction, is one of these. It is the purpose of this thesis to constitute itself as a part of a larger stream of research that investigates visual objects as a possible approach to advancing the use of semantics in content-based visual retrieval

The notion of using objects in video retrieval has been seen as desirable for some years, but only very recently has technology started to allow even very basic object-location functions on video. The main hurdles to greater use of objects in video retrieval are the overhead of object segmentation on large amounts of video and the issue of whether objects can actually be used efficiently for multimedia retrieval. Despite this, there are already some examples of work which supports retrieval based on video objects.

This thesis investigates an object-based approach to content-based visual retrieval. The main research contributions of this work are a study of shot boundary detection on compressed domain video where a fast detection approach is proposed and evaluated, and a study on the use of objects in interactive image retrieval. An object-based retrieval framework is developed in order to investigate object-based retrieval on a corpus of natural image and video. This framework contains the entire processing chain required to analyse, index and interactively retrieve images and video via object-to-object matching. The experimental results indicate that object-based searching consistently outperforms image-based search using low-level features. This result goes some way towards validating the approach of allowing users to select objects as a basis for searching video archives when the information need dictates it as appropriate.

Acknowledgements

I would like to express my sincere appreciation and gratitude to my academic supervisor Dr. Noel E. O'Connor, and to Prof. Alan Smeaton for giving me the opportunity to pursue this work, for their guidance and extensive support in critical moments.

The thesis' reviewers, Prof. Stefan Rüger and Dr. Gareth Jones, are kindly acknowledged for their feedback and constructive comments.

I am indebted to my colleagues in the Centre for Digital Video Processing whom have been very supportive from both the technical and social standpoint. I would like to specially acknowledge in here the "engineering" side of the CDVP: Csaba Czirik, Tomasz Adamek, Orla Duffner, Ciarán Ó Conaire, David Sadlier, Bart Lehane, Phil Kelly, Saman Cooray, Jovanka Malobabić, Andrew Kinane, Daniel Larkin, Kealan McCusker, Hervé Le Borgne, Eddie Cooke, Valentin Mureşan, Roman Jarina, Dr. Noel Murphy and Dr. Sean Marlow.

Special thanks to Hyowon Lee for designing the smart graphical interfaces used in this work, for the papers and experiments he had contributed to and first of all for his friendship. Many thanks to Tomasz for his friendship, the many discussions we had during these years and for his early approach on interactive segmentation which is incorporated in this work.

Last, but not least, I owe deep gratitude to all those unnamed friends who by their constant encouragement supported me through these years.

Table of Contents

Abstract	i
Acknowledgements	ii
Table of Contents	iii
1 Introduction	1
1.1 Challenges of visual content retrieval	1
1.2 Objectives of this research	3
1.3 Structure of this thesis	3
2 Content based visual information retrieval	5
2.1 Introduction	5
2.2 Video retrieval	7
2.3 Characteristics of content based visual information retrieval	9
2.3.1 Visual search	11
2.3.2 Visual browsing	11
2.3.3 User interaction in retrieval	12
2.4 Visual features: color, shape, and texture	13
2.4.1 Colour retrieval	14
2.4.2 Shape retrieval	14
2.4.3 Texture retrieval	15
2.5 Object-based retrieval	16
2.6 Evaluation of visual retrieval systems	18
2.7 Selected visual retrieval systems	19
2.8 Discussion	24
3 Visual content descriptors for image and video retrieval	27
3.1 Introduction	27
3.2 Colour	28
3.2.1 Colour space	29
3.2.2 Colour moments	30
3.2.3 Colour histogram	30
3.2.4 Colour coherence vector	31
3.2.5 Colour correlogram	32
3.2.6 Invariant colour features	32

3.3	Shape	33
3.3.1	Moment invariants	33
3.3.2	Compactness, eccentricity, and major axis orientation	34
3.3.3	Turning angles	35
3.3.4	Fourier descriptors	36
3.3.5	Curvature scale space	37
3.4	Texture	38
3.4.1	Tamura features	38
3.4.2	Wold features	40
3.4.3	Simultaneous auto-regressive (SAR) model	41
3.4.4	Gabor filter features	42
3.4.5	Wavelet transform features	43
3.5	MPEG-7 visual descriptors	44
3.5.1	MPEG-7 Colour descriptors	44
3.5.2	MPEG-7 Shape descriptors	45
3.5.3	MPEG-7 Texture descriptors	45
3.6	Similarity measures	46
3.6.1	Earth Mover distance	46
3.6.2	Minkowski form distance	47
3.6.3	Fractional distances	47
3.6.4	Quadratic form distance	48
3.6.5	Mahalanobis distance	48
3.6.6	Kullback-Leibler divergence	49
3.6.7	Jeffrey divergence	49
3.7	Video sequence segmentation	50
3.7.1	A brief survey of shot boundary segmentation	50
3.7.2	Evaluation measures	53
3.8	Conclusions	56
4	Image segmentation	57
4.1	Introduction	57
4.2	Pixel-based segmentation	59
4.2.1	Histogram thresholding	59
4.2.2	Clustering techniques	60
4.2.3	Fuzzy clustering	62
4.3	Area-based segmentation	62
4.3.1	Region-growing techniques	63
4.3.2	Split-merge techniques	64
4.3.3	Graph-theory techniques	64
4.4	Contour-based segmentation	65
4.5	Neural network based segmentation	66
4.6	Physics-based segmentation	67
4.7	Motion-based segmentation	68
4.8	Interactive tools for image segmentation	70
4.9	Evaluation of image segmentation	72

4.10	Conclusions	73
5	Relevance feedback	75
5.1	Introduction	75
5.2	Principle of relevance feedback	77
5.2.1	Vector model	78
5.2.2	Probabilistic model	80
5.3	Relevance feedback in image and video retrieval	81
5.3.1	Utilisation scenarios	83
5.3.2	Feedback information	83
5.3.3	Data models	84
5.3.4	Selection strategies	86
5.4	Evaluation of retrieval with relevance feedback	87
5.4.1	Residual ranking	89
5.4.2	Rank freezing	89
5.5	Summary	90
6	An approach to object-based video retrieval	91
6.1	Introduction	91
6.2	Overall structure of the framework	92
6.3	Video sequence segmentation	93
6.3.1	Abrupt shot change detection	95
6.3.2	Threshold settings for abrupt transitions	96
6.3.3	Gradual shot change detection	97
6.3.4	Threshold settings with gradual transitions	97
6.3.5	Evaluation of shot change detection	98
6.3.6	Extension to gradual transition classification	100
6.3.7	Applicability to different encoding patterns	102
6.3.8	Keyframe extraction	103
6.4	Object outlining and visual feature extraction	103
6.4.1	Object outlining	103
6.4.2	Dominant colour	106
6.4.3	Shape compactness descriptor	107
6.4.4	Texture browsing descriptor	108
6.5	A Gaussian mixture model for relevance feedback	109
6.5.1	Gaussian mixture model	109
6.5.2	Estimation of the mixture's components	110
6.5.3	Estimation of mean and covariance matrix	111
6.5.4	Weighting of mixture components	112
6.6	Object interaction interface	112
6.6.1	Interacting with objects	112
6.6.2	Refinement with query branching	114
6.7	Summary	115

7 Retrieval experiments	117
7.1 Introduction	117
7.2 Investigation of object-based retrieval performance	118
7.2.1 Experimental setup	118
7.2.2 Experimental procedure	119
7.2.3 Refinement with query branching	120
7.2.4 Results interpretation	121
7.2.5 Discussion	122
7.3 Investigation of object-based searching	122
7.3.1 Experimental setup	123
7.3.2 Search topic formulation	124
7.3.3 Experimental design methodology	125
7.3.4 Experimental procedure	126
7.3.5 Evaluation metrics	127
7.3.6 Results interpretation	128
7.3.7 Discussion	129
7.4 Experimental setting for further investigation of object-based retrieval . . .	129
7.5 Summary	131
8 Conclusions and future work	137
8.1 A Brief Review	137
8.2 Conclusions	139
8.3 Future work	140
References	142
Appendix A. List of author's publications	A-1
Appendix B. Examples of retrieved images	B-1

Chapter 1

Introduction

Information technology has a vast impact on human society. New technologies for communication and information sharing shape novel ways for human creativity and interaction. This trend is likely to continue over future decades since what we see nowadays is just its embryonic stages. Nowadays digital video is one of the areas which attracts the spotlight when information sharing comes in discussion and in this dissertation we focus on accessing information stored under digital video format.

1.1 Challenges of visual content retrieval

Until a few years ago the content on the Internet was predominantly textual information, however, recently the amount of visual data, in all its forms, is expanding at an amazing rate. While it is not expected that visual data will ever supersede textual information, at least in terms of number of documents, the opportunities and challenges raised by this fast-growing visual content cannot be ignored.

Today digital video is the focus of much debate in particular over aspects related to sharing and copyright management of video content. However, beside these aspects which regularly feature in the news broadcasts, there are many other issues related to the management of video content, hardly ever known to the wider public.

1. Introduction

Many Internet users experiencing the efficiency and reliability provided by web search engines such as Google would find it puzzling that current visual retrieval performance is very poor in comparison to text retrieval. Indeed, text-based search engines have proved invaluable tools in navigating the Web. However, when it comes to searching for visual or audio content results seldom match expectations.

Text documents contain semantic information in a pure form, often organised into a clear and rigorous structure. Within this structure there are well-defined markers such as punctuation marks, interword spaces and typographical marks, which unambiguously delimit the semantic units (i.e. words). Stop words, stemming and dictionaries further improve indexing making the implementation of basic text search and retrieval a straightforward statistical problem. In fact, considering all the above-mentioned factors it comes as a surprise that text retrieval does not successfully perform yet at the semantic level.

Visual documents reflect semantic information but the information is not organised into a semantic structure. Moreover, in the case of video, beyond the level of frame and sequence of frames, the structure is largely variable and sometimes ambiguous. From a text retrieval perspective visual content looks like a surreal book where all punctuation marks, spaces and interword spaces and typographical marks have been removed, and the letters (equivalent to the pixels) are conglomerated all together. Nevertheless, the paragraphs (equivalent to the frames) are clearly delimited. Ironically, in the case of video, humans do not notice the paragraphs markers but can clearly decipher the words (objects appearing in the frames). This is exactly contrary to what a reader of our hypothetical book would experience, who would face a tenfold more complex challenge than Champollion by trying to structure foreign text (the reader has no knowledge about the book or how to separate it into words) without a lexicon of terms.

Imagine the custodian of a vast library containing a few millions of such surreal volumes being asked to recommend a few books on the same topic as a given one. This is the challenge faced by content-based visual information retrieval systems. The parallel drawn here between textual and visual media is somehow forced and inaccurate since each of these media have specific attributes and formats which do not equate each other, however it gives an intuitive understanding on the complexity of visual retrieval. The discussion in this thesis focuses on object-based retrieval as a modality to incorporate semantics into content-based visual retrieval.

1.2 Objectives of this research

The first objective of this thesis is to place in context the investigation carried out here. To achieve this objective, an overview of content-based visual retrieval is first presented. The low-level visual descriptors that form the core of indexing and matching are then introduced. Although, low-level features can be automatically extracted from the content they are largely disconnected from the semantic concepts intuitive to the user. Image segmentation is introduced as an approach to organising visual descriptors into semantic structures, namely objects, which are afterwards used in the retrieval process. The background review ends with an overview of relevance feedback.

The second objective is to investigate an object-based approach to content-based visual retrieval. The approach incorporates a novel relevance feedback mechanism. A object-based retrieval framework is developed by the author in order to experimentally validate the proposed approach. This framework contains the entire processing chain required to analyse, index and interactively retrieve images and video via object-to-object matching. A set of experiments are carried out to illustrate its performance.

An implicit objective of any research work is to indicate directions for further research. The final objective of this thesis is to consider the proposed approach and to present possibilities for improvement.

1.3 Structure of this thesis

The structure of this thesis is as follows. The next chapter gives an overview of content-based visual information retrieval. It covers aspects specific to image and video retrieval. First, issues such as search, browsing and user interaction are summarised. Then the role played by visual features such as colour, texture and shape are briefly reviewed. The use of objects in retrieval is discussed in reference to previous work done in the field. The chapter also presents aspects related to the evaluation of retrieval systems and a brief overview of a few well known content based image and video retrieval systems.

The third chapter details the commonly used visual content descriptors for image and video. It covers the descriptors associated with colour, shape and texture features. A section of

this chapter is dedicated to the MPEG-7 descriptors. The similarity measures frequently used in computing features are then introduced. The chapter ends with a presentation on video sequence segmentation, which is the starting point in any video analysis process.

Image segmentation as an approach to organising visual descriptors into semantic structures is introduced in chapter four. This chapter provides a review of segmentation covering the main techniques reported in the research literature and issues related to the evaluation of segmentation output. The problem of semantic segmentation is also discussed from an interactive perspective.

An overview of relevance feedback is given in chapter five. It covers the general principle and also aspects specific to the use of relevance feedback mechanisms in content-based image and video retrieval. The presentation focuses on utilisation scenarios, data models and selection strategies in the feedback process. The evaluation of retrieval with relevance feedback is also described in this chapter.

The sixth chapter describes an approach to using semantic objects in content-based visual retrieval. A framework based on this approach is implemented and discussed. This framework contains the entire processing chain required to analyse, index and interactively retrieve images and video via object-to-object matching. Relevance feedback is employed to drive the query updating process.

The empirical investigation carried out on the proposed framework is presented in chapter seven. Two experiments have been designed for this purpose. The first experiment attempts to investigate the usability of semantic objects in content based retrieval by involving real-users on a set of retrieval tasks, while the second experiment targets evaluating the retrieval performance of the proposed framework by using an expert user.

The final chapter reviews the conclusions of this thesis and suggests directions for future research. The benefits and limitations of object-based approaches to retrieval of visual content are also discussed.

Chapter 2

Content based visual information retrieval

Although today much video data is produced in analog form there is an increasing trend in moving to digital format. The digital format holds a powerful advantage allowing fast and reliable processing of video data, the core enabling technology for content based retrieval of video. Content based retrieval holds the key to searchable video databases which in future will help users locate content of interest based on rich visual, audio and text cues. Ideally content based retrieval technology will allow computers to automatically annotate, summarise and retrieve clips by interpreting the semantic content of the video and matching this against user information needs.

2.1 Introduction

Digital technologies have dramatically increased the amount of multimedia content produced nowadays. The extensive amount of multimedia data available raises challenges for managing vast quantities of content. The need for efficient storage and retrieval of visual content has been recognised more than a decade ago [1].

Content based visual information retrieval (CBVIR) describes the process of retrieving images or video from a large collection on the basis of visual features, such as colour, shape

and texture, that are directly extracted from the image or video content. Although metadata information such as keywords from manual annotation or structural information like title, duration, recording format, etc, is sometimes used in conjunction with content features the general understanding of the term excludes image retrieval based on such annotation.

Although image retrieval by textual annotations can be easily implemented using existing search technology, this requires humans to annotate every image or video clip in a database, a scenario obviously impractical for large amounts of content. Another drawback with textual annotation derives from the possible confusing use of synonyms. These inherent limitations of metadata-based systems have stimulated a growing interest in CBVIR. The extent of the CBVIR potential today and the directions for its future evolution is discussed below.

The aim of content based visual information retrieval is to develop new algorithms, techniques and interfaces that could improve users access to image and video collections. Access to a video collection is defined by the effectiveness and efficiency of a retrieval system in supporting users in their searches for content of interest. User satisfaction in interacting with the retrieval system is also an important aspect of retrieval. The focus of this thesis is on general ad hoc retrieval by a hypothetical professional user performing visual queries on a video and image archive. This chapter covers the state-of-the-art in content based image and video retrieval with particular emphasis on the aspects related to the architecture of a retrieval system. However, we present here only a top level overview of a CBVIR system, the detailed description of the low-level features, similarity measures and relevance feedback strategies mentioned in this chapter being presented elsewhere in the thesis.

Content based image and video retrieval derives from early computer vision research focused on developing feature based similarity models for images and video [2, 3]. While image features and similarity models are derived from computer vision, the retrieval models and evaluation methodology come from the information retrieval field [4]. Initial research efforts aimed at improving the feature similarity search over images. Influential retrieval systems developed in the early years of retrieval include: QBIC [5], Virage [6], Photobook [7], VisalSEEk and WebSEEk [8, 9], Netra [10], MARS [11], Informedia [12], and ANSES [13].

Feature based similarity search engines proved useful for contexts where basic features such as colour or texture can be directly used for querying [1], some of the most successful applications being trademark searching [14] and detection of objectionable content [15].

However it became clear that feature based similarity search is not user-friendly and could not be used effectively by inexperienced users. This is a general problem termed “the semantic gap” between the easily computable low level content-based media features and the high level concepts intuitive to the user. Approaches based on relevance feedback [16] attempt to derive semantics by continuous feedback from the user. Concept identification, the most common being the human face [17, 18] is another avenue to introducing semantics into video retrieval.

The remainder of this chapter is organised as follows. The next section introduces aspects related to retrieval of video data. Some characteristics of visual information retrieval are presented in Section 2.3. The visual features commonly used in content-based image and video retrieval are briefly described in Section 2.4. A more detailed presentation of these features can be found in Chapter 3. Section 2.5 introduces the concept of object based-retrieval, which is the topic of this thesis. Evaluation of visual retrieval is presented in Section 2.6 followed by a selective list of CBVIR systems in Section 2.7. The chapter concludes in Section 2.8 with a discussion on the retrieval of visual information.

2.2 Video retrieval

Video is a linear medium made of a sequence of frames that can be logically organised into shots, which are defined as the contiguous set of frames taken by a single uninterrupted camera over time. Shots can be further grouped into logical or semantic units termed scenes. Higher levels of abstraction can be built by organising shots or scenes into a thread of narration such as a storyline. Various other forms of organisation can be imagined, but not all of them may have a meaningful structure.

Robust shot boundary detection is the entry point of any video processing system. Uncompressed and compressed domain approaches have been proposed most commonly based on computing the distance between colour histograms corresponding to consecutive frames in a video [5, 19, 20]. An alternative approach exploiting the motion within video [21] proved good results and also allowed classification of the video shots into categories such as zoom-in, zoom-out, pan. Recent research work focuses on detection and classification of gradual transitions [22].

2. Content based visual information retrieval

There is currently a sustained research interest directed towards automatic grouping of shots into scenes and further into storylines, however this is proving quite challenging apart from few a well-structured domains. Distinct successes have been recorded in the analysis of the TV news genre [23] which follows a particularly well-defined structure. The temporal segmentation of other types of less constrained video is more problematic.

Once shot boundaries are detected in the video the next step is the selection of representative keyframes for each shot. Keyframes are intended to provide for users an intuitive visual description and pointers to the content, and also can be used as a computationally less expensive way of data reduction in the processing tasks. Although attempts to develop sophisticated algorithms for keyframe determination have been researched and some proved successful, there is no definitive technique of selecting appropriate keyframes. The most common approach taken in practice is the selection of a frame close to the middle of the shot.

Other primitives apart from shot boundaries and keyframes can be computed and used in the indexing and retrieval of video data. These primitive include [24]: speech recognition, audio discrimination between speech and music, speaker segmentation, camera and object motion, slow motion detection in sports action replays, face recognition, text overlay, scene characterisation such as indoor/outdoor classification. However, throughout this thesis we focus only on low-level visual features and other types of features are mentioned for the sake of completeness.

Generally video retrieval is done at the level of image-to-image and sometimes at shot-to-shot matching. Image-to-image matching has the advantage of relying on well investigated techniques from still image retrieval. There is still a largely untapped potential in shot-to-shot matching based on other features such as camera and object movement although this is not feasible on large scale collections [24]. However, apart from specific features available only in video, visual retrieval shares similar characteristics and similar processing flows for both image and video data.

2.3 Characteristics of content based visual information retrieval

Digitized images consist purely of arrays of pixel intensities, with no inherent meaning, and in this sense image databases are essentially unstructured collections of documents. Therefore one of the key issues with image and video retrieval is the need to extract structure from the raw data such as recognising the presence of particular shapes, texture or colours. This is not the case in the retrieval of textual information where the raw material has already been structured by the author [25].

Since the goal of CBVIR is to facilitate efficient user access to multimedia, cataloging and indexing of data depends on the information needs of the users. Access to content from a multimedia repository might involve searching for an image or video clip depicting a specific object or scene, or evoking a particular mood, or containing a specific texture or pattern. The content users target in their searches was categorised in [26] as follows:

- ◇ a particular combination of colour, texture or shape features (e.g. sky, grass)
- ◇ an object or arrangement of specific objects (e.g. chairs around a table)
- ◇ a particular type of event (e.g. news conference)
- ◇ of named individuals, locations, or events (e.g. the president's visit abroad)
- ◇ subjective emotions one might associate with the content (e.g. surprise)
- ◇ metadata such as who created the content, where and when or what it represents.

According to their capabilities of retrieving specific content attributes, CBVIR systems can be classified into three levels of increasing complexity [26]:

- ◇ *Level 1*, characterised by retrieval based on primitive features such as colour, texture, shape or the spatial location of image elements. Generally, this level of retrieval is suited to querying by example (i.e. "find more pictures like this").

2. Content based visual information retrieval

- ◇ *Level 2*, which requires some degree of logical inference about the content such as specific objects or events. Prior knowledge regarding the structure of features associated with a given individual object or event is necessary in order for the system to perform retrieval tasks of this level. According to [27] most user searches fall within this level of retrieval.
- ◇ *Level 3*, defines retrieval by abstract attributes specific to high-level reasoning such as, named events and activities, or subjective emotions associated with the content. The link between the visual content and abstract concepts relies on complex reasoning and subjective judgement.

The above introduced classification is useful in illustrating the strengths and limitations of different retrieval techniques. As pointed out by many authors there is a gap between levels 1 and 2 of retrieval, termed in the research literature as, the semantic gap. This gap is not exclusive to CBVIR but present in any type of processing that aims at emulating human reasoning capabilities.

In the early years of visual retrieval, images were the main data form of retrieval. Recently, video has become an important medium of retrieval and poses its own challenges due to its additional complexity and volume of data. Video data brings additional primitive features, such as those related to motion and sound. However video indexing demands comparably more complex computational resources than image processing. Inside a video sequence the presentation is organised into a number of distinct scenes, each of which can be further broken down into individual shots depicting a single view, conversation or action. A common way of organising video for retrieval is into storyboards of still images, termed keyframes, representing each scene. Another modality, described sometimes as video skimming, consists of replacing keyframes with short clips each capturing the essentials of a video sequence [28].

In the context of the aforementioned retrieval classification, video features enhance the descriptive ability of the query. For level 1, queries can include motion additional to image features such as colour, texture or shape, while on level 2 can describe types of actions. However the difference between video and still image queries is insignificant at level 3 of retrieval. Other cues which it is possible to exploit in the retrieval of video are the soundtrack and text when they appear in the sequence.

2.3.1 Visual search

Since in information retrieval the process of query formulation and document matching is inherently inaccurate the result of a search is not a unique and exact match, but a list of candidate documents which satisfy, at least partially, the constraints of a given query. The ultimate goal of a search is to minimise the number of such candidates without omitting items of interest.

The information utilised in the search is embedded into the video content itself, thus in order to make the search possible this information needs to be extracted. The extraction process can be automatic, manual or a combination of both. Machines can easily handle large amounts of data but unfortunately lack the capacity of determining semantic notions into the data. Therefore, most extraction tasks involve some sort of human interaction.

Although it does not involve visual attributes, text based filtering, where available, can help in reducing the search space before engaging retrieval on the visual features. Textual descriptions associated with the visual content can filter out irrelevant documents, include potential relevant document in the search, and/or direct users towards a successful search strategy. Other non-visual attributes derived from the content, such as the number of frames in the video sequence or number of shots can also provide useful information during the search.

Query by image content is one of the most popular approaches to query formulation [1]. Queries are formulated by using sample images or videos, rough sketches or component features of images (outline of objects, colour, texture, shape, etc). The data flow of query-by-content is illustrated in Figure 2.1.

Another approach is iconic search, where icons represent real-world objects or entities (e.g. textures, shapes) [29]. The icons are arranged into relational or hierarchical classes and queries are formulated by selecting the appropriate icons. However, iconic queries tend to have a rigid structure since only the icons provided can be utilised.

2.3.2 Visual browsing

The result of a search is a list of candidate images and/or video clips. The presentation of video results should represent a summary of the video's content, such that a user can quickly

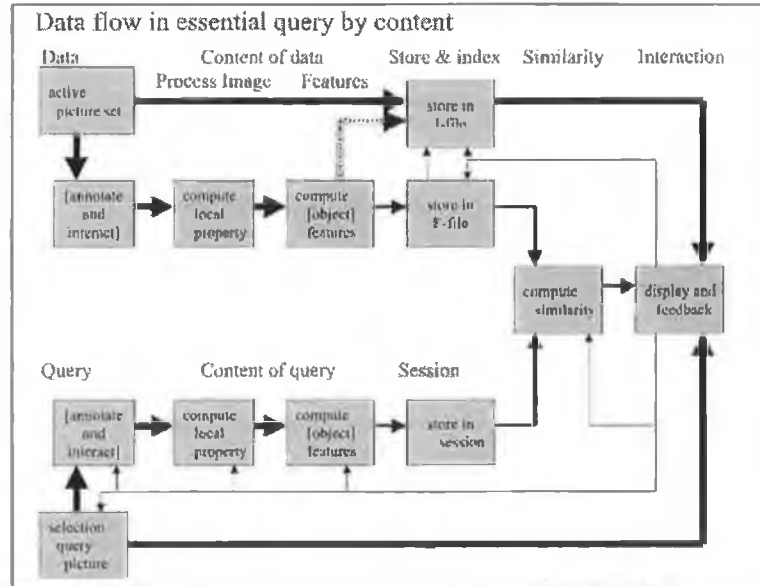


Figure 2.1: Query by example data-flow scheme, as depicted in [1]

understand the content and browse through the list without actually playing each video clip. Keyframes are the most common summarisation technique for video content [30]. However, summaries are dependant on the video category, a good summary of a soccer match requires a different summarisation approach than a movie summary. Therefore, different forms of visual summaries are required for different video categories.

Navigation based searching [25, 31, 32] is an alternative and sometimes a complementary approach to query-by-content. This approach is well suited for a scenario where a user's information need (i.e. a specific query) develops in the course of and as a result of the interaction with the image collection. The crucial aspect is to organise the content into a browsable structure, which is particularly challenging for dynamic collections.

2.3.3 User interaction in retrieval

Perhaps the most critical aspect of any retrieval system is the ability to effectively support users in expressing their search needs accurately and easily. The most appealing paradigm in visual query formulation is query-by-example, which provides a sample of the desired output the system should retrieve. However, an image example is not always easily available to the user. In alternative query formulation approaches, users can select from a catalogue

[5] of colours, texture and shapes, or by sketching the desired object on the screen [26]. However these methods prove cumbersome and generally largely ineffective.

User interaction is particularly useful in refining the search through the relevance feedback mechanism. Through this mechanism the system updates the parameter space, feature space or classification space to reflect the relevant and irrelevant examples provided by the user. Although users may not always formulate queries that accurately reflect their search needs, they can normally judge whether the results returned by the system match their interests. The simplest relevance feedback method introduced in [33] attempts to move the query vector toward relevant examples and away from the irrelevant ones. An overview of relevance feedback is provided in Chapter 5.

User interaction is not limited to relevance feedback. Interaction is especially demanded in the context of high-level feature extraction, usually related to abstract semantic concepts, which typically require fully manual or at least semi-automatic extraction. As well, various other parameters can be explicitly or implicitly selected by the user by other means than providing feedback through image examples.

Interaction is supported through graphical user interfaces. Designing interfaces for visual retrieval is a complex task. Particularly sensitive aspects of design are related to stimulating and collecting feedback from users and displaying the retrieved output. Various interfaces for search and browsing of video have been proposed in the literature supporting different interactions modes, content genre and hardware devices [34, 35]

2.4 Visual features: color, shape, and texture

Visual content can be modeled as a hierarchy of abstractions [36], where the raw pixels are the base level. On the next level groups of pixels define features such as edges, lines, colour and texture regions. These layers can be further combined into objects and attributes, until reaching the highest level which defines human concepts based on objects and relationships among them. Although automatic detection of low-level features is achievable most objects, attribute values and high-level concepts cannot be extracted accurately by automatic methods. Semi-automatic methods and textual annotations are required in such cases. In the following we briefly introduce the visual features used in content based retrieval. A detailed presentation of these descriptors is given in Chapter 3.

2.4.1 Colour retrieval

Colour is one of the most widely used visual features since it is relatively robust to background complication and independent of image size and orientation. Retrieval based on colour similarity has been extensively reported in the research literature. Although many colour representations have been proposed most of them are variations on the same basic representation schemes.

The colour histogram is the most commonly used colour descriptor for image retrieval [30]. Histograms represent the joint probability of occurrence for the three colour components of an image (e.g. red, green and blue for the RGB colour-space). A popular matching technique, histogram intersection, was introduced in [37]. An extension of histogram intersection to include distances between similar, but not identical colours was proposed in [5], while in [38] the cumulative histogram was developed in order to reduce the sensitivity to noise.

The issue of histogram spatiality are approached in [39] by adding an element of spatial matching and region-based grouping in [40]. Colour coherence vectors are proposed in [41] as a histogram refinement that enhances the spatial coherence. Correlograms are introduced in [42] as another colour descriptor that incorporates spatial information.

Several others colour representations have been developed for image retrieval. The use of colour moments is proposed in [38] to deal with the presence of noise in images. In this case a weighted Euclidean distance was used to calculate the colour similarity. A wide variety of photometric colour invariants for image retrieval were derived in [43].

2.4.2 Shape retrieval

Depending on the application domain some retrieval systems may require shape representations that are invariant to translation, rotation, and scaling. Generally, for retrieval, shape representation should allow robust matching even in the presence of considerable deformations. Shape representation is a well researched field of computer vision, however many sophisticated models developed in this field lack the robustness demanded by retrieval [1]. However, in interactive retrieval human judgment accommodates for less accurate matching, allowing trade-offs in favour of robustness and computational efficiency.

2. Content based visual information retrieval

Usually shape representations are divided in two categories, boundary-based and region-based, with Fourier descriptors and moment invariants as the most commonly used descriptors of each category [30]. Fourier descriptors encode the boundary of a shape according to the Fourier transform. Retrieval based on these descriptors has been proposed in [44, 45] while their robustness to noise and invariant geometric transformation was investigated in [46].

As suggested by the name, the moment invariants do not vary with transformations such as translation, rotation and scaling. The first seven moments were introduced in [47] while a more elaborated approach was developed in [48]. Retrieval approaches that incorporate moment invariant descriptors are reported in [49, 50].

Other shape descriptors used in retrieval include the finite element method [51], turning angles [52] and curvature scale space methods [53]. A comparative study of various shape descriptors was reported in [54]. According to the findings' the retrieval performance of different shape descriptors depends on the choice of database. However combined representations consistently outperformed the simple representations.

2.4.3 Texture retrieval

Texture is a natural property of all surfaces and is characterised visually through patterns of variation of intensity and colour. Texture analysis is a challenging processing task in computer vision as an image may contain different textures at varying scales. While retrieval by texture similarity may seem of limited practicality, the ability to match texture can be useful to distinguish between areas of images with similar colour [26].

A variety of techniques have been proposed to compute texture similarity. Early approaches are based on statistical moments such as the co-occurrence matrix, contrast and entropy [3]. A set of six texture proprieties derived from psychological studies in human visual perception is introduced in [55]. This set of features is visually meaningful therefore it brings an additional advantage for interactive retrieval. A similar set of texture features has been derived in [56]

Texture representations based on wavelet transforms were investigated in [57, 58]. In [9] texture statistics extracted from the wavelet sub-bands are used for image retrieval. Other

alternative methods of texture analysis for retrieval include Gabor filters [59] and fractals [60]. A texture dictionary is developed in [61].

Evaluation studies comparing the classification performance of various texture descriptors have been reported in the literature. In [62] a comparison of texture representations by various wavelet and Gabor transforms found that the Gabor transform best matches human perception from among the tested descriptors. A more recent evaluation [63] on Tamura and Gabor texture features concluded that retrieval performance increases when using combinations of such features.

2.5 Object-based retrieval

Object detection and recognition is an important aspect of content based visual information retrieval. Identifying image regions that correspond to objects allows for complex inferences that attempt to bridge the semantic gap based on recognising real objects from within images and video content. This could facilitate more effective query formulation and searching for visual content that contains objects of interest.

Retrieval based on the presence or absence of given objects has received some attention in the literature, but its potential is far from being comprehensively studied. However, detection of all possible object types is generally considered a computationally infeasible problem [64]. Generally there are three classes of techniques for object identification for subsequent retrieval:

- ◇ *Graph decomposition* where an image is decomposed into a tree or graph of constituent sub-regions and then objects are detected by matching parts of this graph to a predefined object graph or object query. In this approach an object is detected by identifying first its sub-component parts and the relationships among them.
- ◇ *Template matching* where a correspondence measure is computed between regions of an image and a model template of the desired object. A number of object templates are built and possible matches against image regions are extracted from the input image.

2. Content based visual information retrieval

- ◊ *Motion detection* of objects based on the assumption that foreground objects will exhibit a different type of motion as the rest of the image, often the dominant motion in the scene.

Attempts at using objects for retrieval have been reported for a decade [65], but only with recent technology has object-based functionality been enabled for video. We briefly mention here some of the object-based retrieval work reported in the research literature.

Recent approaches to object modeling attempt to find recurring patterns in images that contain objects of interest. The approach taken in [66] focuses on identifying patterns of feature clusters reoccurring in a set of training images while in [67] objects are modeled as colour adjacency subgraphs.

In [68] images are segmented into a set of regions termed as “objects” although not in the semantic sense. An evaluation of the approach is carried out on an object retrieval system using a cartoon video. The authors demonstrate that objects can be located in video based on a user query specified as a set of regions. Similar work is reported in [69] where arbitrary-shaped objects are located based on shapes and shape deformations over time.

A different approach is taken in [70, 71] where an object segmented by the user during query formulation is then matched and highlighted against similar objects appearing in keyframes. The object’s model is improved by using contiguous frames within a shot to estimate changes in viewpoint illumination and occlusions. The approach is illustrated on a set of movies where the search for a given object is limited to within the movie content.

An approach based on motion representation and object tracking without actually performing object segmentation is described in [72]. Object retrieval is achieved by locating objects which have a similar trajectory to a given object in the query clip. In [73] video frames are automatically segmented into regions based on colour and texture. Then the largest blobs in the frames are termed objects and tracked through the video sequence. Retrieval is performed using a query video clip to find video sequences similar in terms of object motion, edges, texture and colours.

An approach that combines text, image and object based searching into an iterative video shot retrieval system is presented in [64]. In this approach object segmentation is achieved

by working in a closed domain of animated cartoons. Extensive experimentation is carried out to investigate the interactive use of the system in a controlled retrieval exercise with real users.

As with any other modality available in information retrieval, object retrieval works best on certain search types. Clearly the best overall approach is to facilitate as many modalities as possible and let the user decide on the best search strategy which is possibly some combination of them.

2.6 Evaluation of visual retrieval systems

Since image and video retrieval is basically a tool that supports users in meeting their informational needs the assessment of performance should derive from direct human evaluations. Evaluation of retrieval is typically carried out on a common set of tasks and test data with accompanying ground truth. Performance is generally measured by various metrics that capture the overall efficiency of the retrieval process generally related to precision and recall. Other measured values can include speed of convergence to the target ranking for interactive systems, user satisfaction, and utilisation statistics. Generally, the aim of benchmarking is not just the evaluation of performance for a particular algorithm, but also to obtain metrics comparable between algorithms.

TRECVID [4] is perhaps the most prominent evaluation effort in recent years bringing together private industry and academic research in assembling a common set of test data and retrieval tasks. Various aspects of image and video processing and retrieval tasks are benchmarked within this project according to a commonly agreed evaluation protocol. The test collections include many features of video data such as speech transcripts, closed caption, metadata, shot boundaries, keyframes and automatically extracted features. The set of retrieval tasks covers fully-automatic, manually-assisted and interactive searching.

The VIPER group (Visual Information Processing for Enhanced Retrieval) based at the University of Geneva has been associated with several benchmarking efforts including query by example and image browsing evaluations, and also evaluation of retrieval markup languages. The group has been involved in the Benchathlon event [74] (Benchmark for Image

Retrieval using Distributed Systems over the Internet), an initial step towards a standardized benchmark, aimed at developing metrics, test collections and systematic comparison between retrieval systems.

Another evaluation framework was proposed in the IAPR (International Association for Pattern Recognition) TC-21 Benchmark [75] consisting of a set of still natural images, a representative set of queries and ground truth associated with these queries, and a set of recommended performance metrics. The envisaged objective of this evaluation framework is the retrieval of semantic content.

Some other initiatives related to the area of image and video retrieval focus on the evaluation of content extraction tools and only tangentially on retrieval itself. The Berkeley Segmentation Dataset and Benchmark [76] consists of 12,000 hand-labeled segmentations of 1,000 images from 30 human subjects. A method for measuring how well an automatically generated segmentation matched the ground-truth segmentation is provided together with the data set.

The MUSCLE Network of Excellence (Multimedia Understanding through Semantics, Computation and Learning) develops systems and methodologies for automatically extracting semantic information from multimedia data. A similar evaluation framework for benchmarking of feature extraction algorithms, and image and video content-based retrieval systems has been developed in the SCHEMA Network of Excellence and COST projects.

2.7 Selected visual retrieval systems

This section presents a brief overview of the major content based image and video retrieval systems. It will cover QBIC, Photobook, Netra, ImageRover, Virage, Webseek and VisualSeek, Mars, Blobworld, Istorama and PicSOM. However, this is not intended to be an exhaustive selection. A overview of these systems and their indexing features is presented in Table 2.1 derived from [77].

Blobworld [40] developed at University of California, Berkeley, segments a query image into regions (blobs) of uniform colour and texture. The segmentation process is driven by the Expectation Maximisation (EM) algorithm. The colour feature used is a 218 bin histogram in the Lab colour space. The shape features are taken as area, eccentricity and

2. Content based visual information retrieval

orientation of the blobs. Texture is represented by the mean contrast and anisotropy over the blobs. The user selects one blob from the image and query-by-example is performed based on this selected blob. In order to facilitate better understanding of the query results the retrieved images are presented together with their segmented representation.

MARS - Multimedia Analysis and Retrieval System [11, 30, 78, 46], developed at University of Illinois at Urbana-Champaign, allows queries on combinations of image features and textual annotations. Images are described by colour, texture, shape and layout features. Colour is represented using an 8 x 8 2D histogram over the HS coordinates in the HSV colour space. Texture is described by coarseness and directionality histograms complemented by a scalar value defining the image contrast. The shape information is encoded using Fourier Descriptors (FD). Colour and texture layout is extracted by dividing the image into 5 x 5 subimages. MARS was the first content-based image retrieval system implementing relevance feedback.

Photobook [7], developed at Massachusetts Institute of Technology, proposes a “semantic-preserving image compression” approach to image retrieval. In this approach specific representation models are constructed for each class of image content. Three representations are implemented for interactive retrieval: faces, 2D shapes and texture images. Faces and shapes are modelled as projections onto the eigenvectors of the covariance matrixes obtained from offline training on a set of face and respective shape prototypes. Texture is modelled as a 2D discrete random field along three orthogonal coordinates: periodicity, directionality and randomness.

QBIC - Query by Image Content [5, 49], an image retrieval system designed by IBM allows querying by keywords and visual features. Colour, texture and shape features are extracted for images or objects. The colour features are the average colour vector in the RGB, YIQ, Lab or Munsell colour spaces and the 256 dimensional RGB colour histogram. Area, circularity, eccentricity and moment invariants are the features used to describe and match shapes. The texture features used are coarseness, contrast, and directionality. Objects are extracted by using a semi-automatic segmentation tool based on an enhanced flood-fill technique. The world-famous Hermitage Museum employed a variant of the QBIC system to provide image search on its collection of paintings.

VIR - Visual Information Retrieval Image Engine [6] produced by Virage, is designed as a series of independent modules, termed “primitive”, which indicates a feature’s type,

computation and matching distance. The system provides a set of universal primitives, such as global colour, local colour, texture and shapes. The data types implemented in the system are: global values and histogram, local values and histograms, and graphs. The predefined set of universal primitives can be extended with domain specific primitives when developing an application. A set of GUI tools are provided for the development of user interfaces allowing image query, weights adjustment, keyword inclusion and query-by-sketch. Users can construct queries by using various combinations of primitives.

WebSeek [8, 9] is an image and video catalog and search tool for the world wide web, developed at Columbia University. Visual material is automatically collected from the web, analysed and indexed into subject categories using an extensible subject taxonomy. Text and colour features are used to index the retrieved items. Colour is represented by a normalised 166-bin histogram in the HSV colour space. The system allows users to modify an image colour histogram before reiterating the search.

Istorama [79], developed at Informatics and Telematics Institute, allows users to perform queries by example on image regions resulting from unsupervised segmentation. The regions resulting from the segmentation process are described by their colour, size and spatial location within the image. A weighted combination of features is used for retrieval. The user can put emphasis on a specific feature by manually adjusting the feature weighting accordingly. An extension to Istorama is the SCHEMA reference system [80] which includes additional segmentation algorithms and encodes the region features into MPEG-7 XML format.

ImageRover [81] developed at Boston University uses text and visual features for content-based search of a web image database. Textual analysis is based on the latent semantic indexing (LSI) of the HTML document in which an image appears. The visual features are described by colour and texture orientation histograms. The user initiates a search by describing the desired images with keywords. Later visual and textual cues are used to refine the query into a relevance feedback loop.

Netra [10] is an image retrieval system developed at the University of California Santa Barbara. It segments images into regions of homogeneous colour and represents each segment in terms of colour, texture, shape and spatial location features. The segmentation is done offline using an edge flow segmentation technique.

WebSeer [82], developed at University of Chicago, separates web images into photographs and graphics through a set of colour tests. Images classed as photographs undergo a face detection procedure. Keywords are extracted from the textual information (file name, caption, links) on the web page where the image is located.

PicSOM [83, 84], developed by the Laboratory of Computer and Information Science at Helsinki University of Technology, is an image browsing system based on Self-Organising Maps (SOM). The SOM organises images into a two-dimensional grid where similar images are adjacently located. Images are described by their average RGB colour, texture features and Fourier-based shape descriptors. The Tree Structured Self-Organising Map (TS-SOM) algorithm represents the image data base into a hierarchical structure for each feature type used. Image queries are performed through a Web interface. The query vector is refined through relevance feedback as the system exposes more images to the user.

Físchlár [85], developed by Centre for Digital Video Processing at Dublin City University, is an online video capture and retrieval system with shot level browsing and playback. The system implements a number of browsing strategies: sequential browsing by time-line, slide show browsing, hierarchial browsing and overview/detail browsing. The *Físchlár* News version of the system allows browsing by news stories. The indexing of video content is done based on low-level visual and audio features, semantic features (news story segmentation and face detection) and textual metadata extracted from the associated teletext content.

ANSES [13], developed by the Multimedia and Information Systems Team at Imperial College London combines video scene change, text segmentation and summarization to build an automatic news summarization and extraction system. News stories are identified, extracted from the video, and summarized in a short paragraph.

Informedia [12], developed at Carnegie Mellon University, is one of the earliest online digital video libraries. A first version of the system targeted as an integration framework for speech image and text was developed around automatic speech recognition (ASR) and novel browsing methods. The current version focuses on summarisation and retrieval of TV news content. It incorporates temporal and geographical location obtained from ASR. Browsing for content of interest can be done via a map of the world and a timeline bar. Face matching is incorporated in the indexing and retrieval process.

CueVideo [86], developed at IBM's Almaden Research centre, provides tools for video analysis and segmentation, visualization and summarization techniques, spoken document

2. Content based visual information retrieval

retrieval and cross-modal indexing of audio/video, related slides and text material. It incorporates sophisticated speech recognition technology for automated audio to text extraction (ASR).

*Video Google*¹ is a free Google video upload and retrieval service. Users can search and play videos directly from Google Video, as well as download video files. Although the indexing and retrieval of videos is based only on textual annotations, this system is included here in order to illustrate that successful video retrieval systems can be implemented based on simple textual annotation.

The large rise in the volume of digital images has led to the development of many visual information retrieval systems. The majority of these systems are research prototypes which usually explore a particular retrieval approach or set of visual features. Colour, shape and texture are well represented across research systems presented above. Although most systems listed in Table 2.1 utilize these features, it is not apparent whether there are significant differences in the efficacy among similar matching features, and if so, what these differences are.

Early content-based retrieval systems were designed for retrieval of images while those developed in the last 5 years target retrieval of video content, and in particular highly structured video data such as news broadcast. These latter systems make extensive use of textual annotation (metadata and closed caption) and less of the visual features to the extent that Video Google, perhaps the most commercially successful system to the date, exclusively uses text.

Only very few systems attempt to bridge the semantic gap by image segmentation. Notably, Blobworld and Istorama allow users to view the results of the segmentation of both the query image and returned results to highlight how the segmented features have influenced the retrieval results. The majority of the systems provide very little indication as to why certain images have been returned. This lack of explanation is likely to have a negative impact on users perception of the systems retrieval effectiveness since not all users are aware of the gap between low-level features and semantic concepts.

As pointed by many authors retrieval by texture does not always provide enough discrimination unless the collection contains a large number of images with a dominant texture.

¹<http://video.google.com/>

Retrieval on shape is even more unreliable since shape is affected by changes in camera perspective, object motion or position. Most of these systems complement the visual features with textual annotations and additional metadata.

2.8 Discussion

The increasing amount of video data available today requires the development of efficient technologies for its management. There are two approaches to managing digital video: using manually inserted annotations and metadata, and automatically processing video by deriving content descriptors [24]. The former approach is prevalent nowadays in archives although it is slowly incorporating elements of automation as they reach technological maturity. The later approach is still a relatively recent development. In the years since digital technology became ubiquitous, a variety of approaches and modalities for automatic content description have emerged, although they are still far from achieving their full potential.

The automatic analysis of video content raises difficult technical challenges. The complexity of automatic analysis of video data is demonstrated already by the difficulties that originate with processing of still images [24]. Although several content-based image retrieval systems achieve notable results, the performance of such systems is often based on a rigorous selection in the type of manageable content [24]. Applications that are able to operate on generic content often build on semi-automatic analysis since automatic processing cannot provide the required robustness.

Automatic analysis of video is considerably more complicated and less feasible than the analysis of still images. Although the present state of the art in the processing and retrieval of visual content is largely confined to automatic scene detection, promising new directions have been opened. Object based retrieval which allows users to manipulate video objects as part of their searching and browsing interaction is one of these directions.

The ability to extract relevant features from visual data is conditioned by the amount of a priori information available in specific application scenarios. For some restricted applications such as video surveillance it is possible to achieve fully automated processing; however for the large majority of applications only some portion of the analysis tasks can be performed entirely automatically. The type of video analysis techniques used in applications

2. Content based visual information retrieval

depends not only on the choice of processing tools but also on the role the user interaction plays in the overall process.

Powerful video analysis systems can be obtained only by combining the use of the best automatic analysis tools and user interaction. Interaction allows further control of the analysis process and refinement of results. User interaction can be regarded as a complement to overcoming the difficult processing tasks rather than a substitute for poor automatic analysis.

Interaction is especially required in the context of high-level feature extraction, usually related to abstract semantic concepts, which typically require fully manual or at least semi-automatic extraction. In many cases interaction may serve not only feature selection but also to refine automatic processing. User interaction can be employed in off-line tasks such as the initial setting of various system parameters, but as also in providing real-time guidance to the processing such as marking or selecting semantically relevant objects.

System	QBIC	Informedia	Photobook	VIR	WebSeer	MARS	WebSeek	ImageRover	Netra	Blobworld	PicSOM	CueVideo	Istorama	Fischlär	ANSES	Video Google
Starting year	1993	1994		1996			1997				1999		2001		2003	2005
Keywords (annotations, captions)	*	*		*	*	*	*	*		*		*	*	*	*	*
Face detection		*	*		*									*		
Layout	*			*		*	*	*	*				*			
Colour	Dominant colours	*		*			*				*	*	*	*		
	Global histogram	*	*	*	*	*		*			*		*	*	*	
	Region histogram								*	*	*		*			
	Block histogram			*		*		*						*		
	Coherence vector															
Shape	Eigen image		*													
	Bounding box/ellipse	*	*	*			*			*			*			
	Curvature scale space										*					
	Elastic models			*					*							
	Fourier descriptors					*										
Texture	Template matching	*														
	Edge statistics		*	*				*			*					
	Local binary patterns			*												
	Random fields															
	Tamura features & variations	*		*		*		*		*						
	Wavelet/Fourier transform					*			*							

Table 2.1: Summary of selected retrieval systems and their indexing features

Chapter 3

Visual content descriptors for image and video retrieval

Primitive image features such as colour, shape, and texture are the underlying medium of content-based indexing and retrieval. These features are potentially an extremely valuable source of information, but their value is limited unless they can be effectively extracted and retrieved. Higher semantics may be inferred combining and interpreting these features for meaningful searching and browsing in image and video collections. Thus accurate representations of image content in terms of a wide range of features is crucial. However some features are easier to extract than others. Colour features and texture features are generally straightforward while shape features require prior object segmentation. Most semantic features are difficult or impossible to extract with the state-of-the-art in feature detection. This chapter provides a detailed review the commonly used low-level visual content descriptors for colour, shape and texture.

3.1 Introduction

Video includes visual, audio and semantic content. Visual content can be described by features such as colour, shape, texture and their spatial relationships. The semantic content is generally obtained from textual annotation, caption extraction, automatic speech recognition or by inferring semantics based on visual content.

Invariance and discriminative power are the two main attributes of a descriptor. However there is tradeoff between these attributes since wide invariance often comes at the expense of ability to discriminate between essential differences. The visual content descriptor can be global or local. Global descriptors take account of a feature's distribution in the entire image, whereas local descriptors quantify a feature's distribution for a partition of the image. Image partitions can be obtained by dividing the image into a uniform grid of tiles of equal size and shape, or by extracting homogenous regions according to some criterion using image segmentation algorithms. Depending on the end-user application more complex segmentation algorithms could feasibly be used to partition images into semantically meaningful objects.

The intention here is not to provide an extensive presentation of all possible visual descriptors but to introduce the most widely used ones for informative purposes. Given the extent of available descriptors we keep only an overview approach throughout the chapter without going into details on the various aspects of each descriptor. This chapter also presents some of the commonly used measures for computing similarity between feature vectors, since representations alone do not have meaning without a method of assessing their similarity. Finally we move from image descriptors into the video domain by introducing the process of shot boundary segmentation.

In the following section we introduce visual descriptors for colour features. The descriptors for shape are presented 3.3 and those for texture in section 3.4. We felt that MPEG-7 descriptors should be presented in a separate section 3.5 since they are standardised modalities of representing colour, shape and texture features. Section 3.6 introduces the measures commonly used for similarity matching in content-based indexing and retrieval. A brief survey on video sequence segmentation is given in section 3.7. Section 3.8 summarises the content of the chapter.

3.2 Colour

Colour is the most extensively used visual feature in CBVIR [87, 88, 89, 90, 38, 37, 91]. It is relatively robust to background complications and independent of image size and orientation.

3.2.1 Colour space

There is no consensus within the research community on which colour space is most appropriate for retrieval. Perceptual uniformity is largely regarded as a desirable characteristic of a colour space used for image retrieval [89]. Some representative studies of colour perception on different colour spaces can be found in [92, 93, 94]. A few of the most important colour spaces are briefly summarized below.

The **RGB** colour space consists of the three additive primaries: red, green, and blue. Colour is produced by the additive combination of these three primary components. The RGB colour space is widely used for image display.

The **YUV** colour space is used in the PAL television standard. Y stands for the luminance component (the brightness) and U and V are the chrominance (colour) components. Most image and video storage formats store data in YUV format. The transformation between RGB space and YUV space can be found in [95]. The **YIQ** colour space previously used in the NTSC television standard is closely related to the YUV space.

The **CIE $L^*a^*b^*$** and **CIE $L^*u^*v^*$** colour spaces consist of a luminance component L (the brightness) and two chromatic components a and b or u and v . The CIE $L^*a^*b^*$ is designed for subtractive colorant mixtures while CIE $L^*u^*v^*$ is designed for additive colorant mixtures. Both spaces are device independent and perceptually uniform. The transformation of RGB space to CIE $L^*u^*v^*$ and CIE $L^*a^*b^*$ can be found in [88].

The **HSV** (closely related to HSL) colour space is an intuitive way of describing colour. The colour components are hue, saturation (lightness) and value (brightness). The hue is invariant to the changes in illumination. The transformation among RGB, HSV and HSL colour spaces can be found in [96].

The **Munsell Renotation System** [93] is very similar in concept to HSV describing colour in terms of hue, chroma and value based on a colour arrangement scheme. The Munsell Book of Colour displays a collection of colors laid out in rows and columns for different hue values. Each colour is identified numerically using different scales.

3.2.2 Colour moments

Colour moments have been used in many retrieval systems [49] and proved to be efficient and effective in representing the colour distribution of images [38]. The first three colour moments are in order: the mean μ , the variance σ and the skewness s . Their mathematical formulation is defined as:

$$\mu_i = \frac{1}{N} \sum_{j=1}^N f_{ij} \quad (3.2.1)$$

$$\sigma_i = \left(\frac{1}{N} \sum_{j=1}^N (f_{ij} - \mu_i)^2 \right)^{\frac{1}{2}} \quad (3.2.2)$$

$$s_i = \left(\frac{1}{N} \sum_{j=1}^N (f_{ij} - \mu_i)^3 \right)^{\frac{1}{3}} \quad (3.2.3)$$

where f_{ij} is the value of the i -th colour component of the image pixel j , and N is the number of pixels in the image.

The third-order moment brings an overall increase in retrieval performance when used jointly with the first two moments. However, sometimes it can prove sensitive to context changes and thus may reduce the overall performance. The colour moments are a very compact representation but may also have lower discrimination power compared to other colour features. Generally they are employed to narrow down the search space by rejecting highly dissimilar documents before more sophisticated techniques are used for retrieval.

3.2.3 Colour histogram

A colour histogram is an effective way of representing both the global and local distribution of colours within an image. Histograms are robust to translation and rotation and change only slowly with scale or occlusions. The discrimination power of a histogram increases with the number of bins. However having a large number of bins increases the computational cost and becomes unfeasible for indexing image databases. Moreover very fine bin quantization

does not always improve the retrieval performance. One way of reducing the number of bins is by down sampling the luminance component, the chrominance information being preserved. A more complex approach to bin selection is to use clustering methods to determine the k most representative colours (bins) for a given set of images. If applied over the entire image database this approach minimizes the likelihood of histogram bins that hold no or very few pixels. An alternative approach is to use the bins that have the largest pixels numbers thus selecting a small number of bins that capture the dominant image colours [97]. Usually such an approach enhances the performance of histogram matching since the less populated bins are likely to be noisy.

Computing histograms on each colour component separately may not provide sufficient discriminative power on large image collections. The joint histogram approach [90] further increases the histogram's discriminative capabilities. As histograms do not take into consideration the spatial distribution of colour, quite dissimilar images could present close matching histograms. This becomes an acute disadvantage in large scale databases. A way to incorporate spatial information is by partitioning the image into regions and subsequently extracting local histograms for these regions. The partitions can be simple rectangular tiles or more complex homogeneous regions or semantic objects.

3.2.4 Colour coherence vector

A different way of incorporating spatial information into the colour histogram is the colour coherence vector (CCV) [41]. In this approach each histogram bin will contain two values, one quantifying the number of coherent pixels in the bin and the other quantifying the number of incoherent pixels. A pixel is counted as coherent if it belongs to a large uniformly-colored region or else is counted as incoherent. The colour coherence vector can be defined as:

$$< (\alpha_1, \beta_1); (\alpha_2, \beta_2); (\alpha_i, \beta_i); \dots; (\alpha_N, \beta_N) > \quad (3.2.4)$$

where α_i is the coherence value in the i -th bin and β_i is the incoherence value. The histogram of the image in this case is:

$$< \alpha_1 + \beta_1; \alpha_2 + \beta_2; \alpha_i + \beta_i; \dots; \alpha_N + \beta_N > \quad (3.2.5)$$

It has been shown in [98] that for image retrieval the colour coherence vectors perform better than the colour histograms, especially for images having mainly uniform colour or texture regions.

3.2.5 Colour correlogram

The colour correlogram [42] is a modality for representing colour information with spatial layout, while retaining the advantages of histograms. A colour correlogram is a table indexed by colour pairs, where the k -th entry for pair (i, j) specifies the probability of finding a pixel of colour j at a distance k from a pixel of colour i in the image. The formal definition of colour correlograms is given as:

$$\gamma_{i,j}^{(k)} = \Pr_{p_1 \in I_{c(i)}, p_2 \in I} [p_2 \in I_{c(j)} \mid |p_1 - p_2| = k] \quad (3.2.6)$$

where I represents the entire set of image pixels and $I_{c(i)}$ represents the set of pixels whose colors are $c(i)$; $i, j \in \{1, 2, \dots, N\}$, $k \in \{1, 2, \dots, d\}$ and $|p_1 - p_2|$ is the distance between pixels p_1 and p_2 .

When all possible combinations of colour pairs are considered the colour correlogram becomes very large in size (O^2d). The colour autocorrelogram is an often used alternative to the correlogram since it captures only the spatial correlation between identical colors and thus reduces the dimension to $O(Nd)$. Compared to colour histograms the colour autocorrelogram provides better retrieval performance, but is also more computationally expensive due to its high dimensionality.

3.2.6 Invariant colour features

Colour representation can be affected by change of illumination, shadows and camera perspective. Most colour features provide only limited robustness to such environmental variations. Invariant colour features for content-based image retrieval are investigated in [43]

where a set of colour invariants is derived based on the Schafer model of object reflection. Shape and illumination invariant representation based on blue ratio vector $(r/b, g/b, 1)$ was proposed in [99] and surface geometry invariant features were introduced in [50].

Invariant colour features increase the robustness in the representation of images to illumination, scene geometry and viewing geometry changes, but may also lead to some loss of discrimination power among images.

3.3 Shape

Image retrieval by shape features of regions and objects has been investigated in many applications [100, 101, 102, 52]. Shape features are usually described after images have been segmented into regions or objects. Since robust and accurate image segmentation is difficult to achieve, shape-based image retrieval is limited to special applications where objects or regions are readily available. A good shape descriptor for image retrieval should satisfy several properties such as affine invariance, robustness, compactness, low computation complexity and perceptual similarity measurement [54].

Shape description methods can be classified into two main categories [103], boundary-based and region-based descriptors. The most representative boundary-based methods are: rectilinear shape analysis [102], polygonal approximation [51], finite elements models [104] and Fourier-based descriptors [105, 45, 106]. Region-based shape characterisation exploits the statistical moments of shape [47, 48].

3.3.1 Moment invariants

The moment invariants characterise the overall shape appearance. For an object or region R represented as a binary map, the central moments of order $p + q$ of its shape are defined as:

$$\mu_{p,q} = \sum_{(x,y) \in R} (x - x_c)^p (y - y_c)^q \quad (3.3.1)$$

where (x_c, y_c) is the centre of object or region and (x, y) are the coordinates of the contour points. The central moment normalised for scale invariance is $\eta_{p,q}$ [47]:

$$\eta_{p,q} = \frac{\mu_{p,q}}{\mu_{0,0}^\gamma} \quad (3.3.2)$$

$$\gamma = \frac{p+q+2}{2}$$

From the above definition a set of moments invariant to translation, rotation and scale can be derived as [47, 48]:

$$\begin{aligned} \phi_1 &= \mu_{2,0} + \mu_{0,2} \\ \phi_2 &= (\mu_{2,0} - \mu_{0,2})^2 + 4\mu_{1,1}^2 \\ \phi_3 &= (\mu_{3,0} - 3\mu_{1,2})^2 + (\mu_{0,3} - 3\mu_{2,1})^2 \\ \phi_4 &= (\mu_{3,0} + \mu_{1,2})^2 + (\mu_{0,3} + \mu_{2,1})^2 \\ \phi_5 &= (\mu_{3,0} - 3\mu_{1,2})(\mu_{3,0} + \mu_{1,2}) [(\mu_{3,0} + \mu_{1,2})^2 - 3(\mu_{0,3} + \mu_{2,1})^2] \\ &\quad + (\mu_{0,3} - 3\mu_{2,1})(\mu_{0,3} + \mu_{2,1}) [(\mu_{0,3} + \mu_{2,1})^2 - 3(\mu_{3,0} + \mu_{1,2})^2] \\ \phi_6 &= (\mu_{2,0} - \mu_{0,2}) [(\mu_{3,0} + \mu_{1,2})^2 - (\mu_{0,3} + \mu_{2,1})^2] + 4\mu_{1,1}(\mu_{3,0} + \mu_{1,2})(\mu_{0,3} + \mu_{2,1}) \\ \phi_7 &= (3\mu_{2,1} - \mu_{0,3})(\mu_{3,0} + \mu_{1,2}) [(\mu_{3,0} + \mu_{1,2})^2 - 3(\mu_{0,3} + \mu_{2,1})^2] \end{aligned} \quad (3.3.3)$$

Higher order moments capture less critical shape information and tend to be affected by noise, reason for which they are often discarded [103].

3.3.2 Compactness, eccentricity, and major axis orientation

Some simple region-based shape descriptors are: compactness, eccentricity and major axis orientation [107].

Compactness, also called circularity by some authors, is computed as:

$$\alpha = \frac{4\pi A}{P^2} \quad (3.3.4)$$

where A is the area and P is the perimeter of the shape. The value of α ranges between 0 (corresponding to a line segment) and 1 (corresponding to a perfect circle).

Eccentricity is the ratio of major and minor axes of a region. Major axis orientation also called shape direction, is a property associated with elongated regions only. The major axis orientation is the direction of the longer side of a minimum bounding rectangle. When the shape moments are known, the direction θ can be computed as:

$$\theta = \frac{1}{2} \tan^{-1} \left(\frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} \right) \quad (3.3.5)$$

3.3.3 Turning angles

In a boundary based approach, a shape contour can be represented as a closed sequence of successive boundary pixels (x_s, y_s) . The *turning function* or *turning angle* $\theta(s)$ measures the angle between two pairs of points on the shape contour. The turning angle, which is expressed as a function of the arc-length s between the points, can be defined as:

$$\begin{aligned} \theta(s) &= \tan^{-1} \left(\frac{y'_s}{x'_s} \right) \\ y'_s &= \frac{dy_s}{ds} \\ x'_s &= \frac{dx_s}{ds} \end{aligned} \quad (3.3.6)$$

Turning angle representations are dependent on the rotation of shape and the choice of the reference point. In order to compare the similarity of two shapes the minimum distance needs to be calculated over all possible shifts t and rotations ω . A shift in the reference point along the shapes's boundary results in a new turning function $\theta(s+t)$ and a shape rotation in a new turning function $\theta(s) + \omega$. The minimum distance between two shapes A and B can be expressed as:

$$d_p(A, B) = \left[\min_{\omega \in R, t \in [0,1]} \int_0^1 |\theta_A(s+t) - \theta_B(s) + \omega|^p ds \right]^{\frac{1}{p}} \quad (3.3.7)$$

The perimeter length of the shape is re-scaled to 1. The measure is invariant to rotation, translation and change of scale.

3.3.4 Fourier descriptors

The contour of a 2D shape can be represented by the Fourier transform of its boundary pixels [44]. Three types of contour representations can be defined for a closed sequence of successive boundary pixels (x_s, y_s) : *curvature*, *centroid distance*, and *complex coordinate function*.

The curvature $K(s)$ at a point s along the contour is defined as the rate of change in tangent direction of the contour:

$$K(s) = \frac{d}{ds}\theta(s) \quad (3.3.8)$$

where $\theta(s)$ is the turning function introduced in (3.3.6).

The centroid distance is defined as the distance function between boundary pixels and the centroid (x_c, y_c) of the object:

$$R(s) = \sqrt{(x_s - x_c)^2 + (y_s - y_c)^2} \quad (3.3.9)$$

The complex coordinate representation is obtained by representing contour pixels as complex numbers:

$$Z(s) = (x_s - x_c) + j(y_s - y_c) \quad (3.3.10)$$

The Fourier transforms of the above-introduced contour representations describe the shape as sets of complex coefficients in the frequency domain. The frequency domain representations are called Fourier descriptors. The general shape properties are described by lower frequency coefficients while higher frequency coefficients reflect shape details. Invariance to rotation (invariance to choice of the reference point) is obtained by taking only the amplitudes of the complex coefficients, discarding phase components. Scale invariance is achieved by normalising the amplitudes of the coefficients by the amplitude of the DC component or of the first non-zero coefficient. Invariance to translation is intrinsic since the description is obtained directly from contour points.

The curvature description in the frequency domain (Fourier descriptor of curvature) is:

$$f_K = [|F_1|, |F_2|, \dots, |F_{M/2}|] \quad (3.3.11)$$

The Fourier descriptor of the centroid distance is:

$$f_R = \left[\frac{|F_1|}{|F_0|}, \frac{|F_2|}{|F_0|}, \dots, \frac{|F_{M/2}|}{|F_0|} \right] \quad (3.3.12)$$

where F_0 is the amplitude of the DC component. Since the curvature and centroid distance are vectors of real numbers only the positive frequencies are considered in their description.

The Fourier descriptor of the complex coordinate is:

$$f_Z = \left[\frac{|F_{-(\frac{M}{2}-1)}|}{|F_1|}, \frac{|F_{-1}|}{|F_1|}, \frac{|F_2|}{|F_1|}, \dots, \frac{|F_{\frac{M}{2}}|}{|F_1|} \right] \quad (3.3.13)$$

where F_1 is the first non-zero frequency component. The DC component is not used since it is dependent on the position of the shape.

For comparison of two shapes, the Fourier descriptors of both shapes should have an equal number of coefficients, which means that both shapes should have the same contour length. This can be ensured by sampling each shape to an equal number of samples.

3.3.5 Curvature scale space

The Curvature scale space (CSS) is an approach to shape representation based on the curvature of the closed contour of an object [53]. The closed contour C defined by the coordinate function $C(s) = (x(s), y(s))$, with s as the arc-length parameter, is convolved with a Gaussian kernel ϕ_σ of width σ :

$$\begin{aligned} x_\sigma &= \int x(s) \phi_\sigma(t-s) dt \\ y_\sigma &= \int y(s) \phi_\sigma(t-s) dt \\ \phi_\sigma(t) &= \frac{1}{\sqrt{2\pi}\sigma} \exp \left[-\frac{t^2}{2\sigma^2} \right] \end{aligned} \quad (3.3.14)$$

For continuous increasing σ the resulting contour become smoother and the curvature zero-crossing points move along the contour until the contour is convex. When two zero-crossing points meet they annihilate each other. Matching two objects can be done by matching the points of annihilations in the (s, σ) plane.

3.4 Texture

Perceptually adequate descriptors of image texture are important cues for image retrieval. In CBVIR, texture descriptors [60, 61, 108] can be classified in to two classes of representation methods: structural and statistical. Structural representations describe texture by identifying structure primitives and placement rules. Morphological operators [109] and adjacency graphs [59] are the most common structural representations which prove effective in the description of very regular texture such as artificial texture patterns. Statistical descriptors, which characterise texture by the statistical distribution of the image intensity, are more frequently used and cover a large variety of representations such as: Fourier power spectra, co-occurrence matrices, shift-invariant principal component analysis (SPCA), Tamura features [110], Wold decomposition [56], Markov random fields [111], fractal models, multi-resolution Gabor filtering [112] and wavelet transforms [57, 113, 62, 114].

3.4.1 Tamura features

The Tamura features are a set of texture features derived from studies of human visual perception. This set of features [55] include coarseness, contrast, directionality, linelikeness, regularity, and roughness. Only the first three components of Tamura features are commonly used in image retrieval.

Coarseness is a measure of the granularity of the texture. The first step in obtaining the coarseness is computing the moving averages $A_k(x, y)$ on $2k \times 2k$ ($k = 0, 1, \dots, 5$) size windows at each pixel location (x, y) . The moving averages windows are expressed as:

$$A_k(x, y) = \frac{\sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \left(\sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} I(i, j) \right)}{2^{2k}} \quad (3.4.1)$$

3. Visual content descriptors for image and video retrieval

where $I(i, j)$ is the pixel intensity at location (i, j) .

For each pixel location the differences between pairs of non-overlapping moving averages in the horizontal and vertical directions are computed:

$$\begin{aligned} E_{k,h}(x, y) &= \left| A_k(x + 2^{k-1}, y) - A_k(x - 2^{k-1}, y) \right| \\ E_{k,v}(x, y) &= \left| A_k(x, y + 2^{k-1}) - A_k(x, y - 2^{k-1}) \right| \end{aligned} \quad (3.4.2)$$

The value of k that maximizes the pair differences E in either direction is used to set the optimal window size $S_{optimal}$ for each pixel:

$$S_{optimal}(x, y) = 2^k \quad (3.4.3)$$

Finally the texture coarseness is obtained by averaging $S_{optimal}$ over the entire image:

$$F_{crs} = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n S_{optimal}(i, j) \quad (3.4.4)$$

A single value representing the texture coarseness for an entire image may not be sufficient for images with multiple regions of different texture. The coarseness representation can be enhanced by using a histogram that captures the distribution of the $S_{optimal}$ values within the image.

Contrast captures the dynamic range of grey levels in an image, together with their distribution.

$$F_{con} = \frac{\sigma}{\alpha_4^{1/4}} \quad (3.4.5)$$

where α_4 is the kurtosis calculated by dividing the fourth moment about the mean intensity, μ_4 , by the variance squared, σ^4 :

$$\alpha_4 = \frac{\mu_4}{\sigma^4} = \frac{\sum_i \sum_j (I_{i,j} - \mu)^4}{N \sigma^4} \quad (3.4.6)$$

where μ is the mean intensity and $N = ij$ is the number of samples.

Directionality is obtained by convoluting the image with two 3×3 kernels and then computing a gradient vector for each pixel. The convolution kernels are:

$$\begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

The magnitude and angle of the gradient vector are defined as:

$$\begin{aligned} |\Delta_G| &= (|\Delta_H| + |\Delta_V|)/2 \\ \theta &= \tan^{-1}(\Delta_V/\Delta_H) + \pi/2 \end{aligned} \tag{3.4.7}$$

where Δ_H and Δ_V are the horizontal and vertical differences of the convolution.

The histogram of magnitudes $|\Delta_G|$ will exhibit strong peaks for highly directional images and will be relatively flat for images without strong orientation. The histogram is summarised into an overall directionality measure:

$$F_{dir} = \sum_p^{n_p} \sum_{\phi \in w_p} (\phi - \phi_p)^2 H_D(\phi) \tag{3.4.8}$$

In the above relation p are the peaks, n_p are the number of peaks in the histogram, w_p is the set of bins in the histogram $H_D(\phi)$ and ϕ_p is the bin that takes the peak value.

3.4.2 Wold features

The Wold features describe texture by three components: harmonic, evanescent and indeterministic [56]. These components correspond to periodicity, directionality and randomness of texture, a strong harmonic component indicating a periodic texture, highly directional textures exposing a strong evanescent component and unstructured textures being accompanied by a large indeterministic component.

The 2D Wold decomposition separates a random field $\{y(m, n), (m, n) \in Z^2\}$ into three mutually orthogonal components:

$$y(m, n) = u(m, n) + d(m, n) = u(m, n) + h(m, n) + e(m, n) \quad (3.4.9)$$

where $u(m, n)$ is the indeterministic component and $d(m, n)$ represents the deterministic component which can be further decomposed into an harmonic component $h(m, n)$ and an evanescent component $e(m, n)$. In the frequency domain the expression becomes:

$$F_y(\xi, \eta) = F_u(\xi, \eta) + F_d(\xi, \eta) = F_u(\xi, \eta) + F_h(\xi, \eta) + F_e(\xi, \eta) \quad (3.4.10)$$

where $F_y(\xi, \eta)$, $F_u(\xi, \eta)$, $F_d(\xi, \eta)$, $F_h(\xi, \eta)$, $F_e(\xi, \eta)$ are the spectral distribution functions of $y(m, n)$, $u(m, n)$, $d(m, n)$, $h(m, n)$ and $e(m, n)$.

The Wold components can be obtained by thresholding Fourier spectral magnitudes in the frequency domain.

3.4.3 Simultaneous auto-regressive (SAR) model

Markov random field (MRF) models have been successfully used for texture modeling. The SAR model [111] is one such Markov fields method that uses a reduced number of parameters. In this approach the intensity $I(x, y)$ of a pixel at location (x, y) is estimated as a linear combination of the neighboring pixel values $I(x', y')$ and an additive noise term $\varepsilon(x, y)$:

$$I(x, y) = \mu + \sum_{(x', y') \in D} \theta(x', y') I(x', y') + \varepsilon(x, y) \quad (3.4.11)$$

where μ is a bias value dependent on the average image intensity, D is a neighborhood of (x, y) , $\theta(x', y')$ are weighting factors associated with each of the neighboring pixels, and $\varepsilon(x, y)$ is an independent Gaussian random variable with zero mean and variance σ^2 . The parameter θ indicates the texture orientation and σ captures the texture granularity.

In order to obtain invariance to rotation the model's parameters are computed on a circular neighborhood D of various radii centered at each pixel (x, y) . For rotation invariance the intensity $I(x, y)$ at pixel (x, y) can be expressed as:

$$I(x, y) = \mu + \sum_{i=1}^p p\theta_i(x, y)l_i(x, y) + \varepsilon(x, y) \quad (3.4.12)$$

where p is the number of circular neighborhoods, usually $p = 2 \times l(x, y)$ and can be computed as:

$$l_i(x, y) = \frac{1}{8} + \sum_{(x', y') \in N_i} w_i(x', y') I(x', y') \quad (3.4.13)$$

where N_i is the i th circular neighborhood of the pixel at location (x, y) and $w_i(x', y')$ are the weights indicating the contribution of the pixel (x', y') in the i th circle.

To allow for different texture granularities the image can be represented by a multi-resolution Gaussian pyramid with low-pass filtering and sub-sampling applied at several successive levels. The SAR model is then applied to each level in the pyramid.

3.4.4 Gabor filter features

Gabor filters have been widely used for texture characterisation [112, 63]. A Gabor filter is a Gaussian envelope modulated by a sinusoidal plane wave. A bank of such filters at different scales and orientations can be applied to an image to extract a texture description. The scale of the filter is given by the standard deviation of the Gaussian envelope localising the texture to a specific size within the image.

A two dimensional Gabor function $g(x, y)$ is defined as:

$$g(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp \left[-\frac{1}{2} \left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} \right) + 2\pi j W x \right] \quad (3.4.14)$$

where W is the modulation frequency, σ_x^2 and σ_y^2 are the variance of the Gaussian envelopes along the x and y direction.

A set of Gabor filters for different scales and orientations can be obtained by appropriate dilation and rotation of this function:

$$\begin{aligned} g_{mn}(x, y) &= a^{-m} g(x', y') \\ x' &= a^{-m} (x \cos \theta + y \sin \theta) \\ y' &= a^{-m} (-x \sin \theta + y \cos \theta) \end{aligned} \quad (3.4.15)$$

where $a > 1$, $\theta = n\pi/K$, $n = 0, 1, \dots, K-1$, and $m = 0, 1, \dots, S-1$, with K and S being the number of orientations and scales. The scale factor a^{-m} ensures that energy is independent of m .

A Gabor wavelet transform is a Gabor filter bank containing a quasi-orthogonal subset of Gabor filters:

$$W_{mn}(x, y) = \int I(x, y) g_{mn}^*(x - x_1, y - y_1) dx_1 dy_1 \quad (3.4.16)$$

where g_{mn}^* indicates the complex conjugate. The mean μ_{mn} and the standard deviation σ_{mn} of the magnitude $|W_{mn}|$ can be used to represent the texture feature of a homogenous texture region, $f = [\mu_{00}, \sigma_{00}, \dots, \mu_{mn}, \sigma_{mn}, \dots, \mu_{s-1k-1}, \sigma_{s-1k-1}]$.

3.4.5 Wavelet transform features

The wavelet transform is another multi-resolution approach to texture analysis [113, 114]. In the wavelet approach, a signal is decomposed with a family of basis functions $\psi_{mn}(x)$ obtained through translation and dilation of a mother wavelet $\psi(x)$:

$$\psi_{mn}(x) = 2^{-m/2} \psi(2^{-m}x - n) \quad (3.4.17)$$

where m and n are dilatation and translation parameters. The signal is then represented as:

$$f(x) = \sum_{m,n} c_{m,n} \psi_{mn}(x) \quad (3.4.18)$$

The 2D wavelet transform involves recursive filtering and sub-sampling. At each level, the signal is decomposed into four frequency sub-bands, LL, LH, HL, and HH, where L stands for low frequency and H stands for high frequency. There are two major wavelet transforms used for texture analysis: the pyramid-structured wavelet transform (PWT) and the tree-structured wavelet transform (TWT). The PWT is used to extract the LL band. In addition the TWT is used to decompose other bands such as LH, HL or HH since for some textures the most important information appears in the middle frequency channels. The feature vectors can be constructed using the mean and standard deviation of the energy distribution for each sub-band at each level. Comparison of different wavelet transform features [62] shows the particular choice of wavelet filter is not critical for texture analysis.

3.5 MPEG-7 visual descriptors

The MPEG7 standard [115] formally named the Multimedia Content Description Interface, defines the syntax and semantics of video descriptions. Previous MPEG (Moving Pictures Expert Group) standards dealt with video storage (MPEG-1), video for digital television (MPEG-2) and higher compression of digital video and support for object-based encoding (MPEG-4) whereas the MPEG7 standard concerns the description of video. The MPEG7 descriptors allow for interoperability between video retrieval systems by providing a common interface between video indexing tools. The MPEG7 System tools provide an XML mechanism to encode the descriptors in compact binary representations.

3.5.1 MPEG-7 Colour descriptors

MPEG-7 Colour Layout describes the spatial layout of colours in an image or a region based on a 8×8 spatial grid. The Discrete Cosine Transform (DCT) is performed for each cell in the grid and the most significant DCT coefficients (i.e. the DC coefficient and some of the low frequency AC coefficients) are stored.

MPEG-7 Dominant Colour describes an image or an arbitrary shaped region with a small number of representative colours. The number of dominant colors can vary from image to image, with a maximum of eight dominant colours being sufficient to represent an image. The description indicates the fraction of the image represented by each colour and its

variance. This descriptor is a very compact description of the colour distribution in the image. It was found to be more suitable for representing colours of image regions or objects where a limited number of colours may be sufficient [116].

MPEG-7 Scalable Colour is a colour histogram in the HSV colourspace encoded by a Haar transform. The histogram is uniformly quantised into 256 bins (16 hue \times 4 saturation \times 4 brightness). Its binary representation is scalable in terms of bin numbers and bit representation accuracy. Inversion of the Haar transform is not necessary for similarity matching which can be performed in the transform domain.

MPEG-7 Colour structure is a histogram-based feature that captures both colour content and information about the spatial arrangement of the colours. The representation is calculated by counting how many times a given colour is present in a 8×8 window sliding across the image.

3.5.2 MPEG-7 Shape descriptors

MPEG-7 Region Shape describes the shape of arbitrary objects. The shape of an object may consist of either a single connected region or a set of disjoint regions, as well as some holes in the object. The region-based shape descriptor utilizes a set of ART (Angular Radial Transform) coefficients. ART is a 2-D complex transform defined on a unit disk in polar coordinates.

MPEG-7 Contour Shape describes a closed contour of a 2D object or region in an image or video sequence. The representation is based on the Curvature Scale Space which describes the curvature zero crossing points. The shape is smoothed by filtering until it takes a convex hull then the highest peak and optionally up to 62 less prominent peaks of the Curvature Scale Space image are stored.

MPEG-7 Shape 3D allows 3-dimensional shapes to be described with a 3D mesh model.

3.5.3 MPEG-7 Texture descriptors

MPEG-7 Edge histogram counts the number of 5 different edge types (vertical, horizontal, 45 degrees, 135 degrees and non-directional edges) occurring in an image. The image is

divided into a 4×4 rectangular grid and the edges are counted for each cell of the grid. The histogram bins are normalised by the number of pixels in the source image.

MPEG-7 Texture Browsing characterises texture in terms of regularity, coarseness and directionality. The texture regularity is defined based on four values: irregular, slightly irregular, regular and highly regular. The coarseness (scale) of the texture can take four scales: fine, medium, coarse and very coarse. The directionality of the texture can be specified as non-directional or on 30 degree increments.

MPEG-7 Homogeneous texture describes texture by the mean and standard deviation of the image intensity obtained from a bank of 30 Gabor filters. The bank of filters is configured in 6 orientations of 30 degrees and 5 radial centre frequencies spaced in an octave scale. The obtained values are subjected to nonlinear scaling and quantization into 8-bits.

3.6 Similarity measures

Due to the inherent difficulties in formulating well defined queries, content based image and video retrieval very seldom depends on exact matching of feature vectors, but rather on similarity metrics and distances. Accordingly, the retrieval result is not a single image, but a list of images ranked by their similarities with the query image. There are many similarity measures proposed for image retrieval and a comprehensive analysis can be found in [1, 117, 118]. The performance of a retrieval system is affected by the similarity measure used. In this section, we will introduce some of the frequently used similarity measures.

3.6.1 Earth Mover distance

The Earth Mover distance is a general and flexible metric [119] derived from a dynamic programming approach to transportation optimization. From an information theory perspective the Earth Mover distance models the minimal cost paid to transform one distribution into the other. It allows for partial matches, and it can be applied to variable-length representations of distributions.

3.6.2 Minkowski form distance

The Minkowski-form distance L_p is used to calculate the distance between two images when each dimension of the image feature vector is independent. The distance is defined as:

$$d_{L_p}(i, j) = \left(\sum_k |i_k - j_k|^p \right)^{1/p} \quad (3.6.1)$$

For $p = 1$ the distance is known as the Manhattan distance (L_1 norm or city block distance), for $p = 2$ as the Euclidean distance (L_2 norm) and as the maximum distance between vector elements (L_∞ norm) when $p = \infty$. Minkowski distances are distance metrics when $p \geq 1$. Minkowski-form distance is the most widely used metric for image retrieval, being used as Euclidean distance to compute texture similarity in the MARS system [30], in Netra [10] for colour and shape similarity and as well in the Blobworld system [40]. Comparative studies [120] have shown L_1 to perform better than L_2 and both better than L_∞ for texture and colour classification.

For histogram intersection the L_1 distance was proposed as a measure of similarity between colour images [37]. The intersection of the two histograms is defined as:

$$d_{Int}(H_i, H_j) = \frac{\sum_k (H_i(k) - H_j(k))}{\sum_k H_i(k)} \quad (3.6.2)$$

Histogram intersection is robust to changes in image resolution, histogram size, occlusion, depth, and viewing point.

3.6.3 Fractional distances

Fractional distances [121] are Minkowski distances with $p \in [0, 1]$, however they are not a distance metric. Experimental studies [122] showed that optimal values for p are located between 0.25 and 0.75 depending on the feature and test collection.

3.6.4 Quadratic form distance

The Quadratic form distance, was introduced as an alternative to the Minkowski distance for comparison of colour histograms. Unlike the Minkowski distance this measure captures the perceptual similarities between pairs of histogram's bins. The distance is defined as:

$$d(H_i, H_j) = \sqrt{(H_i - H_j)^T A (H_i - H_j)} \quad (3.6.3)$$

where $A = [a_{ij}]$ is the cross-bin similarity matrix. Typically the entries a_{ij} in the similarity matrix A are chosen as either:

$$a_{ij} = 1 - \frac{d_{ij}}{d_{max}} \quad (3.6.4)$$

or,

$$a_{ij} = \exp \left[-\sigma \left(\frac{d_{ij}}{d_{max}} \right)^2 \right] \quad (3.6.5)$$

where d_{ij} is the Euclidean distance between the two colours i and j , d_{max} is the maximum of such distances and σ is a positive constant.

3.6.5 Mahalanobis distance

The Mahalanobis distance metric [123] takes into account the statistical correlation among the dimensions of the feature vector. The distance is defined as:

$$d(F_i, F_j) = \sqrt{(F_i - F_j)^T \sum^{-1} (F_i - F_j)} \quad (3.6.6)$$

where F_i and F_j are the feature vectors and \sum^{-1} is the covariance matrix of the feature vectors. When feature dimensions are independent the covariance matrix is diagonal and the measure can be simplified to:

$$d(F_i, F_j) = \sum_k \frac{(F_i(k) - F_j(k))^2}{\sigma_k^2} \quad (3.6.7)$$

where σ_k^2 is the variance of each component of F_j . In its simplified form the distance is referred as the normalised Euclidean distance.

3.6.6 Kullback-Leibler divergence

The Kullback-Leibler (KL) [118] is a directional distance that measures the average entropy of encoding one feature distribution by using another feature's distribution as codebook. The KL divergence is defined as:

$$d_{KL}(F_i, F_j) = \sum_k p_k(F_i) \log \frac{p_k(F_i)}{p_k(F_j)} \quad (3.6.8)$$

where $p_k(F_i)$ and $p_k(F_j)$ are the probability distributions of two feature vectors F_i and F_j .

3.6.7 Jeffrey divergence

The Jeffrey divergence (JD) measure is a variation of KL divergence with the added advantage of being symmetrical and fully defined when comparing two empirical distributions. The measure is expressed as:

$$d_{JD}(F_i, F_j) = \sum_k \left(p_k(F_i) \log \frac{p_k(F_i)}{\hat{p}_k} + p_k(F_j) \log \frac{p_k(F_j)}{\hat{p}_k} \right) \quad (3.6.9)$$

where:

$$\hat{p}_k = \frac{p_k(F_i) + p_k(F_j)}{2} \quad (3.6.10)$$

3.7 Video sequence segmentation

Generally, a video shot can be defined as “a single sequence of frames in motion pictures obtained by one camera without interruption” [124]. Improvements in video production software have increased the complexity of shot change effects available for video editing introducing transitional effects such as wipe, fade, or dissolve. The common occurring shot transitions are introduced below:

- ◇ A cut (hard cut) is an instantaneous transition from one scene to the next, and occurs over two frames
- ◇ A fade is a gradual transition between a scene and a constant image (fade out) or between a constant image and a scene (fade in).
- ◇ A dissolve is a gradual transition from one scene to another, where the old scene fades out as the new scene fades in.
- ◇ A wipe occurs as a line moves across the screen, with the new scene appearing behind the line.

Previous work dealing with video sequence segmentation is quite extensive. Existing approaches encompass a large range of methods. In this section we provide a brief overview on some commonly used techniques and performance measures for boundary segmentation.

3.7.1 A brief survey of shot boundary segmentation

Classical methods developed for video sequence segmentation used video features from the uncompressed-domain, such as histogram [125], edge tracking [126], pixel-wise difference [127] and shape and colour content analysis [128], etc. The drawback of the high computational load of decompression has stimulated direct processing on conventional compressed data standards, such as H.261 or MPEG [129]. Methods based on compressed-domain features, such as motion vector information [130] and correlation of DCT coefficients [131, 132], have been developed in order to obtain similar results while requiring less computational power. A trend in new approaches is to squeeze more performance using only the macroblock type information.

Early approaches detected shot changes by using a simple global inter-frame difference measure where the dissimilarity of two successive frames is computed at pixel level [125]. The sum of absolute differences for pixels located in the same spatial position in two frames is compared to a fixed threshold in order to locate shot changes. A similar approach where each pair of pixels is considered on a boolean basis is reported in [127].

Shot change detection methods based on gray-scale and colour histograms have been proposed in [125, 133]. In [133] the histogram difference is computed for HSV, YIQ, $L^*a^*b^*$, $L^*u^*v^*$ and Munsell colour spaces whereas in [134] a cosine similarity measure is computed between combined Y, U and V histograms. In [135] video segmentation is performed by using a genetic algorithm to determine dynamic thresholds on colour histogram differences. A twin comparison method where histogram difference values are compared with two thresholds, a high threshold for cuts and a low threshold for gradual transitions is proposed in [127].

In [136] shot changes are detected by weighting the histograms of each colour component according to their perceived importance. A learning procedure to determine the optimal weights for histogram difference computation is introduced in [137]. Several measures computed on normalized histograms have been proposed and evaluated in [138, 139]

In [140] shot change detection is performed on block-sampled images. Successive frames are divided into uniform blocks and then the difference in luminance between pairs of blocks with the same spatial coordinates is compared to a fixed threshold in order to locate shot transitions. In [141] the block-based difference is computed in the HSV colour space in order to avoid camera flashes and in the HSI colour space in [142]. Other block-based approaches use colour histograms [143] and histogram intersection in the $L^*u^*v^*$ colour space [144] in order to improve robustness to change in lighting conditions.

The approach presented in [128] combines moments invariants and histogram intersection. In [126] edge tracking is used to detect shot changes. Edge information is analysed in [22, 145] for detection of gradual transitions and the edge change fraction on several frames serves as a transition detector in [146]. In [147] the area around an edge is termed as an edge-object and matched in successive frames. Shot transitions are detected based on the number of changes in the edge objects between frames. Video segmentation based on wavelet analysis is proposed in [148] for abrupt transitions and extended to gradual transitions in [149]

In [150] gradual transitions such as fades and wipes are detected by applying a model of their temporal effects on frame histograms. A similar model for fade transitions is used in [22]. In another approach gradual transitions are modelled based on variance of pixel intensities during fades [151] and dissolves [152]. Gradual transition models based on the variance of pixel intensities are also proposed in [153] and [154]. In [155] a Bayesian model is assumed for gradual transition while in [156], transition detection errors are modelled as a probability minimisation problem.

A video segmentation approach based on principal component analysis is reported in [157] while the approach presented in [158] uses the principal coordinate system. A segmentation algorithm based on singular value decomposition is introduced in [159]. In [160] hidden Markov models are used to perform video indexing and a similar approach is proposed in [161] based on visual, and audio motion features. In [162] a segmentation method is developed by using a spatio-temporal representation of joint probability images between frames.

Motion based approaches to video segmentation are developed in [163] based on an affine transformation model, and in [164] based on dominant multi-resolution motion estimation. In [165] shot changes are detected using a motion smoothness measure, an approach extended in [166] with a motion-controlled temporal filter that ensures robustness against false detections due to motion. A technique based on motion correlation between successive frames is proposed in [167].

Two video segmentation systems using combinations of features are proposed in [168]. The first system is based on colour histogram comparison while the second system uses motion compensation with optical flow. In [169] transitions are detected based on a combination of three image features: average image brightness, colour distribution and change in pixel values.

The approach reported in [170] uses a majority voting scheme among a combination of five shot boundary detection methods based on: average image intensity [150], Euclidian distance [171], histogram comparison [127], likelihood ratio [140] and motion estimation. In [172] video segmentation is performed using a combination of histogram differences and K-means clustering. The unsupervised K-means clustering approach introduced in [173] is extended with additional features in [174]. A similar approach is taken in [175] while the

method presented in [176] uses a technique based on a Gaussian pyramid representation of the background area of images.

Due to the increasing amount of video material stored in compressed format (in particular MPEG format) it is more efficient to perform the temporal segmentation directly in the compressed domain. The MPEG bit-stream contains features that can be exploited for shot transition detection. The first approach to using compressed domain features is a cut detection algorithm based on the comparison of DCT coefficients [177] on successive intracoded (I) frames. A similar approach is reported in [178]. Successful approaches from the uncompressed domain are adapted to the DC coefficients in the MPEG stream in [132, 179].

A fast algorithm using the DC colour coefficients in MPEG compressed bit-stream is reported in [180] and its extension to motion vectors in [181]. Other approaches relying on DC coefficients extracted from I frames are developed in [182, 183]. An advanced approach is presented in [184] where luminance and chrominance information is extracted for blocks in every I and P frames and PCA is performed on the resulting feature vectors.

Algorithms exploiting the coding mode of the macro-blocks for P and B frames were proposed in [185, 186, 187]. As mentioned in [179], the main drawback of compressed domain techniques is the dependence of the performance on the input bit-stream itself, as sequences encoded with different encoders may lead to significant differences in performance. Generally the performance achieved using compressed domain techniques is lower than that achieved using uncompressed domain approaches, especially for gradual transitions [188].

3.7.2 Evaluation measures

Different measures and evaluation protocols have been proposed in order to compare shot boundary detection algorithms [189, 150, 190]. In recent years, TRECVID [4] is perhaps the most prominent evaluation framework for shot boundary algorithms. The performance of transition detection algorithms is generally expressed as error rate (or sometimes success rate) computed based on correctly detected shot changes, missed shot changes (deleted transitions), and false detections (inserted transitions). The actual definition of error or success rate may be specific to the application domain. Some of the most common employed measures are presented below.

Accuracy is a simple detection measure proposed in [191]:

$$Accuracy = \frac{N_T - (N_D + N_I)}{N_T} = \frac{N_C + N_I}{N_T} \quad (3.7.1)$$

where N_T , N_D , N_I and N_C are respectively the number of actual transition effects present in the video, the number of transition effects deleted, inserted and correctly identified by a boundary detection method. However the accuracy measure can provide erroneous values (negative values) when the number of false transitions inserted is higher than the correctly detected transitions. The size of the video sequence should be taken into account since the number of errors may potentially be equal to the number of frames N_F

Error rate is a measure which inadvertently assigns implicit importance to deleted transition effects over the inserted ones [190]:

$$Error\ rate = \frac{N_D + N_I}{N_T + N_I} = \frac{N_D + N_I}{N_C + N_D + N_I} \quad (3.7.2)$$

The measure lacks an explicit weighting factor to control the importance assigned to deleted transition effects, hence the actual importance of each error type is difficult to assess, which is the reason why it is not adequate for comparison between algorithms.

Precision and recall have been proposed in [192] for evaluation of shot change detection methods:

$$\begin{aligned} Precision &= \frac{N_C}{N_C + N_I} \\ Recall &= \frac{N_C}{N_C + N_D} \end{aligned} \quad (3.7.3)$$

In order to deal with gradual transitions involving several frames, the precision and recall measures defined above are adapted as follows:

$$\begin{aligned} Recall_{cover} &= \frac{b}{a} \\ Precision_{cover} &= \frac{b}{c} \end{aligned} \quad (3.7.4)$$

3. Visual content descriptors for image and video retrieval

where a is the duration of the real transition, c is the detected transition and b is the overlap window between the real and detected effects.

A set of probabilistic measures for evaluating the performance of a temporal segmentation method is proposed in [190]. The measures are presented below.

Error probability computes the probability that a temporal segmentation algorithm will make a deletion or insertion error.

$$p(error) = \frac{N_D + N_I}{N_F} \quad (3.7.5)$$

where N_F is the total number of frames in the video sequence.

Insertion probability is the probability of detecting transitions where such a transition is not present.

$$p(insertion) = p(detection|no\ transition) = \frac{N_I}{N_F - N_T} \quad (3.7.6)$$

Deletion probability is the probability of failing to detect transitions where such a transition is present.

$$p(deletion) = p(no\ detection|transition) = \frac{N_D}{N_T} \quad (3.7.7)$$

Correctness probability is the probability of detecting existing transitions without inserting false ones.

$$\begin{aligned} p(correctness) &= k_1 \cdot p(detection|transition) + k_2 \cdot p(no\ detection|no\ transition) \\ &= k_1 \cdot (1 - p(deletion)) + k_2 \cdot (1 - p(insertion)) \end{aligned} \quad (3.7.8)$$

The importance of deletion and insertion factors can be weighted by setting the weights k_1 and k_2 accordingly.

3.8 Conclusions

This chapter has provided an overview of visual descriptors used in content-based image retrieval. The aim of such descriptors is to provide uniform, robust, discriminant and accurate representations of the visual content. We introduced the main descriptors for colour, shape and texture - features that relate to the work presented in this thesis.

Colour features are one of the most widely used and most reliable visual features. Numerous methods for retrieving images on the basis of colour similarity have been described in the literature. Shape features are needed to represent regions and objects to obtain a more semantic representation of an image. There is considerable evidence that natural objects are primarily recognized by their shape. In addition to colour and shape, texture is an important feature in image retrieval. Although retrieval by texture seems to have limited usability, texture can make a significant contribution to distinguishing between images with similar colours such as sky and sea.

Although not discussed in detail in the chapter, every descriptor has its advantages and as well as shortcomings for specific application domains. The benefit of using a particular representation has to be determined for each scenario based on its type of discriminative power, robustness, invariance, storage capacity and computation complexity for an individual task.

Content representation itself is useful only when a measure of similarity can be computed between feature vectors. In this chapter we have described some of the similarity measures commonly used in image matching and retrieval. In the last section we crossed the bridge from still image to moving images (video) and introduced temporal video segmentation.

The majority of current techniques in content-based image retrieval are based on low-level features. However, low-level features do not have explicit semantic meaning. Moreover the similarity measures between features do not necessarily match human perception and for this reason retrieval approaches based on low-level features are generally unsatisfactory.

Low-level descriptors alone cannot provide enough discrimination power for image and video retrieval. Although research may further improve feature extraction and representation, especially directly from the compressed domain, there is a large gap in relating these basic features to human constructed semantics. However, robust and discriminant features are the basis of content representation, hence the starting point of efficient retrieval.

Chapter 4

Image segmentation

Image segmentation is one of the primary image analysis tasks in object identification and recognition. Segmentation is defined as the process of partitioning an image into disjoint and homogeneous regions belonging to different objects present in the scene. Generally object identification is not considered as part of the actual segmentation process and cannot be achieved without constructing a semantic model of the desired object.

4.1 Introduction

There are several types of images such as media photographic images, range (depth) images, magnetic resonance (MRI) images, thermal images and so on, and related applications that deal with domain specific image processing tasks. In this chapter and this thesis in general we focus our presentation on processing applied to photographic images. Although image segmentation may have different connotations according to the type of images and applications considered, the characteristic purpose of segmentation irrespective of domain is to isolate important “objects” from the background.

Segmentation depends on the presence of salient features in images, more specifically on the presence of dissimilarity of colours and intensity levels within an image, dissimilarity which manifests itself via colours patches, texture patterns, edges, shadows, etc. Prior and during segmentation stages, enhancement techniques are used to improve the saliency of relevant

4. Image segmentation

features within an image. These techniques emphasise particular features of interest in the original image in order to simplify the segmentation task. It is expected that appropriately chosen operators will increase the differences between objects and background improving the segmentation results. Issues related to segmentation involve selecting appropriate segmentation methods, measuring their performance, and understanding their implications on the overall performance of the image analysis application.

As argued in [193], no general algorithm will work for all images. Most studies conclude that future segmentation approaches should be directed at combining spatial and semantic information with low-level visual features. In this chapter the term object is used interchangeably to indicate a semantic (real-world) object and as well as a homogenous region or group of regions presumed to be the segmented representation of a real world object.

Image segmentation was initially proposed for grey level images. Comprehensive surveys on this topic can be found in [193, 194, 195]. With advances in image technology colour information has started to be used since it permits a more complete representation of images and more reliable segmentations. As noted in existing literature reviews [196, 197, 198, 199], many approaches used in segmentation of colour images are extensions to well established grey-level techniques.

The rest of this chapter reviews existing approaches for colour image segmentation. Given the variety of techniques and application scenarios it is clearly impossible to cover all available techniques in this field of research. Consequently the exposition focuses mainly on the methods related to content-based retrieval. The structure of the presentation is modeled based on the segmentation classification introduced in [197] which is widely adopted by many authors.

The methods related to pixel-based segmentation are reviewed in the next section. In Section 4.3 the techniques used for area-based segmentation of images are presented followed by the contour-based methods in Section 4.4. Section 4.5 deals with the neural network approach to segmentation and Section 4.6 with models derived from the properties of light rays as reflected by various materials in the scene. Section 4.7 links image segmentation to video by covering motion-based segmentation methods. Approaches that integrate user interaction are presented in Section 4.8 and aspects related to the evaluation of image segmentation are discussed in Section 4.9. Section 4.10 summarises the content of the chapter.

4.2 Pixel-based segmentation

These techniques make use of the statistics of pixel values mainly considering an image as a set of independent points drawn from a probabilistic distribution. The basic approaches in this category can be classified as:

- ◇ *Histogram-based techniques* where clusters of gray values or colour are identified via peaks of frequency in the histogram data. Since colour images have multidimensional histograms, peaks can be located independently on each colour channel or globally on the 3D histogram.
- ◇ *Clustering techniques* which derive a number of clusters based on the values present in the image that are ultimately used in assigning pixels as belonging to specific objects or regions.
- ◇ *Fuzzy clustering techniques* which are variations of clustering where pixels are not uniquely assigned to regions but rather have membership degrees to multiple regions in the image.

4.2.1 Histogram thresholding

Thresholding is a straightforward technique often used in image segmentation on its own or in combination with other methods. There are many variations of thresholding developed for segmentation using global or local adaptive thresholds as well as single or multiple thresholds.

In [200] histogram thresholding is used for segmenting outdoor images based on colour and hue histograms. Thresholds are set at the salient peaks in the histogram and the process iterates on subsequent image partition in a top-down approach as long as salient peaks are available. A peak is considered salient when it is at least twice as high as peaks in its immediate vicinity.

A recursive segmentation approach is introduced in [201] where a salient homogeneous region is extracted from an image at each iteration. The pixels in the image are separated into two classes: a salient object (region) and background. At each iteration a histogram of

the image is computed and the pixels belonging to the most salient peak are extracted from the image. The approach performs well on images that can be clearly partitioned into two classes: object and background, usually grey-scale images. However, when there are salient regions in the background the segmentation result is unsatisfactory.

In many application domains, images may contain more than a single salient object and the image histogram may not contain prominent peaks such that thresholds can be easily selected. Methods to address this issue have been proposed in [202] and [203] where pixels in the image are modelled by their probability of occurrence in a histogram. Then, an iterative gradient relaxation process is performed on an eight pixel neighborhood followed by thresholding.

Spatial histograms are used in [204] as data projections on which pixel clusters are better defined and hence easier to segment by appropriate thresholding. Spatial information is incorporated in histograms by averaging values on pixel neighborhoods. Experimental results show the proposed technique to perform better than using only histograms without considering the spatial distribution in the image.

A novel adaptive thresholding approach is introduced in [205]. Local thresholds are computed at local salient points (edges) within the image, then a thresholding surface is interpolated from these points. The image is segmented by the thresholding surface. The method is performed iteratively in order to extract multiple objects within an image.

In [206] the thresholds for each colour component are dynamically computed by maximizing the within-group variance and the results are combined with a predicate logic function afterwards. A watershed scheme is adopted to segment 3D histograms in [207] while in [208] only the hue information is exploited on a circular histogram. In [209] histogram segmentation is performed by an entropy-based thresholding method modeled for two distinct classes object and background. The distribution of the chrominance components of objects is modeled in [210] as a Gaussian probability allowing dynamic thresholding. In [211] an adaptive threshold function for RGB and HSI colour spaces is implemented using B-splines.

4.2.2 Clustering techniques

Clustering refers to a class of unsupervised classification techniques designed to determine the intrinsic grouping in a set of unlabeled data. In the context of image segmentation the

problem of clustering can be stated as determining a set of image regions such that every pixel in the image is assigned to only one region such that pixels within a region should show a high degree of similarity. Partitioning of data into clusters is computed based on distance measures, the most commonly used being the Euclidean distance.

K-means clustering is one of the widely adopted techniques for image segmentation [212]. ISODATA (Iterative Self-Organizing Data Analysis Techniques) is another popular algorithm used for colour feature clustering [213]. A evaluation of various clustering methods used in colour image segmentation is available in [214].

Another approach to clustering is the mean-shift algorithm applied to image segmentation in [215, 216, 217]. Mean-shift uses iterative gradient minimisation to locate the position in the feature space where the mean value shows the minimum variation in respect to other neighboring positions. Unlike K-means, it does not require prior knowledge of the number of clusters, and does not constrain the shape of the clusters.

Segmentation techniques derived from probabilistic learning based on finite mixture statistical modeling of data are presented in [40, 218, 219]. In the probabilistic approach, pixels are assumed to be produced from a set of unknown sources and segmentation consists of inferring the parameters of these sources and identifying which source produced each pixel. Since this approach relies on well defined theoretical models, estimation of parameters can be addressed in a more formal way as opposed to heuristic methods such as K-means. The standard approach to computing mixture models is the Expectation-Maximization (EM) algorithm [220, 221] which iteratively converges to a Maximum Likelihood (ML) estimate of the mixture parameters.

An original technique is proposed in [222] based on the idea on constrained gravitational clustering. Points in the colour space are modeled as particles that interact according to the gravitational laws. In [223] pixels are represented in a tree structure and clustering is achieved by recursive merging of branches, whereas an connected components approach is adopted in [224]. In [225] spatial constraints are incorporated in the K-means clustering for grey-level images and in [226] the method is extended for colour images.

4.2.3 Fuzzy clustering

In real images, the border between two regions is not always apparent in many cases, and there exists more than one valid partitioning. Uncertainty regarding which region a pixel should belong to can arise within each level of image segmentation. Since at any level decisions are based on the results of previous levels, early decisions impact strongly on the outcome of segmentation. In the fuzzy clustering approach sufficient information is preserved at lower levels such that ambiguous decisions can be deferred to higher levels where more information is available. Fuzzy set theory provides a mechanism to represent and manipulate uncertainty and ambiguity [227, 228].

In one of the first approaches to fuzzy segmentation [229], the membership value of a pixel to a given region is calculated based on the distance to the region's centre. A generalised "farness" measure for fuzzy segmentation which takes account of the distance in the spatial domain and the contrast in colour space is derived in [230]. The approach introduced in [231] uses a two stage coarse to fine segmentation, where some pixels are definitely assigned to particular regions while others close to actual region's borders are subject to fuzzy membership. In [232], the maximum fuzzy entropy principle is applied to map the colour image from the space domain to the fuzzy domain by preserving colour homogeneity.

4.3 Area-based segmentation

These techniques rely on uniformity constraints imposed on image regions. While in the pixel-based approach each independent data point satisfies a threshold or distance criteria, in area-based approaches homogeneity criteria are applied to entire segments. The basic methods in this category can be classified as:

- ◇ *Region-growing techniques* make use a initial set of seeds around which pixels are iteratively added based on a general uniformity criterion. Finally, each pixel in the image will be joined to one of the seeds. These methods are sensitive to the choice of seeds.
- ◇ *Split-merge techniques* are top-down approaches which start from non-uniform regions and recursively divide them until a uniformity criterion is satisfied. The obtained

regions can be merged in bottom-up approach in order to obtain larger homogenous segments.

- ◇ *Graph-theory techniques* which partition a graph describing the whole image into a set of connected components that correspond to image regions.

4.3.1 Region-growing techniques

Region growing is similar to a sequential clustering process. Hence, the results may depend on the order in which the data points are processed. The advantage of these techniques is that the regions constructed are rather compact, large and spatially connected. However, similar to other clustering techniques, there are issues related to choosing suitable seed points and adequate homogeneity criteria. A common postprocessing step in the region growing approach is the merging of small regions in order to generate larger regions. Apart from variations in the selection of seeds, the typical method applied to growing is the watershed scheme [233].

The work in [234] suggests several homogeneity criteria and a merging stage based on similarity of colour distribution. In [235] colour and luminance are used together for identifications of seeds (markers) by morphological open/closed operations. Both approaches implement the region-growing as a watershed algorithm.

In [236] the initial seeds are positioned at points of local minima in the colour image gradient. An a posteriori procedure prunes the initial seeds in order to obtain just one marker for each region. The region growing is performed with a modified watershed algorithm directly on the original colour image instead of a gradient image. The approach in [237] determines the location of markers through colour quantisation and limits the number of regions according to the number of colour classes present in the image. The homogeneity criterion is evaluated at multiple scales within a variable local window.

A special hexagon topology is introduced in [238]. The regular hexagonal grid makes region growing independent from the starting point and the order of processing. The work in [230] implements region-growing in a fuzzy segmentation framework. Several growing algorithms are evaluated and a number of variations on the watershed transform are proposed in [239].

4.3.2 Split-merge techniques

The split-merge approach starts from an initial inhomogeneous partition, usually the entire image, and recursively splits these partitions until homogeneity is obtained. The path followed by splitting is generally recorded in a tree representation. The splitting phase usually gives a large number of small uniform regions. A merging step is often necessary to join together small regions into large segments.

In [240] colour texture homogeneity is modeled by a Gaussian Markov Random Field (GMRF), while in [211] the Markov Field is defined over the tree representation of the image. In both approaches a relaxation process controls splitting and merging on a grid of regular image blocks.

Numerous variations of split-merge strategies have been reported in the literature. In [241] the K-means algorithm is used for both splitting and then for merging. The approach presented in [242] performs splitting on the watershed transform of the gradient image and merging based on Self-Organising Maps (SOM). Watershed is also used in [207] in the splitting phase, and the resulting regions are later merged according to their colour contrast. A similar approach is adopted in [243] where the merging is performed on a region adjacency graph.

Although the common approach to trace the path followed during recursive partitioning is the tree representation, alternative approaches have been also proposed. In [244] the tree representation of the partition path is abandoned in favour of incremental Delaunay triangulation while Voronoi diagrams are used to capture the partition connectivity in [245].

4.3.3 Graph-theory techniques

In the graph-based approach, images are described as a graph of connected components. In this view, the problem of segmentation is to find the partition of the graph that satisfies a set of constraints usually related to colour or texture homogeneity and similarity, or length of description for a partition.

One of the first approaches [246] has as partition criteria the length of the description for the Minimum Spanning Tree (MST) in the graph. In [247] this is based on computation

of the minimum cut in the graph representing the image. The cut criteria is designed to minimise the similarity within the regions being split. Similar approaches are described in [248, 249] where a normalised version of the minimum cut is used. The last two methods mentioned above are based on local image features and are computationally more efficient.

The approach in [250] creates partitions in the image graph based on a measure of local variability defined upon the vicinity around a data point. Since local features alone do not provide reliable segmentation, in [251] the variability measure is extended to also incorporate global feature. In [252] graph partitioning is based on a simulated annealing model and hierarchical approximation in order to minimise the space of all possible partitions.

In [253] the minimum spanning tree algorithm is altered by further splitting the regions with large variance in homogeneity and merging those of low variance in an iterative process. A dynamic partitioning approach is described in [254] where a heuristic cost function is derived based on colour features in order to reduce the complexity of the spanning tree. Although most authors prefer a top-down region splitting scheme, in [251] is proven that a bottom-up scheme such as region growing performs adequately on graph-based segmentation.

4.4 Contour-based segmentation

The edges of various regions present in an image are important cues for segmentation. The abrupt changes in a pixel colour or intensity associated with edges of regions can indicate the real borders of physical objects existent in the image. However, not all image discontinuities are necessarily semantic contours as some of them can be due to capture or compression artifacts or part of a textured region.

Contour techniques make use of local or global image information. The local information is limited to vicinities of pixels which provides for fast computation. Such approaches are based on gradient operators on gray scale images. Extensions to colour image have been proposed in [255, 256, 257]. Methods that use global image information are generally optimisation processes, mainly iterative, and often slow to converge. However they seem to perform well especially on noisy images.

Boundary analysis is used in [258] to perform segmentation of natural scenes. The edges extracted by a differential filter are joined in line segments in order to produce closed

contours. The length, contrast, frequency, mean, variance and location of each line segment are computed and several image partition alternatives are examined. This approach does not provide a unique partitioning of an image into a set of regions but provides a score for every possible region that could be constructed from the given contours, although regions may overlap.

A problem which occurs in edge based segmentation is that edges are usually disconnected line segments rather than continuous contours. This leaves gaps that may cause the merging of dissimilar regions. An approach to closing these gaps is proposed in [259] based on an expansion-contraction technique. The detected edges are modeled as active contours which can expand and contract in order to enclose homogenous regions.

An approach derived from the predictive coding model is proposed in [260]. The direction of the edge is detected by changes in the flow of colour and texture in the hue space. The work described in [261] proposed a framework for object segmentation based on snakes and active contours. In the snake-based approach, an initial contour is deformed towards the boundary of the detected object. Deformation is obtained by minimising a global energy function such that a position of minimum energy is obtained at the object boundary. The formulation of active contours for colour images was introduced in [262, 263]. Colour invariant snakes were proposed in [264] in order to achieve robustness to disturbances due to shadowing and lighting variance. As pointed out in [265], the active contour approach shows a close relationship to other segmentation frameworks such as anisotropic diffusion and partial differential methods.

4.5 Neural network based segmentation

Neural networks are interconnected groups of large numbers of elementary processors each performing simple functions. Their high degree of redundancy and parallelism allows for fast computation times and robustness to disturbances which makes them suited for pattern recognition tasks. In the case of image segmentation, neural networks facilitate taking spatial information [196] into account. However, the number of segments within an image must be known beforehand and the network has to be trained to recognise patterns.

A number of neural algorithms are reviewed in [195] for grey-level image segmentation. In

[266] two neural-based approaches are introduced on a Hopfield network. One of the algorithms consists of three dedicated networks for colour features with their results combined afterwards. The other algorithm has a single network which classifies pixels into classes according to a learned histogram distribution.

Neural techniques for segmentation are considered optimal solutions for specific classification problems where the number of possible classes is known beforehand [196]. Medical applications and human face localisation in colour images are two such successful fields of application. In [267] a voting system with multiple networks locates faces of people in photographs. False detections are added into the training set in order to improve the network performance.

A neural network scheme for face detection and eyes localisation in colour images is presented in [268]. The scheme is based around a Self-growing Probabilistic Decision-based Neural Network (SPDNN) which learns the conditional distribution for each colour class. In [269] a three layer neural network is developed to segment stained medical images from three different classes while a backpropagation network is used in [270]. An unsupervised approach using Hopfield networks is applied in [271] to the segmentation of colour images of stained liver tissues.

A neural network-based tool for colour image segmentation has been proposed in [272]. In this approach a large feed-forward network is used to categorize pixels into specific groups (segments) using the surrounding contexts in which the pixels are located. However, due to the size of the neural network, the tool requires extensive training and should be run on a cluster of computers in order to provide real time results. In their experiments the authors give the training time as one month on a 2.2 GHz PC.

4.6 Physics-based segmentation

Segmentation methods founded on physical models of light interaction with coloured surfaces have been proposed in order to provide robustness to lighting and shadowing effects [196]. These phenomena can produce changes in the appearance of uniformly coloured surfaces thus introducing ambiguity in the segmentation process. In order to overcome these drawbacks, the segmentation algorithms should incorporate models for the reflection

4. Image segmentation

of coloured materials. In reviews [196, 199], materials are usually classified into three main categories: optically inhomogeneous dielectrics, optically homogeneous dielectrics, and metals.

A major contribution to the field of physics-based segmentation is the work presented in [273] which derives a model of dichromatic reflection for inhomogeneous dielectrics. The proposed model can be used to distinguish colour changes at material boundaries from changes due to shading. It can also be used to determine the colour of the inter-reflected light and hence to remove it so as to facilitate an accurate shape-from-shading and colour-based object recognition

An extension of the above model is presented in [274] where colour reflection based on the dichromatic model is defined on a particular colour space, called S-space. Light reflected by various bodies produces clusters with specific shapes in the S-colour space. Specular and diffuse interface reflections are separated in this space by analysing variations in brightness, hue, and saturation. The authors show that their approach allows segmentation of uniformly coloured dielectric surfaces.

A reflection model for metals is explored in [275, 276] within extensive experiments. With this model, the reflectance function of metals can be separated into a geometrical and a spectral component. The geometrical effects in the scene can be factored out through normalisation. The papers also describe a normalisation method for colour segmentation of inhomogeneous dielectrics and metals.

4.7 Motion-based segmentation

The segmentation methods reviewed in the previous sections of this chapter are mainly related to still images. For video applications, it may be more practical to segment moving objects from a dynamic scene with the aid of motion information. Segmentation of moving objects plays an important role in image sequences, once objects are extracted they can serve a variety of purposes such as: image compression, object recognition, enhanced content interaction, etc

Motion-based segmentation techniques can be categorised according to the dynamics of the scene. The simplest case is when a static camera records a moving object, thus when

the only motion activity is produced by the object of interest. Object segmentation in scenes involving camera motion or zooming effects is a challenging task. In such scenes segmentation is possible only when the dominant motion of the camera can be determined efficiently. Usually this implies that a large part of the image is background. Assuming the dominant motion in the scene to be produced by the camera and can be determined, compensation for this motion can be performed.

The general approach to motion segmentation is to partition an image into regions of different motion characteristics. However, this may result in a large number of regions since different parts of a non-rigid moving object might undergo different motions. Consequently, recovery of objects from motion regions is a challenging task.

Various motion segmentation algorithms have been proposed in the research literature. Good surveys on this topic can be found in [277, 278]. In [279] object segmentation is performed by thresholding the change detection mask. Regions of uncovered background are removed from the object mask by using a displacement vector field. In [280] an edge map is calculated from the inter-frame difference. The edge map containing edge pixels from both frames is compared to the edge map of the reference frame. The final segmentation is achieved through morphological and area filling operations. Optical flow and morphological operators are combined with connected component analysis on inter-frame difference images in [281].

A geometric model based on partial difference equations for segmenting and tracking moving objects is proposed in [282]. This model is an extension of the active contours framework by adding motion-based terms. The model does not account for camera motion. A fast version of active contour for segmentation and tracking is developed in [283] by integrating temporal and spatial edges. Again the model assumes a static camera and a moving object. In [284] segmentation and tracking is performed within an active contours framework by using inter-frame differences.

An approach able to deal with segmentation in the presence of camera motion is proposed in [285]. Again moving objects are detected by means of inter-frame differences modeled within a statistical framework. Camera motion is approximated with a three-parameter model (two translation and one zoom parameter). An extension of this work that integrates an optical flow algorithm is presented in [286]. In [287] a number of motion classes are determined

from the correspondence of feature points. These motion classes serve to initialize the segmentation process in a level-set based segmentation algorithm.

4.8 Interactive tools for image segmentation

Image segmentation is a well studied domain but still remains an ill-posed problem and the notion of correct segmentation is essentially dependent on the application. Segmentation is a tedious and difficult procedure if performed manually. Although automated methods for image segmentation have been developed and successfully applied to certain well constrained problems even the most advanced methods require some form of user input in order to adapt to a large range of segmentation scenarios. Adding an appropriate level of user interaction into the automatic analysis can significantly improve the accuracy of image segmentation.

User interaction supporting image processing tasks falls into one of the following categories [288]:

- ◇ *Algorithm initialization.* Most algorithms require some kind of initialization by having the user select starting parameters for the algorithm or initial markers in the image to be segmented.
- ◇ *Intervention or feedback response.* This type of interaction consists of either steering the process continuously or intermittently towards a desired result, or stopping the process midway to introduce corrections when there are erroneous results.
- ◇ *Evaluation of results.* When the final results are unsatisfactory, the entire process can be repeated with different parameters or in some cases the result are simply rejected.

Computer mice and graphic tablets are the most wide-spread and efficient input devices for graphical applications. Consequently, most image editing applications use mouse-input for user interaction tasks. The commonly used graphical tools associated with arbitrarily-shape image segmentation are:

- ◇ *The pencil.* The user brings the mouse over the area of the image that needs to be selected (painted). Although this is primarily an editing tool, when the set of modified pixels (painted) are considered as a mask it can be seen as a selection tool.

- ◇ *The brush* selects pixels within a user defined radius around the mouse cursor. The radius around the cursor location can shrink or expand according to the duration of the mouse click or the pressure exercised on the input pen in the case of a graphic tablet.
- ◇ *The lasso or scissor* lets the user draw a closed contour, and selects all pixels within the contour. In the Intelligent Scissor mode the user selected area is adapted to automatically detected contours [289].
- ◇ *The magic wand* is a region growing algorithm which starts from an initial seed point and selects all connected pixels within a given colour distance.
- ◇ *The blow tool* is a combination between brush and magic wand. It expands a region around the mouse cursor by following the motion direction of the mouse. The expansion factor is derived from the distance between different pixels touched by the mouse.

Different multi-scale segmentation approaches have been proposed in the research literature, but as far we are aware are still to be implemented in commercial applications. In the multi-scale approach the selection tools operate at region level rather than pixel level. Underlying algorithms for this type of interaction are described in [290, 291, 292, 293].

There are many successful interactive segmentation tools developed within the research community. A tool that allows region-based interaction on nested partitions is presented in [294]. Pre-segmented regions are represented in a hierarchical structure where different levels in the hierarchy can be associated to the number of regions contained in the partition. Semantic objects are created by merging regions at different levels within the hierarchy.

In [295] the classical interactions pencil, brush, lasso and magic wand are extended to operate on pre-segmented regions. The user can choose the level of pre-segmentation desired and the interaction that suits the task at hand. In [296] an image is partitioned into pre-segmented regions which are then represented into a binary partition tree. The user can build semantic objects by selectively clicking on image regions.

A scribble-based approach to semantic object segmentation is described in [297]. In order to mark objects the user draws two coloured scribbles over the foreground and background

parts of the image respectively. Since the image is pre-segmented into uniform colour regions, the user's scribbles specify a number of regions that are then classified as background or foreground. The regions that are not initially intersected by one of the scribbles are merged to regions of similar colour already classified.

A different interaction approach is taken in [298] where pre-segmented regions need to be removed one by one until only the foreground object is left in the image. At first glance this seems to require quite a considerable amount of interaction and to be starting from the wrong point, deleting rather than adding regions.

4.9 Evaluation of image segmentation

The amount of precision needed for image segmentation depends on the domain of application. Segmentation masks can be coarse for surveillance applications whereas multimedia applications require high accuracy for the object's contour. However, the goal of image segmentation is to accurately extract the contours of real objects contained in a scene. A standard evaluation measure, if available, would provide a ranking among different segmentation algorithms or a way to optimally set the parameters of a given algorithm [299].

Many authors consider human assessment as the best form of evaluation for any segmentation algorithm [195]. Other authors have proposed objective evaluation procedures and metrics, including application-oriented methods [300, 301]. However, there is no universally accepted method of objective evaluation of segmentation results which makes the automated selection of an optimal segmentation algorithm a real challenge.

The algorithms related to objective evaluation of still image segmentation can be classified into two groups [302]:

- ◇ *Analytical methods* which evaluate segmentation algorithms by considering their underlying principles, requirements and complexity.
- ◇ *Empirical methods* which evaluate segmentation according to the obtained results. Some of these methods estimate the quality of segmentation based on intuitive measures such as uniformity or contrast between regions. Others, usually termed as discrepancy metrics, compare the segmentation mask to reference masks (i.e. manually annotated ground truth).

The analytical methods to evaluation of segmentation algorithms rely on an extensive theoretical framework and thus do not suffer from influences caused by the arrangement of evaluation experiments. However they have not received much attention in the literature perhaps because of the inherently limited comparison possible by such analytical means [299].

The empirical methods estimate the quality of segmentation according to human intuition. These methods rate different algorithms by computing a “goodness” measure based on the segmented image. Different types of measures have been proposed: colour uniformity [302], entropy [195], intra-region uniformity [303], inter-region contrast [304], shape complexity of the segmented regions [305], etc.

For foreground/background segmentation the quantitative measures typically involve the number of misclassified pixels and their positions. The most commonly used measures are: the percentage of area misclassified and the pixel distance error [302]. A evaluation criterion modelled on human perception of segmentation accuracy is proposed in [301]. In this approach the mask of segmented objects are compared to manually extracted ground truth and discrepancies are computed based on the spatial accuracy of a region’s contours. A generic evaluation framework designed to meet a large class of segmentation applications is introduced in [299].

4.10 Conclusions

This chapter provides an review of algorithms used in image segmentation with a main focus on the methods related to segmentation for content-based retrieval. In the context of image retrieval, segmentation provides a way to focus the search on the most relevant areas of the image, the semantic objects present in the scene. We introduce here the main techniques developed in this research area covering methods such as histogram thresholding, clustering, region-growing, split-merge algorithms, contour and motion based segmentation.

There is no universal theory of image segmentation, all existing approaches rely on ad hoc assumptions to some extent. Most techniques are tailored to particular applications and operate under certain constraints. General purpose algorithms are neither robust or efficient. Most segmentation approaches are based on the similarity between neighbouring

pixels or on the colour homogeneity of regions. These assumptions are often problematic in the presence of inhomogeneities induced by shadows or texture.

The problem of image segmentation is basically one of psychophysical perception. Many authors consider human assessment as the best form of evaluation for any segmentation algorithm. Extensive research has been dedicated to subjective and objective qualitative evaluation of segmentation, but estimation of the algorithmic complexity of segmentation methods has not received much attention in the research literature.

Robust segmentation is achieved only for particular applications in well specified scenarios. Image segmentation depends on so many factors, such as homogeneity, spatial compactness, continuity, or correspondence with the psycho-visual perception. Although automated methods for image segmentation have been developed and successfully applied to certain problems, even the most advanced methods require some form of user input in order to adapt to a large range of segmentation scenarios.

Since automatic segmentation is not achievable on generic content, employing user interaction significantly improves the accuracy of segmentation for any type of content. However, manual segmentation is a tedious and difficult procedure therefore the interaction should be minimal. Scribble-based interaction described above has a number of advantages over other interactive segmentation tools. The interaction performed by the user is a simple line scribble obtained with a single click and a short mouse motion. This is much easier and much faster (less than 3 seconds) than other interactive segmentation methods which either require the user to select all individual pixels in the desired object one by one, or to accurately follow an object's contour. The scribbles are drawn over an image that is already pre-segmented in small homogenous regions obtained from an automated shortest spanning recursive tree (RSST) approach. This means that once the user has scribbled (labelled) some foreground and background regions, those which are left unlabelled can be directly assigned to the background/foreground according to a simple colour distance without any need for further interaction.

Chapter 5

Relevance feedback

Relevance feedback (RF) is an efficient means of narrowing down the gap between low-level visual feature representation of an image and its semantic meaning in a content-based image retrieval (CBVIR) scenario. Without detailed knowledge of the video archive structure, and of the retrieval environment, most users find it difficult to formulate well-designed queries. Since the query formulation process is not transparent to retrieval system users, the initial query is likely to be far from an optimal formulation. Consequently, the initial retrieval operation can be considered as being a trial run only designed to retrieve a few useful items from a given collection. The items retrieved in the initial run can then be examined for relevance and the query formulation adapted accordingly in the hope of retrieving additional useful items during subsequent search operations.

5.1 Introduction

The relevance feedback process was introduced in the mid '60s [306] as an automatic method for query reformulation in text retrieval. The main idea of relevance feedback consists of choosing important features of certain previously retrieved items that have been identified as relevant by the users and emphasising these features in a new query formulation. Additionally, the irrelevant features can be de-emphasized in the future query formulations. This has the effect of altering the query closer to relevant items and further away from

5. Relevance feedback

non-relevant items. The expectation is that more relevant items are retrieved in subsequent search iterations.

The relevance feedback mechanism provides additional advantages for a retrieval system. The most significant of these are:

- ◇ It acts as a conceptual screen between the user and the query formulation mechanism, allowing the user to formulate powerful queries without intimate knowledge of the search process or of the archive structure.
- ◇ It structures the search process by breaking the search operation into sequences of iterative steps designed to gradually approach the targeted relevant documents.
- ◇ It provides a controlled environment for query formulation and subsequent adaptation by allowing the user to emphasise relevant items and their features as required by the particular information needs of the user.

However any relevance feedback mechanism relies on a set of general assumptions that need to be fulfilled in order to exploit the above-mentioned advantages effectively:

- ◇ The first and the most important assumption is that discrimination between relevant and non-relevant items is possible with the available features. Without this condition satisfied relevance feedback is futile.
- ◇ There can be established a relatively straightforward transformation between the topology of the feature space and the semantic characteristics of the items the user wants to retrieve.
- ◇ There are relevant items in the archive and they are a small part of the entire available collection. When such items form the majority of the collection, the retrieval process may perform effectively without necessitating a relevance feedback mechanism.
- ◇ Users may provide only limited and sometimes inadequate feedback information usually predominantly labeling positive items and less often negative items. This assumption plays an important role in the selection of the feedback strategy and in the design of user interfaces.

The remainder of the chapter is structured as follows. The principle of relevance feedback is presented in section 5.2 covering the query update models. Section 5.3 presents aspects specific to the use of relevance feedback in multimedia retrieval such as those related to utilisation scenarios, data models and selection strategies. Performance evaluation of relevance feedback is discussed in Section 5.4. Section 5.5 summarises the content of the chapter.

5.2 Principle of relevance feedback

Relevance feedback was originally designed for text retrieval where the query model consists of a weighted selection of search terms [33, 307, 308]. A query vector can be written as:

$$Q_0 = (q_1, q_2, \dots, q_i) \quad (5.2.1)$$

where q_i represents the weight of term i in the query. The weights are in the range 0 and 1; with 0 representing a term absent from the query vector and 1 representing a fully weighted term. A term could be a word chosen from a term dictionary or even a full phrase in the natural language of the user.

Through relevance feedback, an updated query vector Q'_0 is derived starting from the initial query vector:

$$Q'_0 = (q'_1, q'_2, \dots, q'_i) \quad (5.2.2)$$

where q'_i represents the altered term weight assignments for the i index terms. New terms are introduced in the query by assigning them a positive weight whilst older terms are removed by reducing their weight to 0.

In this approach, the feedback process can be visualized as a shift in the query vector from one area to another into the T -dimensional space defined by the T index terms.

The relevance feedback process is illustrated in Figure 5.1 for a two-dimensional case where the indexing terms are *information* and *retrieval*. If document D_1 is specified as relevant to the initial query Q_0 the feedback operation updates the query to Q' which is closer to D_1 . For D_2 specified as relevant, the updated query is Q'' . The updated queries are expected to retrieve more relevant documents similar to the previously identified D_1 or D_2 respectively.

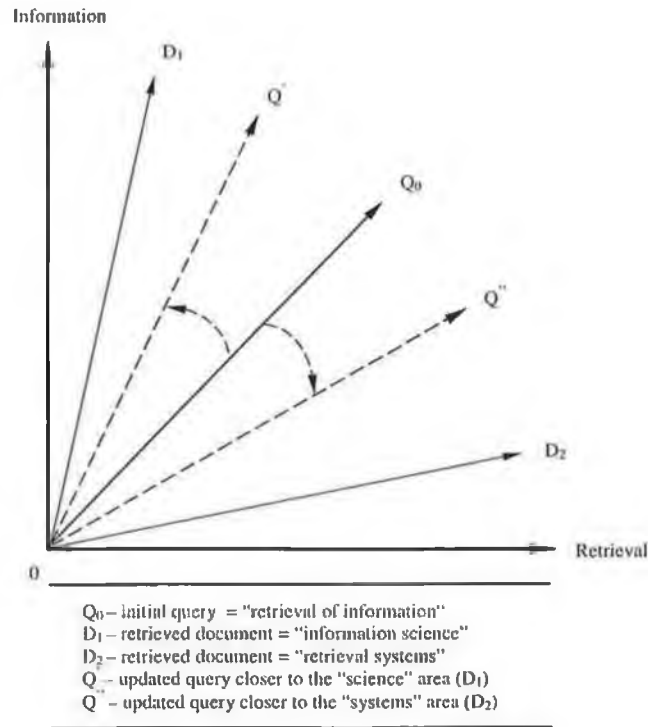


Figure 5.1: Relevance feedback illustration, as depicted in [308]

5.2.1 Vector model

Both the information items D stored in the collection and the requests for information Q can be represented as T -dimensional vectors of the form:

$$\begin{aligned}
 D_0 &= (d_1, d_2, \dots, d_i) \\
 Q_0 &= (q_1, q_2, \dots, q_i)
 \end{aligned}
 \tag{5.2.3}$$

where d_i and q_i represent the weight of term i in D and Q , respectively. The query-document similarity measure can then be computed as the inner product between corresponding vectors:

$$\text{Similarity}(D, Q) = \sum_{i=1}^T d_i' \cdot q_i
 \tag{5.2.4}$$

5. Relevance feedback

The optimal query $Q_{optimal}$ that provides best retrieval results for the above given similarity is of the form [33]:

$$Q_{optimal} = \frac{1}{n} \left(\sum_{\text{relevant}} \frac{D_i}{|D_i|} \right) - \frac{1}{N-n} \left(\sum_{\text{non-relevant}} \frac{D_i}{|D_i|} \right) \quad (5.2.5)$$

where D_i represents the document vectors, $|D_i|$ is the corresponding Euclidian vector length, N is the size of the collection and n the number of relevant documents in the collection.

However, the above optimal query cannot be used in practice as an initial query formulation because the set of n relevant documents is not known in advance. The optimal query is employed in generating a feedback query once relevance assessments are available for some of the items previously retrieved in the initial search iteration. In this case the updated query following the retrieval of n_1 relevant and n_2 non-relevant items can be formulated as:

$$Q_1 = Q_0 + \frac{1}{n_1} \left(\sum_{\substack{\text{known} \\ \text{relevant}}} \frac{D_i}{|D_i|} \right) - \frac{1}{n_2} \left(\sum_{\substack{\text{known} \\ \text{non-relevant}}} \frac{D_i}{|D_i|} \right) \quad (5.2.6)$$

where Q_0 and Q_1 represent the initial and first iteration queries respectively.

In the general formulation the expression (5.2.6) can be written as:

$$Q_{i+1} = \alpha Q_i + \beta \left(\sum_{\text{relevant}} \frac{D_i}{|D_i|} \right) - \gamma \left(\sum_{\text{non-relevant}} \frac{D_i}{|D_i|} \right) \quad (5.2.7)$$

where the normalized weights α , β and γ are between 0 and 1.

The vector alteration approach presented is conceptually simple; the modified term weights being directly obtained from the weights of the corresponding terms in relevant and non-relevant documents. When the weights accurately reflect the real values of the terms, standard vector modification provides a powerful query construction method.

5.2.2 Probabilistic model

Another approach to relevance feedback is the probabilistic retrieval model [309, 310, 311] which uses an optimal retrieval rule to rank the documents in decreasing order according to the probabilistic expression:

$$\log \frac{P(X|relevant)}{P(X|non-relevant)} \quad (5.2.8)$$

where $P(x|relevant)$ and $P(x|non-relevant)$ represent the probabilities that a relevant or non-relevant item, respectively, is modeled by the query vector X .

Given the above expression a query document similarity value between the query and each document $D_0 = (d_1, d_2, \dots, d_i)$ can be written as a function of two parameters p_i and u_i that represent the probability that the i -th term holds the value 1 in a relevant and non-relevant document, respectively:

$$\begin{aligned} Similarity(D, Q) &= \sum_{i=1}^T d_i \log \frac{p_i(1-u_i)}{u_i(1-p_i)} + constant \\ p_i &= P(X_i = 1|relevant) \\ u_i &= P(X_i = 1|non-relevant) \end{aligned} \quad (5.2.9)$$

However, in practice the values of p_i and u_i are not known for all document terms. The most common assumption is that the p_i values are constant for all terms (typically $p_i = 0.5$) and u_i values are set to the proportion of documents in the collection that carry the term i ($u_i = n_i/N$, where n_i is the number of documents containing the i term and N is the total number of documents in the collection). For the initial search run, the expression (5.2.9) is then reduced to:

$$Similarity(D, Q) = \sum_{i=1}^T d_i \log \frac{N-n_i}{n_i} \quad (5.2.10)$$

During the iterative searches the accumulated statistics for relevance and non-relevance of retrieved items are used to evaluate the expression in (5.2.9). For this purpose the relevant

and non-relevant term distributions within the previously retrieved items are assumed as being the same as the distribution of terms in the entire document collection. Then the probabilities for relevant p_i and non-relevant u_i items in the collection can be written as:

$$p_i = \frac{r_i}{R} \text{ and } u_i = \frac{n_i - r_i}{N - R} \quad (5.2.11)$$

where r_i is the number of relevant documents holding the term i and R is the total number of relevant documents retrieved. Substituting these values in expression (5.2.9), the formula becomes:

$$\text{Similarity}(D, Q) = \sum_{i=1}^T d_i \log \left(\frac{r_i}{R - r_i} \frac{N - R - n_i + r_i}{n_i + r_i} \right) \quad (5.2.12)$$

For the particular values, $R = 1$ and $r_i = 0$ the logarithmic expression in the above formula is reduced to 0. For this reason an adjustment factor (typically 0.5) is added when defining the p_i and u_i probabilities, which changes the formulas to:

$$p_i = \frac{r_i + 0.5}{R + 1} \text{ and } u_i = \frac{n_i - r_i + 0.5}{N - R + 1} \quad (5.2.13)$$

The probabilistic feedback models are optimal under the assumed conditions of term independence and binary (relevant versus non-relevant) document indexing. In the probabilistic approach the feedback process is directly related to the derivation of a weight for query terms

5.3 Relevance feedback in image and video retrieval

The goal of content based image and video retrieval systems is to provide the functionality for retrieving images and video sequences which are visually and conceptually similar to the query image or video clip. In such systems, the indexing terms are usually low-level visual features: colour histograms, texture features, edge-content features, etc. The feature vector is typically represented as a point in an N -dimensional space in which the number of indexing features provides the dimensionality. The implicit assumption is that a vicinity of points in the feature space represents visually similar items (images or video). However some

5. Relevance feedback

low-level features may be meaningless for the system's users or may be highly correlated with other features. Some features could be significant for certain queries but may lose significance for other queries [312]. Significance and similarity are subjective notions and may alter depending on the query, the user or the particular moment in the search task. When there is a major discrepancy between the similarity as perceived by the user and similarity as computed by the retrieval system the search results are inadequate [313].

User feedback in the retrieval results can be exploited in a relevance feedback process on subsequent retrievals with the goal of increasing retrieval performance. A typical feedback supported search session develops as follows: the user presents a query (an image or a feature vector depending on the system's input interface) to the system whereupon the system retrieves a fixed number of images using a default similarity metric. The user rates some of the returned results with respect to the relevance of the result of the retrieval task at hand. The ratings may vary from relevant or non-relevant to finer gradation of relevancy such as somewhat relevant, not sure, and somewhat irrelevant, depending on the particularities of a given retrieval system. The relevance feedback mechanism makes use of feedback information in order to select another set of images for retrieval. The user can rate the new images in a similar way and the process can iterate again in a closed-loop until the user is satisfied with the retrieved results or until no more progress can be achieved. The system's goal is to effectively infer which images are of interest to the users based on their feedback.

In a relevance feedback mechanism, the emphasis is on the online feedback from users. In order to support efficient feedback, the retrieval system has to provide for certain functionalities as follows:

- ◇ The system should require only a reasonable amount of feedback. The user should be required to rate at each iteration only a small number of images from the set of images retrieved. The user may become tired if asked to provide labor-intensive rating.
- ◇ The system should start retrieving acceptable results after only a few iterations. When a large number of iterations is required the users will become increasingly annoyed further limiting their feedback input.
- ◇ The retrieval time per iteration should not exceed a reasonable period (real-time), suitably correlated with the accuracy of retrieval in order to prevent user frustration.

5. Relevance feedback

Exact bounds on what constitutes reasonable amounts of time or accuracy are domain and user specific.

5.3.1 Utilisation scenarios

From the user point of view, content based image retrieval can be described as modelling the following utilisation scenarios [314]:

- ◇ *Explore and search for some relevant items.* This is the case where the user does not have a strong prior notion of relevance and relies on the exploration of the image collection to clarify it. The retrieval system should allow an extensive exploration without necessarily retrieving all relevant images and even more importantly without filtering out all non-relevant images since the relevance is not well-defined.
- ◇ *Retrieve most items from a relevant set.* This is the case where the system has to filter out the non-relevant items. However, a number of non-relevant items are allowed to surface as long as they do not impede the search. In fact the presence of a few non-relevant items can be beneficial since it may help the user in indicating negative examples.

Relevance is a subjective notion associated with visual features or with semantic characteristics that are shared by a group of items. Evaluating relevance is considered to correspond to a ranking problem [16] where items should be ordered by decreasing relevance. The precise ranking of relevant and non-relevant items is generally not required since it is often difficult for the user to choose between two alternative rankings. Ranking the most relevant items before the non-relevant ones is generally considered appropriate for retrieval since this brings the relevant items to the user's attention. Thus, the evaluation of a relevance feedback mechanism is concerned with measuring the quality of ranking relevant items before non-relevant ones and the speed of improvement during feedback iterations.

5.3.2 Feedback information

The source and nature of the information that can be exploited for relevance feedback may be specific to each application. Generally the feedback information can be classified according to the following input sources [315]:

- ◇ Prior feedback information acquired before the actual search session based on domain-specific similarity, prior clustering or related to the nature and context of the current searching session.
- ◇ Correlation of retrieval behaviour, feedback and user profiling within a group of searchers can provide valuable information. Although this information is usually collated offline (e.g. collaborative filtering [316]) correlation of feedback between users may be exploited online in multi-user interactive systems [317].
- ◇ The feedback provided by a user at different stages in the search session or before that. This feedback can include the response in the current iteration, the responses in the previous iterations and possibly a model of subjective perceived similarity.

A relevance feedback mechanism has two components: a learner and a selector. As the user labels retrieved images as relevant or non-relevant, the learner exploits this information to re-estimate the target of the user. The current estimation of the target serves to select the images to be retrieved in the next iteration.

There are various approaches to collecting and exploiting feedback from users with approaches taking account of only the positive examples [318, 319], others based on both positive and negative examples [320, 321], and approaches that operate with multiple “degrees of irrelevance” for each type of example [16, 322]. A novel approach was introduced with the D-EM [323] algorithm which makes use of unlabeled data to complement the labeled examples provided by the user.

Feedback approaches oriented towards enhanced browsing were developed in [25, 324] where the users are asked to re-arrange a layout of images on a panel (2-D space) according to their interpretation of relationships among the images. The machine is expected to learn the feature weighting scheme that could produce similar layouts.

5.3.3 Data models

The positive and/or negative examples provided by the user are the training set for the relevance feedback mechanism. Given a set of training examples, the role of a relevance feedback mechanism is to find the set of feature weights for which the clusters in the

training data can best approximate the semantic classification provided by the user. A large variety of approaches have been proposed for modelling the learning of training data. Early approaches [325, 320, 313, 16, 25] propose to learn a new query based on the relative importance of different features or feature components. Others model relevance feedback as a linear transformation in the feature space taking into account correlations among feature components [318, 319, 326]. More recent work approaches relevance feedback as a problem of density estimation [325], learning [321, 327, 328] or classification [329, 323].

In [83], the data collected during relevance feedback is dynamically clustered by tree-structured self organising maps (TS-SOM). The examples indicated by a user are assigned to clusters on a feature map which implicitly groups positive examples while dispersing negative examples. Feature vectors that fall within compact clusters are more relevant than vectors sparsely located within the map. A similar approach based this time on a probabilistic framework is proposed in [313] while the method introduced in [330] makes use of Kohonen maps and vector quantisation. In [327] the relevance feedback data is reorganised iteratively by a decision tree algorithm until similar feature vectors fall within the same class. In the resulting tree, images located close to a relevant leaf indicated by the user are considered relevant and returned in the retrieval step.

The Gaussian assumption is a common and convenient choice when modeling the distribution of the features of the target class within the data collection [318, 319]. With this assumption, learning the target class model corresponds to estimating the parameters of a Gaussian distribution. The approach presented in [331] takes account of negative examples for query update by comparing the variance of positive examples to the joint variance of positive and negative examples.

A novel approach to relevance feedback that emerged in the recent years relies on support vector machines (SVM [332]). Most SVM-based models of relevance feedback use a 2-class SVM classifier to discriminate positive and negative examples [333, 328, 334]. A simpler approach supporting only positive examples uses a 1-class SVM classifier [325]. However the 1-class SVM classifier returns an increased rate of non-relevant images within the retrieval results.

The use of relevance feedback does not stop at retrieval based on global representations. Schemes that employ relevance feedback in learning object structure from examples based on image segmentation have been proposed in the research literature. Pioneering work in

region based retrieval is introduced in [335] and [330]. An hierarchical formation scheme for object retrieval was proposed in [336]. The approach described in [337] uses a learning model to locate the area in feature space shared by all positive sub-images (regions) but far from all negative ones. A similar technique is developed in [338] where a model is inferred for a class of objects. In [339] objects are represented with attributed relational graphs (ARG) modeled from multiple samples via the EM algorithm. The ARG captures the probabilistic characteristics of both appearance and the structure of the object. An SVM-based approach that integrates several effective relevance feedback algorithms is presented in [340].

A retrieval approach based on an object ontology is presented in [341]. In this approach low-level descriptors for colour, position, size and shape of regions are associated with an object ontology in order to allow the qualitative definition of the high-level concepts (keywords). A relevance feedback mechanism, based on support vector machines using the low-level descriptors, is invoked to rank the potentially relevant image regions and produce the final query results.

5.3.4 Selection strategies

In most approaches present in the literature, the images on which feedback is elicited are images considered by the system as being potentially the most similar to the given query. However in a few cases these images were randomly selected. It could be argued that relevance feedback has two potentially conflicting goals: to provide the user with as many relevant images as possible while maximising the information obtained from the user regarding the relevant and non-relevant items.

Returning the most positive images focuses on the goal of providing the user with many items believed to be relevant early in the search. It has the advantage of attempting to increase users confidence in the retrieval performance and consequently to increase satisfaction in using the system. However, it may have the disadvantage of a slower system learning rate since the identification of the target image class can take longer in the absence of suitable discriminatory examples.

Returning the most informative images aims at maximising the information obtained from the user. An approach directed at identifying at every round images expected to remove

a maximal amount of uncertainty regarding the target is introduced in [314]. In this approach the images selected for feedback are those ambiguous at the current estimation. The complete set of items presented for feedback should have low redundancy among images.

Since both selection strategies mentioned above follow mainly a single goal, a hybrid selection strategy may be able to provide a good compromise. A hybrid selection method applied to text retrieval is presented in [342] where at every iteration a part of the retrieved documents are ambiguous items while the rest are items presumed relevant. The ratio of ambiguous documents decreases as the set of labeled examples expands.

5.4 Evaluation of retrieval with relevance feedback

Testing and evaluating relevance feedback mechanisms is a difficult and time-consuming task since it requires active cooperation of large groups of users in various experimentation contexts. A common evaluation alternative is to use a data collection for which the groundtruth - the set of image classes covering the collection's content - is known. Although the groundtruth for such an evaluation is somehow artificial since in reality different users would often catalogue the collection into different classes. Therefore, several groundtruth databases, of diverse content, will need to be used in order to cover a wide range of context. However, designing the groundtruth for large collections on images is a laborious effort and prone to subjective assessment.

In real-user experimentation, the feedback provided by users during search sessions can have a strong impact on the evaluation of the relevance feedback process. The interest and patience of users may run low when large amounts of explicit feedback is required. The psychological aspect of retrieval has received only limited attention. Some studies have concluded that users perceive attractive interfaces as being more efficient although this is not necessarily echoed in objective performance measures [343]. However an aesthetic interface may encourage the user to actually provide a high level of feedback with in turn can enhance the overall retrieval performance. Generally in evaluating retrieval it is difficult to quantify which part of the system actually influences the retrieval performance of the users.

The performance measure commonly employed for the evaluation of relevance feedback is *document cutoff level* - the proportion of relevant images returned in the top N search results.

5. Relevance feedback

The evolution of the measure during successive feedback iterations is an indication of the speed of convergence to the target ranking. Another measures of retrieval performance are *precision* and *recall* metrics, and the *precision vs. recall graph* (depicted in Figure 5.2) at a fixed number of feedback iterations [310]. The precision and recall measures are defined as:

$$\text{Precision} = \frac{\text{Number of relevant documents retrieved}}{\text{Number of documents retrieved}} \quad (5.4.1)$$

$$\text{Recall} = \frac{\text{Number of relevant documents retrieved}}{\text{Number of relevant documents in collection}}$$

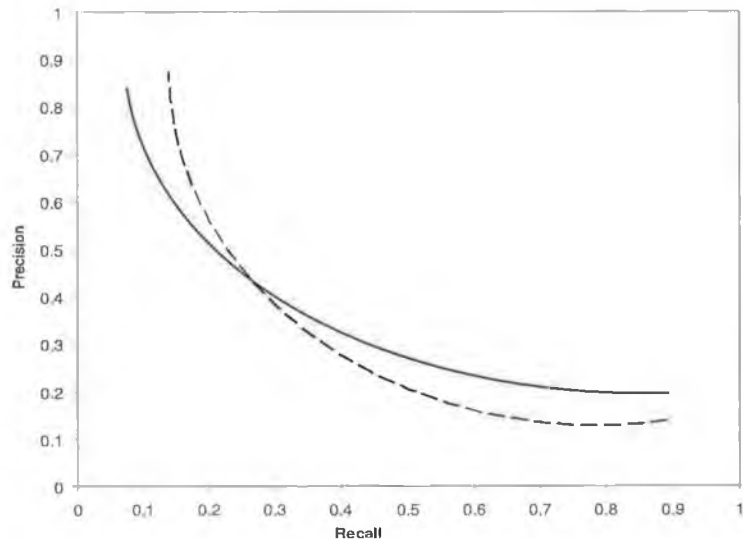


Figure 5.2: Illustration of a typical precision vs. recall graph

Since precision and recall are useful for expressing the performance of a system only in conjunction to each other, a combined measure can be sometimes more appropriate. The *E* and *F* (particularly the F_1 form) measures [310] are the commonly used combinations of precision and recall.

However, as demonstrated in [344] the evaluation of relevance feedback measured in terms of precision and recall is affected by the so called ranking effect. The ranking effect is the artificial improvement of precision and recall values as a result of re-ranking of the known relevant documents to the top of the document list. This effect blurs the actual

improvement feedback has on the retrieval of unseen relevant documents. Residual ranking and rank freezing are alternative evaluation techniques to measure the effect of feedback on the unseen relevant documents [344].

5.4.1 Residual ranking

In residual ranking the relevant and non-relevant documents used in relevance feedback are removed from the document collection before measuring the precision and recall values. Measuring the values on only the remaining (residual) collection this method accurately captures the effect feedback has on unseen relevant documents. However, the ranking on the residual collection is not comparable with the ranking before the feedback iterations, since now there are less documents than in the original collection.

The evaluation of relevance feedback is usually done on a set of queries, the precision and recall figures being computed over all retrieved documents on all queries. For a given query, at each successive iteration of feedback newly retrieved relevant documents are removed from the collection until finally there are no more relevant documents for that query. The query cannot be used in subsequent feedback iterations since it will return no relevant documents. This changes the number of queries over which precision and recall values are computed at different feedback iterations. Queries with large numbers of relevant documents and those with slow increase in feedback are likely to run for many iterations thus having a bigger impact on the measured performance.

5.4.2 Rank freezing

The rank freezing technique is based on preserving the rank position for a number of retrieved documents. There are two variations of this technique: full freezing and modified freezing. In full freezing the top N ranking documents, whose relevance has been judged, are preserved on the same position while the remaining documents are re-ranked. The precision and recall values are then computed over the entire ranking. Since only documents below the $N - th$ are re-ranked, changes in precision and recall happen only as a result of updates in the position of unseen relevant documents. In modified freezing the ranks are preserved to the position of the last marked relevant document [344].

As the number of feedback iterations increases more frozen ranks are accumulated, thus the frozen ranks will have a higher impact on the precision and recall values. Because of this the performance of relevance feedback at later iterations can appear as poor. Precision and recall values can be calculated even when all relevant documents are retrieved, but these figures will not change once all relevant documents are frozen.

5.5 Summary

This chapter provides an overview of relevance feedback in content-based multimedia retrieval. Relevance feedback covers a large range of techniques intended to facilitate retrieval of information relevant to a users information need. Conceptually, relevance feedback acts as a interface that hides the complexity of query modification from the user.

Relevance feedback has proved to be a useful and pragmatic solution to formulating an information need. However relevance feedback alone is not sufficient to dramatically improve retrieval. The variety of strategies and modalities adopted by users during search does not always find adequate support in the feedback process. Integration of relevance feedback with additional functionalities such as enhanced browsing and interaction is required in order to enhance the retrieval performance.

It has to be noted that not all relevance assessments are equal. Users are influenced by many objective and subjective factors when assessing relevance of a given document. The goal of the search, the particular moment in the search and overall knowledge related to the functionality of the retrieval system can largely change the selection of items which the user decides to label.

Chapter 6

An approach to object-based video retrieval

When we look at images we see objects and the relationship between them. It is well known that most people when describing a shot recall “things” (a tree, a car, house, etc) and relative positions of things although they may not remember specific details about these “things” such as their colour or exact shape.

6.1 Introduction

Since “things”, or in our terminology objects, are the units that people operate with why not look at searching image and video content by objects ? Thus, the question we ask is, whether visual retrieval can be effectively performed by finding objects and their relationship and this is an intriguing research problem. Although at the current state of technology, automated object extraction is not possible, except for few rigorously constrained applications, technology is always improving. Moreover, research aims at developing the technology of tomorrow and that can be done only by looking today a step ahead. It may well be that technology will develop some realistic and scalable mechanism for object detection at which point the work in this thesis will become very relevant.

It is the purpose of this thesis to constitute itself as a part of a larger stream of research that investigates visual objects, in their various representations, as a possible approach to

advancing the use of semantics in content-based visual retrieval. The aim of this work is to explore modalities of using objects in visual retrieval. In order to achieve this goal an object-based retrieval approach is presented and a set of experiments are carried out to investigate its performance and utility for interactive retrieval scenarios.

This chapter introduces the proposed approach detailing its component elements. The framework developed here contains the entire processing chain required to analyse, index and interactively retrieve images and video via object-to-object matching.

The remainder of this chapter is organised as follows. The next section gives an overview of the proposed framework. The first component in the framework, the video sequence segmentation algorithm is described in Section 6.3. The approach to object segmentation and the visual features used for object indexing are described in Section 6.4. The relevance feedback mechanism is presented in Section 6.5 followed by a presentation of the interaction with objects in the graphical user interface of the demonstration system developed in Section 6.6. Section 6.7 summarises the contents of the chapter.

6.2 Overall structure of the framework

The framework developed in order to explore the use of object-based retrieval follows the general structure of content based video retrieval systems with the notable difference being the interactive object-segmentation and the use of objects as illustrated in Figure 6.1.

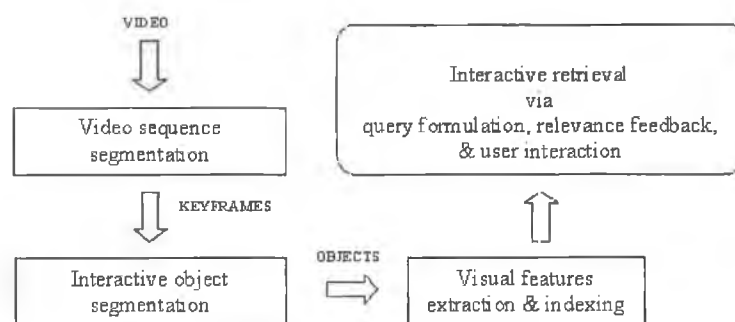


Figure 6.1: Framework diagram

The video sequence segmentation technique developed here takes advantage of the distribution of macroblock types in the MPEG compressed domain. The semi-automatic segmentation is designed around a scribble based interaction on pre-segmented keyframes obtained

from an automatic region segmentation approach. The extraction of MPEG-7 visual descriptors (Dominant Colour Descriptor and Texture Browsing Descriptor) for images and objects is performed with the code provided in the aceToolbox software [345]. The relevance feedback mechanism is modeled by a Gaussian mixture model incorporating positive and negative examples. The Graphical User interface facilitates interaction with objects and object based query formulation. The following sections describe each component in the framework.

6.3 Video sequence segmentation

This section introduces an efficient method to exploit the MPEG macroblock type information for video shot change detection. The approach can be classified in the general category of double threshold techniques for video segmentation.

During the motion estimation stage in MPEG encoders, each macroblock in a given frame is encoded according to one of the MPEG predefined types: intra-coded (I), forward compensated (F), backward compensated (B) or interpolated (FB). The statistical temporal distribution of the macroblock types in the bi-directional predicted (B) and forward predicted (P) frames, can reveal the shot transition. The computational complexity of the proposed method is reduced, as only the macroblock type information is extracted from the compressed video data.

Recent work on shot boundary segmentation makes use of additional information extracted from video compressed domain. An approach based on segmentation and classification of motion texture patterns in DC spatio-temporal slices is proposed in [346] and further improved in [347]. The method described in [348] exploits inter-frame dissimilarity on compressed domain low-level features. These features are used as input to an efficient k-Nearest-Neighbour classifier in order to detect shot transitions. A new approach for video cut detection which completely removes the impact of parameter and threshold settings is introduced in [349]. The basic idea of the approach is to classify the time series of frame differences into cuts and non-cuts by using the c-means clustering algorithm. An original approach to partitioning a video into shots based on a foveated representation of the video is proposed in [350]. The method works by computing, at each time instant, a consistency measure of the fixation sequences generated by an ideal observer looking at the video. A

unified detection model, both for abrupt and all types of gradual transitions is introduced in [351]. The innovation of this approach is centered on mapping the space of inter-frame distances onto a new space of decision better suited to achieving a sequence-independent thresholding.

Two approaches reported in the literature are particularly similar to the method proposed here, the first one using the spatio-temporal distribution of macroblock types for dissolve detection [124], and the second one tracking the shot change for particular macroblocks during the shot transition [352]. One specific difference is that both approaches use the spatio-temporal distribution of macroblock types as well as additional information about the discrete cosine coefficients for gradual transition detection, whereas our proposed method uses only the temporal distribution of macroblock types. Moreover, none of the above-mentioned work reports experiments over an extensive test set of real video sequences.

MPEG encoders compress video information into the following types of frames (pictures): intra-coded picture (I), forward predicted picture (P) and bi-directional predicted picture (B). Each frame is divided into blocks of 16 x 16 pixels called MacroBlocks (MB) [95]. Each macroblock contains information about its type of temporal prediction. A number of frames are grouped together in a Group of Pictures (GOP), which exhibits a typical encoding pattern. The most commonly encountered encoding pattern in a GOP has the *RBBR* structure, where R stands for the reference frame, which can be used for prediction, and B stands for the bi-directional predicted frame. The reference frames are always I frames or P frames, as the B frames cannot be used for prediction in accordance with the MPEG specifications. The MPEG video standard does not define a standard GOP structure thus different encoders may adopt different encoding patterns. However, it is extremely probable that the *RBBR* structure of the GOP implements the most efficient trade-off between video quality and compression gain.

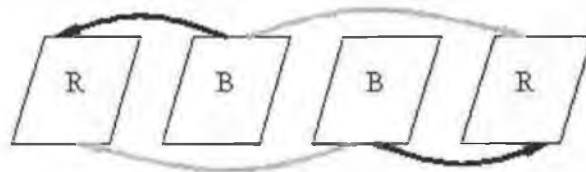


Figure 6.2: Prediction dependence within a neutral GOP

Within the GOP structure the B frames tend to be predicted from the nearest temporal reference frame, since they are more similar than the distant reference frame. As a result,

the first B frame will be predominantly forward and bi-directional predicted, whereas the second B frame will be predominantly backward and bi-directionally predicted. Figure 6.2 illustrates the macroblock prediction dependence within the GOP for a “neutral” (without transition) frame quadruplet.

6.3.1 Abrupt shot change detection

During an abrupt shot change, also called a cut, the image context is switched between two consecutive frames. Naturally, motion estimation in the MPEG encoders cannot extract much prediction between two frames with very different content. Therefore, the first frame of the new shot is not predicted from the previous frame, if it is a B frame. Similarly, if it is a P frame it cannot offer support for backward or bi-directional prediction. In the given frame quadruplet, a shot change can occur in any of the following positions: between the first and the second frame, between the second and the third frame or between the third and the fourth frame. If the shot transition occurs before the first or after the last frame of the quadruplet, the transition is manifested in the anterior or in the posterior quadruplet respectively. Each case poses a different scenario, as follows:

- ◇ When the shot change is between the first and the second frame of the quadruplet, both B frames’ predictions would be obtained from the last reference frame in the quadruplet. Only a small proportion of the macroblocks in each B frame would have forward or interpolated type.
- ◇ When the shot change is between the second and the third frame of the quadruplet, the initial B frames would have forward prediction and the final B frame would have backward prediction. Only a small proportion of the macroblocks in each B frame would be predicted by interpolation (FB type).
- ◇ When the shot change is between the third and the fourth frame, both B frames’ predictions would be obtained from the first reference frame. Only a small proportion of the macroblocks in each B frame would have backward or interpolated type.

The dependence of the prediction on the shot change position within the quadruplet is illustrated in Figure 6.3.

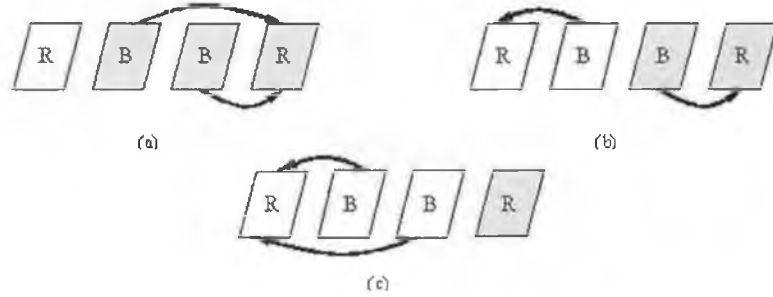


Figure 6.3: The prediction dependence in relation to shot transition (a) Shot front before first B frame (b) Shot front between the B frames (c) Shot front after last B frame

6.3.2 Threshold settings for abrupt transitions

Experiments were carried out on real video sequences with diverse content. The distribution of macroblocks types differs from sequence to sequence and encoder to encoder, especially under the influence of camera or object motion. Therefore, the method we use tracks the change in the distribution of each macroblock type, independently, and detects particular combinations of all macroblock types, which occur at the abrupt shot change. MPEG encoders mainly use a high ratio of interpolated macroblock type to encode B frames. Typically around 80% of the macroblocks within the B frames are interpolated macroblocks. A dramatic reduction of the number of interpolated macroblocks type within a frame quadruplet indicates a possible shot transition. Experimental results have indicated the following shot transition “rules” within a frames quadruplet:

- ◇ When the number of interpolated macroblocks drops to under 20% in the B frames, and the number of backward predicted macroblocks in the first B frame of the quadruplet is higher than the number of forward predicted macroblocks, an abrupt shot transition occurs between the first and second frames of the quadruplet.
- ◇ When the number of the interpolated macroblocks drops to under 10% in the B frames, but the B frames preserve their normal prediction dependence, an abrupt shot transition occurs between the second and third frames of the quadruplet.
- ◇ When the number of interpolated macroblocks drops to under 20% in the B frames, and the number of forward predicted macroblocks in the last B frame of the quadruplet is higher than the number of backward predicted macroblocks, an abrupt shot transition occurs between the third and the fourth frames of the quadruplet.

Flashlights may introduce false shot transitions. However, flashlight detection is straightforward as the shot covered by a flashlight is not usually longer than one or two frames.

6.3.3 Gradual shot change detection

Due to the variety of possible gradual transitions and to the variety of patterns for each specific transition, it is impossible to accurately detect and classify all gradual transitions. However, recognizing the exact type of transition would not represent significant additional information for video browsing and retrieval. Usually, a simple classification of the shot change in terms of abrupt and gradual transitions provides sufficient information. Therefore, our method only classifies the shot changes into abrupt or gradual transitions.

During gradual transitions shot change is progressively introduced from frame to frame. Thus, there will not be any abrupt alterations of the macroblock type distributions present. However, since parts of the image content are continuously changing, the difference between the predicted frame and its prediction reference frame will tend to increase. The effect especially manifests itself with the forward predicted frames (P-frames), as there is a gap of a few (usually two) frames between each P frame and its forward reference frame. As the gradual transition introduces new image content, which cannot be represented based on forward prediction, the number of intra-coded macroblocks within the forward predicted frames (P-frames) increases. Also, gradual transitions increase the dependence of the bi-directional predicted frames (B frames) towards their predominant prediction reference frame. Unfortunately, a similar macroblock type distribution pattern is exhibited during significant object motion or fast camera motion so this characteristic is of limited use in detecting gradual shot changes.

6.3.4 Threshold settings with gradual transitions

Due to different encoders, a variety of possible gradual transitions, and to the encoded content, a universal arithmetic relation describing the macroblock distribution within gradual shot changes is extremely difficult to find. Therefore, tracking a particular combination of the temporal distributions of the macroblock type is more successful for detection purposes. Typically, only a small number of intra-coded macroblocks appear in the encoded forward

predicted frames (P-frames), as sufficient forward prediction can be used within a neutral sequence.

Continuous changes during the gradual transition force the MPEG encoder to introduce intra-coded macroblocks in the predicted frames, in order to encode the newly introduced parts of the frame. An increased number (over 5%) of intra-coded macroblocks within the P-frames, combined with the reduction of interpolated macroblocks within the B frames, indicate the occurrence of a gradual transition [353].

6.3.5 Evaluation of shot change detection

Evaluation of the proposed method has been carried out on the TRECVID 2001 [4] shot boundary test set according to the TRECVID ground-truth. The experimental results are compared qualitatively and quantitatively with a classical shot change, histogram based, approach. A summary of the video content in the test set is presented in Table 6.1. Various statistics of the test set files are presented in Table 6.2.

Generally, uncompressed domain approaches to video segmentation exhibit higher accuracy for shot change detection. Due to its reduced number of features, approaches to video segmentation in the compressed domain are regarded as fast but not accurate. A comparison between uncompressed and compressed video processing approaches offers interesting conclusions. The histogram-based approach, which has been chosen for comparative evaluation, uses the Cosine similarity measure to compare colour signatures within frames. A detailed presentation of the algorithm can be found in [134].

Tables 6.3 and 6.4 show the experimental results obtained for the classical, uncompressed domain, histogram-based approach and for our proposed method respectively. The results are presented separately for abrupt transitions and for gradual transitions. The statistical performance of both methods is measured in terms of recall and precision as defined in Section 3.7.2.

The last column in both tables shows the processing speed of each method in terms of percentage from the actual real-time (playing time) of the video file. It can be seen from the last column in Table 6.3 that the compressed domain method performs at between 5% to 10% of real-time with the exception of short videos, such as those listed on rows 2 and

6. An approach to object-based video retrieval

File	Content description
File 1	Documentary related to aircraft hangar fire protection
File 2	Documentary about Blackpool Beach
File 3	NASA Anniversary Show
File 4	NASA Anniversary Show
File 5	Documentary about Glen Canyon Dam
File 6	Documentary related to utilization of the water
File 7	Documentary related to utilization of the water
File 8	Documentary about Rio Grande
File 9	Documentary about Lake Powell
File 10	Documentary about aerial lifts
File 11	Los Angels by car
File 12	Los Angels by car
File 13	Documentary about Nile River
File 14	Documentary related to space technology used for civil engineering
File 15	Documentary related to space projects
File 16	Documentary related to satellite cartography and habitat monitoring
File 17	Documentary related to space technology
File 18	Documentary related to airline safety and economy
File 19	Documentary about Elmina Fort
File 20	Documentary related to a flexible manufacturing workstation
File 21	Scientific lecture about human perception
File 22	Documentary about quality assurance in industry

Table 6.1: Content description of the test set

19. In the case of short videos the overhead with the actual loading of the video file into memory is a significant part of the entire processing duration, while for longer files this overhead becomes less important compared with the actual length of the video clip.

It is worthwhile to mention that the histogram-based approach is unable to classify the shot transitions as abrupt or gradual, and for gradual transitions the method reports more than one abrupt transition. Therefore, in Table 6.4 the results for gradual transitions are not included. Despite this deficiency, the comparison between the classical approach and the compressed domain method proposed here shows interesting results. Abrupt shot change detection accuracy and processing speed offers relevant comparison points. According to the experimental results, the proposed method offers satisfactory precision with the advantage of fast processing speed.

6. An approach to object-based video retrieval

File	Size [MB]	Time [min]	Frames	Transitions	Abrupt	Gradual
File 1	90.2	09:00	15679	107	62	45
File 2	7.0	00:37	943	2	2	0
File 3	66.9	06:19	11364	65	38	27
File 4	72.4	06:50	12307	103	38	65
File 5	240.5	26:56	48451	237	226	11
File 6	251.0	28:07	50569	528	375	153
File 7	149.4	16:44	30088	22	0	22
File 8	121.9	13:39	24550	135	0	135
File 9	247.2	27:41	49801	246	126	120
File 10	92.3	09:00	16048	81	61	20
File 11	48.8	04:25	6649	8	8	0
File 12	49.1	04:27	6688	7	7	0
File 13	14.5	01:18	1969	1	1	0
File 14	262.7	29:26	52927	298	181	117
File 15	260.1	29:08	52405	239	183	56
File 16	247.1	27:40	49768	214	188	26
File 17	128.0	14:20	25783	158	81	77
File 18	63.4	07:06	12781	67	44	23
File 19	4.4	00:24	601	1	1	0
File 20	84.1	08:15	14686	82	61	21
File 21	484.1	48:16	86789	308	292	16
File 22	128.1	12:23	22276	119	67	52

Table 6.2: Test set statistics - duration, size and transitions

The processing speed of the histogram-based approach is close to the playing time speed, as the video sequences need to be decompressed before analysis. The processing speed presented in Table 6.3 and 6.4 for both uncompressed and respectively compressed domain approaches have been obtained on a 733 MHz Pentium III PC with 256 MB RAM running Red Hat 7.0 Linux.

6.3.6 Extension to gradual transition classification

Our method classifies the shot changes into abrupt transitions and gradual transitions, using gradual as a generic definition for a series of possible gradual changes such as: dissolves, wipes, fades. Generally, each particular type of gradual transition has a pattern of changes,

6. An approach to object-based video retrieval

File	Abrupt detected	Abrupt inserted	Gradual detected	Gradual inserted	Recall abrupt	Precision abrupt	Recall gradual	Precision gradual	Speed [% of real-time]
1	62	9	44	28	100 %	87.3 %	97.7 %	61.7 %	5.58
2	2	0	-	-	100 %	100 %	-	-	18.92
3	38	2	27	14	100 %	95 %	100 %	65.8 %	5.54
4	31	0	63	31	81.6 %	100 %	96.9 %	67.7 %	5.61
5	224	52	11	9	100 %	81.3 %	100 %	55 %	5.63
6	344	0	150	86	91.7 %	100 %	98 %	64 %	5.57
7	-	-	22	12	-	-	100 %	64.7 %	5.68
8	-	-	131	71	-	-	97 %	65.5 %	5.62
9	125	3	115	75	99.2 %	97.7 %	95.8 %	61.5 %	5.46
10	61	0	20	14	100 %	100 %	100 %	58.8 %	5.56
11	8	0	-	-	100 %	100 %	-	-	5.67
12	7	0	-	-	100 %	100 %	-	-	5.48
13	4	0	-	-	100 %	100 %	-	-	8.97
14	181	8	114	62	100 %	95.8 %	97.4 %	65.1 %	5.49
15	183	30	56	37	100 %	85.9 %	100 %	60 %	5.53
16	188	22	26	19	100 %	89.5 %	100 %	57.7 %	5.60
17	81	7	76	39	100 %	92 %	98.7 %	66 %	5.58
18	44	2	23	13	100 %	95.7 %	100 %	63.8 %	5.63
19	1	0	-	-	100 %	100 %	-	-	29.17
20	61	14	20	9	100 %	81.3 %	95.2 %	68.9 %	5.66
21	286	0	16	5	97.9 %	100 %	100 %	76.2 %	5.87
22	65	0	51	31	97 %	100 %	98 %	62.2 %	5.65

Table 6.3: Performance of proposed method

which can be detected from the macroblock types. During dissolves, the intra-coded macroblock type appears to be randomly distributed over the frame surface and tends to follow a normal temporal distribution within the sequence. For the classical wipe transition, the intra-coded macroblocks type appears as a continuous line along the vertical, horizontal or diagonal side of the frame, shifted a constant number of positions within each frame. For fade detection the first DCT coefficient can indicate the continuous change in the sequence luminosity. Obviously, adding methods to detect each particular transition decreases the processing speed and does not always contribute with significant information to the image segmentation or the retrieval tasks. However, it would be difficult to classify combinations of gradual transitions.

6. An approach to object-based video retrieval

File	Abrupt detected	Abrupt inserted	Gradual detected	Gradual inserted	Recall abrupt	Precision abrupt	Recall gradual	Precision gradual	Speed [% of real-time]
1	62	12	-	-	100 %	83.8 %	-	-	100
2	2	0	-	-	100 %	100 %	-	-	100
3	37	5	-	-	97.4 %	88 %	-	-	100
4	38	7	-	-	100 %	84.4 %	-	-	100
5	223	35	-	-	98.7 %	86.4 %	-	-	100
6	372	43	-	-	99.2 %	89.6 %	-	-	100
7	-	-	-	-	-	-	-	-	100
8	-	-	-	-	-	-	-	-	100
9	126	14	-	-	100 %	90 %	-	-	100
10	60	9	-	-	98.4 %	86.9 %	-	-	100
11	2	0	-	-	25 %	100 %	-	-	100
12	1	0	-	-	14.3 %	100 %	-	-	100
13	1	0	-	-	100 %	100 %	-	-	100
14	179	28	-	-	98.9 %	86.5 %	-	-	100
15	182	9	-	-	99.5 %	95.2 %	-	-	100
16	186	16	-	-	98.9 %	92.1 %	-	-	100
17	81	8	-	-	100 %	91 %	-	-	100
18	44	3	-	-	100 %	93.6 %	-	-	100
19	1	0	-	-	100 %	100 %	-	-	100
20	61	4	-	-	100 %	93.8 %	-	-	100
21	288	22	-	-	98.7 %	92.9 %	-	-	100
22	66	9	-	-	98.5 %	88 %	-	-	100

Table 6.4: Performance of the histogram-based approach

6.3.7 Applicability to different encoding patterns

Our method has been developed and tested on video content encoded with bi-directional frame prediction, with and without interpolated macroblocks. The algorithm assumes an encoding pattern of two bi-directional predicted frames enclosed by a pair of reference frames (P or I frames). The method can perform successfully for a different number of bi-directional predicted frames enclosed within a pair of reference frames, with proper threshold settings. Assuming an even number of bi-directional predicted frames, each B frame receives dominant prediction from the closest reference frame. For an odd number of bi-directional predicted frames, the middle B frame prediction should be obtained almost equally from each reference frame.

Each MPEG encoder can use a particular weighting of the macroblocks types within the frames and consequently, different threshold settings provide the best results for each particular encoder. Therefore, the proposed method is optimal for applications that use video

data provided by a single encoder.

6.3.8 Keyframe extraction

Keyframes are intended to provide an intuitive visual description of video shots and pointers to their content. Although sophisticated algorithms for keyframe determination have been researched and some proved successful, there is no universal technique of selecting appropriate keyframes. Depending on the application, one or more keyframes can be extracted for every shot. A common approach taken in practice is the selection of a frame close to the middle of the shot.

In the approach presented here keyframes serve as content descriptions and also in outlining the semantic objects, thus the ideal keyframes are those containing entire non-occluded objects. Since the outlining of objects in the keyframes is done interactively in our work, keyframe selection can be undertaken by a human indexer who would decide on the most representative frame from a small list of automatically extracted keyframes for each shot.

6.4 Object outlining and visual feature extraction

Automatic segmentation of semantic objects is difficult and unreliable on generic content. This is especially evident in video due to compression artifacts, which affect homogeneity in colour and texture. To alleviate the effects of compression, object extraction is supported through user interaction in the work in this thesis. Every extracted object is indexed in terms of colour, shape and texture. This section details the object extraction and indexing process.

6.4.1 Object outlining

Interactive object extraction is performed in two steps: an automatic partitioning of the image into a number of uniformly coloured regions followed by a user assisted grouping of regions into semantic objects.

The automatic segmentation is a straightforward extension of the well known Recursive Shortest Spanning Tree (RSST) algorithm [354]. The RSST algorithm was selected because

it does not impose any external constraint on the image and also permits simple control over the number of segmented regions. The RSST represents an image as a weighted graph, where regions are considered as nodes and each pair of adjacent regions (R_i, R_j) is connected with a link L_{ij} . In the initial step every pixel in the image is considered a region. Regions are merged into an iterative procedure where at each iteration the two regions (R_i, R_j) with the minimum link distance $d_{L_{ij}}$ are joined. After each iteration the entire list of regions is updated. Repeating this procedure, the number of regions can be reduced down to one. The order of merging constructs a so-called spanning tree, depicted in Figure 6.4. In this picture the initial and segmented images are shown on the left side, the resulting RSST graph, with two branches for the background and, respectively, foreground areas, is shown on the right side. Each node in the graph is one of the regions outlined in the segmented image.

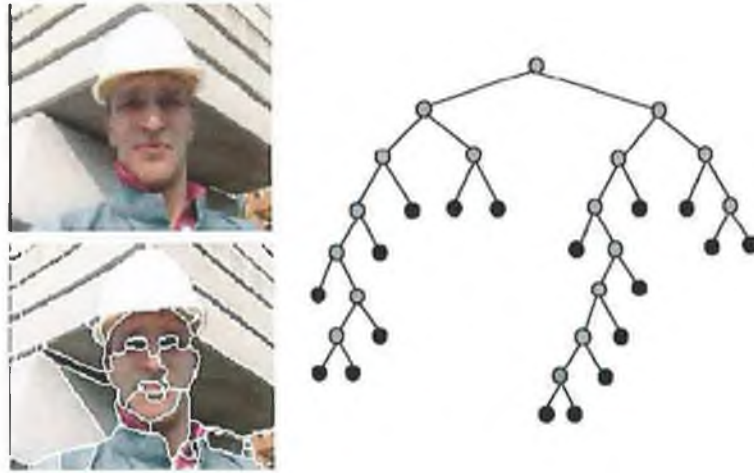


Figure 6.4: Results of RSST colour segmentation

The link distance between two regions is computed according to the expression:

$$d_{L_{ij}} = \frac{N_i \cdot N_j}{N_i + N_j} \sqrt{(Y_i - Y_j)^2 + (U_i - U_j)^2 + (V_i - V_j)^2} \quad (6.4.1)$$

where (Y_i, Y_j) , (U_i, U_j) , (V_i, V_j) are the luminance and chrominance values of the R_i and R_j regions respectively, and (N_i, N_j) are the number of pixels in the regions.

The merging procedure is stopped when a desired number of regions is obtained. However sometimes regions with very different colours are merged together because there is a strong

penalty for joining large regions, which was originally introduced in order to improve spatial homogeneity. The problem is particularly visible when a small number of final regions are desired. To overcome this problem the procedure is that when 255 regions are reached, further joining is done by computing a new distance in the HSV colour space according to the formula:

$$d_{L_{ij}} = \frac{1}{4}|S_i - S_j| + \frac{3}{4}|H_i - H_j|, \quad (6.4.2)$$

where (S_i, S_j) and (H_i, H_j) are the saturation and hue values of the R_i and R_j regions respectively.

The above formula does not discourage large regions, since spatial continuity was obtained in the first stage. Since hue is a circular space, hue operations are computed modulo 360° .

Illustrative results for the described approach are presented in Figure 6.4. This approach has been developed in the QIMERA platform [297] in the context of the EU-IST programme SCHEMA Network of Excellence in Content-Based Semantic Shot Analysis and Information Retrieval ¹.

In order to extract semantic objects, the user draws two coloured scribbles on the desired frame, one scribble over the foreground (object) and the other on the background parts of the image. Since the image is pre-segmented into uniform colour regions, the user's scribbles specify a number of regions that are then classified as background or foreground. If both scribbles touch the same region a conflict occurs for that region. In this case the region is deselected. Usually not all regions in the image will be selected by one of the scribbles. Unclassified regions are labeled iteratively by taking the label of the neighboring region with the nearest colour similarity. Scribble interaction is illustrated in Figure 6.5.

The outlined object is indexed in terms of colour, shape and texture as described below. Only one object is extracted per image.

¹<http://www.itl.gr/SCHEMA/>

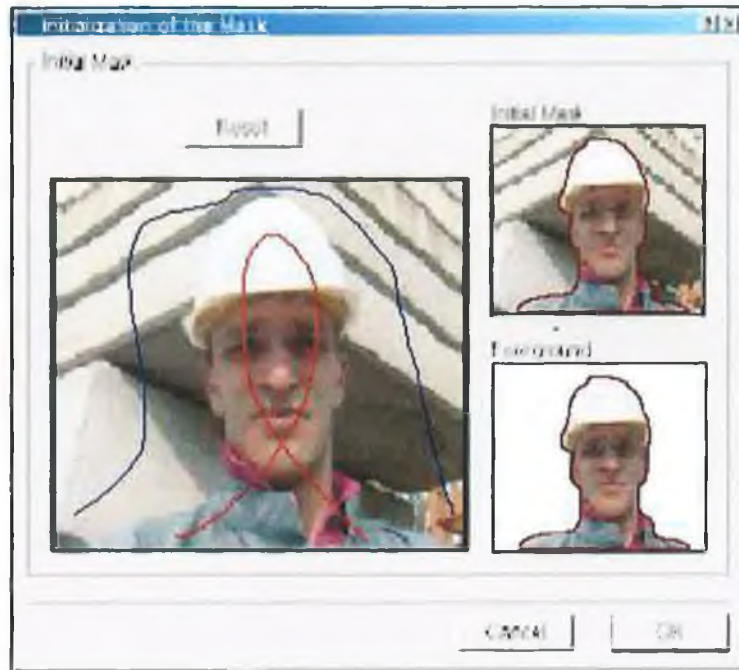


Figure 6.5: Scribble-based interactive segmentation

6.4.2 Dominant colour

The set of colours in an extracted object is represented with the *MPEG-7 Dominant Colour Descriptor* [115]. This descriptor provides an effective, compact, and intuitive representation of salient colours in any arbitrary shaped region. Conceptually the dominant colour descriptor is similar to the colour histogram, the difference being mainly in the number of bins used.

To compute this descriptor the colours present in a given object or image are first clustered in order to reduce the number of representative colours. The descriptor represents the colours and the proportion of these colours. According to the MPEG-7 standard the proportion of colours present in the objects should add up to 1. There are two optional parameters which can be included in the description: a colour variance value and a spatial coherency value. The spatial coherency value differentiates between large colours blobs and colours that are spread over the entire image. The descriptor is:

$$F = \{(c_i, p_i, v_i), s\}, \quad i = 1..N, \quad (6.4.3)$$

where: c_i is the i th dominant colour, p_i is the percentage value, v_i is the colour variance and s is the spatial coherency values.

The number of dominant colours, N , can vary for each object/image to a maximum of eight colours used in the representation. The percentage values are quantised to 5 bits. Typically, 3-4 colours provide a good characterisation of a region [115] or object. In the implementation of the descriptor we have used 4 colours to represent an object in order to maintain a uniform index.

The recommended distance measure between two dominant colour vectors $F_1 = \{c_{1i}, p_{1i}\}$ and $F_2 = \{c_{2i}, p_{2i}\}$ can be computed as [115]:

$$D_{DCD}(F_1, F_2) = \left(\sum_{i=1}^{N_1} p_{1i}^2 + \sum_{j=1}^{N_2} p_{2j}^2 - \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} 2a_{1i,2j} p_{1i} p_{2j} \right)^{\frac{1}{2}}, \quad (6.4.4)$$

where $a_{k,l}$ is the similarity coefficient between two colours c_k and c_l defined as:

$$a_{k,l} = \begin{cases} 1 - \frac{d_{k,l}}{d_{max}} & \text{for } d_{k,l} \leq T_d \\ 0 & \text{for } d_{k,l} > T_d \end{cases} \quad (6.4.5)$$

where $d_{k,l} = \|c_k - c_l\|$ is the Euclidian distance between the two colours (c_k, c_l) ; T_d is the maximum distance for two colours to be considered similar and $d_{max} = \alpha T_d$.

The values recommended in the MPEG-7 standard are T_d between 10-20 and α between 1.0-1.5. In the implementation of the descriptor we have used $T_d = 20$ and $\alpha = 1$.

6.4.3 Shape compactness descriptor

Shape is represented by the compactness moment of the shape computed as [107]:

$$\alpha = \frac{4\pi A}{P^2} \quad (6.4.6)$$

where A is the area and P is the perimeter of the shape. The value of α ranges between 0 (corresponding to a line segment) and 1 (corresponding to a perfect circle).

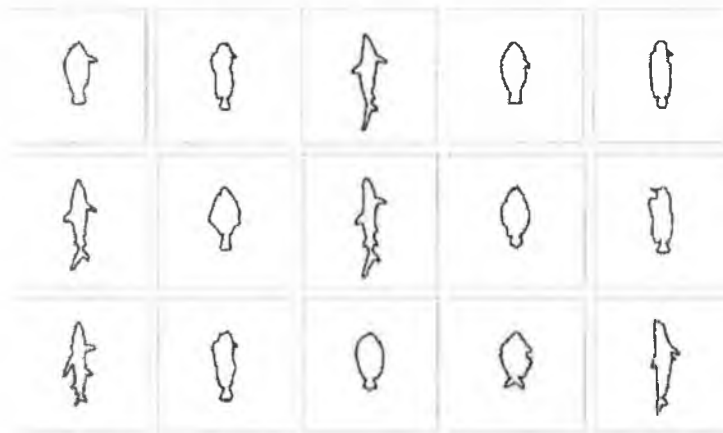


Figure 6.6: Shape variation within the same semantic class

Compactness is not a good discriminant descriptor on its own, however it is invariant to translation, scale and rotation, while providing some degree of differentiation between shapes. More complex shape descriptors are generally less robust to variances. Furthermore the shape of objects within the same class changes significantly depending on the position of specific attributes (e.g. the fins of the fishes illustrated in Figure 6.6), angle of camera, motion, occlusions, compression artifacts, etc. This assumes our eventual retrieval application needs to support rotation invariance and an implication of this is that boat-sails matched against fish bodies during the search experiments described in Section 7.3. A generic shape representation should therefore account for these possible variations.

6.4.4 Texture browsing descriptor

Texture is represented with the *MPEG-7 Texture Browsing Descriptor* [115] which characterises texture (illustrated in Figure 6.7) regularity, directionality and coarseness, based on filtering the image with a bank of 24 Gabor filters (6 orientations, 4 scales). The descriptor allows encoding for a maximum of two directions and coarseness values. The regularity is graded on a scale of 0-3, with 3 indicating textures with highly structured periodic patterns, and 0 indicating irregular or random textures. The directionality is quantised into six angular values between $0^\circ - 150^\circ$ in steps of 30° . A coarseness component is associated with each direction. Coarseness is quantised into four values 0-3, with a value of 3 indicating a very coarse texture and 0 indicating fine grain texture.

Since the texture descriptor is computed on rectangular blocks the arbitrarily shape of

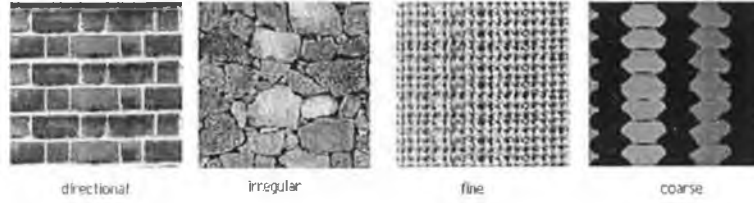


Figure 6.7: Texture examples

the object is approximated with a uniform grid of rectangular blocks. Incomplete blocks are dropped from the object mask in order to avoid introducing changes in the texture, which would be done if these blocks also contained background pixels or were padded with foreground pixels.

6.5 A Gaussian mixture model for relevance feedback

In our proposed approach to retrieval, the distribution of relevant images is modelled using a Gaussian mixture model. The Gaussian distribution assumption for relevance feedback is a reasonable model for images around a local query point, having been successfully applied to image retrieval as discussed in [16]. The well-defined mathematical formulation and the computational tractability makes the Gaussian model a commonly used paradigm in feedback and support vector learning. In this section we present a method to estimate the number of mixture components based on both positive and negative examples. To reduce the problem of small training samples unlabeled data are used in the estimation of covariance matrices.

6.5.1 Gaussian mixture model

Gaussian Mixture Models (GMM) are intensively used for clustering and density estimation. The GMM models a data distribution as a mixture of N Gaussian components and can be expressed as:

$$p(\omega|x) = \sum_{n=1}^N \alpha_n G(x, \mu_n, \sigma_n) = \sum_{n=1}^N \alpha_n \frac{1}{(2\pi)^{\frac{n}{2}} |\theta_n|^{\frac{1}{2}}} \exp \left[-\frac{1}{2} (x - \mu_n)^T \theta_n^{-1} (x - \mu_n) \right] \quad (6.5.1)$$

where α_n is the mixing parameter satisfying $\sum_{n=1}^N \alpha_n = 1$ and $G(x, \mu_n, \sigma_n)$ is the probability density function corresponding to the n -th Gaussian component.

The parameters to be estimated in the Gaussian mixture are: α_n , μ_n and θ_n ($n = \overline{1, N}$). Estimation of these parameters is usually done using the Estimation Maximisation (EM) algorithm [221]. However, the EM algorithm requires a priori selection of the number of mixture components which is generally unknown in the retrieval process. A solution to this problem is suggested in [355] where the positive examples are grouped in the feature space in clusters perfectly separated from the clusters of negative examples. Therefore, the number of mixture components is taken as the minimum number of components for which the positive and negative clusters are separated in the mixture. Here we have adopted this approach to estimate the number of components in the Gaussian mixture.

6.5.2 Estimation of the mixture's components

Both positive and negative examples are used in the estimation of mixture components. Given two sets of examples, a positive set $I^+ = \{I_1^+, I_2^+, \dots, I_K^+\}$ and a negative set $I^- = \{I_1^-, I_2^-, \dots, I_L^-\}$ we can estimate the number of clusters N to cover the positive set. The clusters can be denoted as $C_+ = \{(C_n^+, R_n^+); n = \overline{1, N}\}$, where C_n^+ and R_n^+ are the centre of the cluster and radius of the n -th cluster.

The clusters are constructed by selecting a positive sample I_i^+ and setting it as the centre of a new cluster C_n^+ . The radius of the cluster is computed based on the distances to positive and negative examples:

$$\begin{aligned} d^+ &= \max_j D(I_i^+, I_j^+) \\ d^- &= \max_j D(I_i^+, I_j^-) \end{aligned} \tag{6.5.2}$$

Then the radius is determined as:

$$R_i = \begin{cases} \frac{(d^+ + d^-)}{2} & \text{for } d^- \leq d^+, \\ \rho \cdot d^- & \text{otherwise} \end{cases} \tag{6.5.3}$$

where ρ is a constant satisfying $0 < \rho < 1$, such that a cluster does not extend too close to a negative example.

The positive examples assigned to a cluster are removed from the set of unassigned examples and the clustering process is repeated until all positive examples are assigned. The choice of initial positive example has little influence over the construction of clusters. A drawback of the approach appears when there are no negative examples, since then the radius of the positive clusters can grow unlimited. This can be addressed by imposing limitations over the maximum radius of clusters.

6.5.3 Estimation of mean and covariance matrix

The mean vector of each Gaussian component is obtained by averaging all positive samples in a corresponding cluster:

$$\mu_k = \frac{1}{m_k} \sum_{i=1}^{m_k} x^{(k)}, \quad k = 1..K \quad (6.5.4)$$

where m_k is the number of positive examples in the k -th cluster.

Estimation of the covariance matrix is unreliable due to the small number of training samples. However, the covariation matrix is reduced to a diagonal matrix by assuming that all dimensions of the feature vector are independent of each other. Unlabeled data (objects/images) that fall within a cluster's radius are used together with the positive examples to estimate the covariance matrices. Combining the labeled and unlabeled data has been proven useful in compensating for a small training set [326]. The covariance matrix is computed as:

$$\sigma_{jj}^k = \sqrt{\frac{1}{M_k} \sum_{i \in C_k} (x_{ij} - \mu_j)^2} \quad (6.5.5)$$

where M_k is the total number of positive and unlabeled data falling in the cluster C_k .

6.5.4 Weighting of mixture components

The weight of a component in the mixture is computed from the number of positive sample falling in the corresponding cluster according to the expression:

$$\alpha_k = \frac{m_k}{P^+} \quad (6.5.6)$$

where m_k is the number of positive examples in the cluster and P^+ is the total number of positive examples.

In the retrieval step, for each document in the database the probability of being relevant is calculated according to (6.5.1). When only a single positive example is available, which may happen in an initial query, retrieval is done by computing the similarity to only this sample.

6.6 Object interaction interface

The graphical user interface² (GUI) is an important component of any visual retrieval system since it is the component that links the retrieval engine to the user. In our work the GUI serves in the query formulation process and the presentation of search results.

6.6.1 Interacting with objects

Query formulation is the core user interaction required to achieve more accurate search through iterative refinement of object modeling. Relevance feedback occurs each time a user formulates a query to search objects. Since the emphasis in this thesis is on semantic object retrieval the GUI is designed to facilitate interaction with objects.

Figure 6.8 shows a screen shot from the system in which the interface is divided into four columns. The first column is the browsing area where the user can browse the entire collection. In this column the images in collection are listed by name. The user browses

²The GUI was designed by Dr. Hyowon Lee at Centre for Digital Video Processing, Dublin City University

6. An approach to object-based video retrieval



Figure 6.8: Screen-shot of the GUI

this set of images, views objects and specifies features, then adds some of the objects to the “QUERY OBJECTS” panel (second column). A similar interface facility can be found in numerous experimental image and video retrieval systems in which the user can select example images to be used for subsequent queries as a mechanism for relevance feedback. However, the added examples in this system are objects, not whole images or an image region.

The user can highlight an object of interest by selecting the toggle red button found in the right side of each image. After selecting an object the user can then specify which low-level features (colour, shape or texture) of the specified object s/he is interested in. Each of the feature buttons toggles between positive, negative or neutral for each feature of the object. Once feature indications are specified, the user can copy this object (and its specified features) to the query panel as shown in the second column in Figure 6.8 where the image contains only the specified object with the background stripped away. The feature specification for this object will be now used for relevance feedback. Figure 6.8 currently shows 7 objects added to the query panel. Clicking on the “FIND” button triggers retrieval

based on the 7 objects and the positive, negative or neutral indicators of their features, and the result is presented on the “SEARCH RESULT” panel (3rd column). If a relevant object is found in the search result, the user can save it to the “SAVED OBJECTS” panel (4th column). The user can also add more objects to the query panel from the search result, or from the saved object panel.

6.6.2 Refinement with query branching

In addition to the above, an important feature of our system is to allow the user to view how his/her relevance feedback and set of query objects is semantically consistent/inconsistent by showing the clusters within the set of query objects. If this set of query objects is not visually consistent, using all of this feedback for retrieval will confuse the system and lower retrieval accuracy. This is similar to adding very visually different image examples in Query-By-Example systems. Although a syntactically legitimate action by the user, this behaviour results in degraded retrieval and thus contributes negatively to the interaction. Thus, the clusters resulting from the clustering process described in section 6.5.2 are presented to the user, which can branch the query into two or more sub-queries corresponding to each cluster and then focus on only one of the sub-queries at a time. This enables a model of retrieval where a user wishes to pursue two or more lines of enquiry. At the top of query panel (2nd column), the user can click on the “GROUP” button to view how the system can internally split objects in the query panel. This split of query objects is displayed in Figure 6.9. The user can enable or disable this system feature as s/he wishes.

With the approach in Figure 6.9, the system has partitioned the search into two distinct clusters, one can be pursued as the set of query objects while the other cluster(s) is put on hold and returned to at a later stage. In Figure 6.9, the 7 objects the user added to the query have been split into 2 groups according to the system’s clustering algorithm. The user can now see how s/he has been adding objects of two different types: in the first group (top 4 objects in the second column), the object characteristics indicate white colour, more square shaped vehicles such as a white jeep and in the second group (bottom 3 objects in the second column), the object characteristics indicate red, round shaped vehicles such as a VW Beetle, quite different from the one formulated in the first group. As this split among the added objects is now revealed to the user, s/he can decide to focus on searching for only one type of object (either first or second group) to find more objects that are like only one of the groups.



Figure 6.9: Screen-shot of the GUI query branching functionality enabled

In this way, the user can see semantic clustering of query objects as s/he adds and specifies the features of objects, and can conduct a more multi-threaded search by pursuing one of the clusters of query objects at a time. Inconsistent relevance feedback is still a legitimate possibility by the user, but the system is adaptive in that it suggests a feature-oriented way of searching by automatically splitting the relevance feedback history into semantically coherent clusters, so that the user can continue with a more consistent subset of his/her own feedback objects and can search query object cluster one at a time.

6.7 Summary

In this chapter we introduced and described a framework designed for retrieval of semantic objects from image and video content. The framework uses objects as the unit of retrieval based on low-level visual features. The component elements of the framework are detailed.

The framework contains the entire processing chain required to analyse, index and interactively retrieve images and video via object-to-object matching. The presentation covers aspects related to video shot detection, semantic object segmentation, visual feature extraction for objects, modeling of feedback data, and description of user interaction.

The video sequence segmentation developed here takes advantage of the distribution of macroblock types in the MPEG compressed domain. The semi-automatic segmentation is designed around a scribble based interaction on pre-segmented keyframes obtained from an automatic region segmentation approach. Each object is represented with three features: the dominant colour, the shape compactness and the texture browsing descriptors. The relevance feedback mechanism is modeled by a Gaussian mixture model incorporating positive and negative examples. The Graphical User interface facilitates interaction with objects and object-based query formulation.

In the next chapter we will present a set of experiments that investigates the performance of the proposed framework and how real users exploit object-based searching within this framework.

Chapter 7

Retrieval experiments

7.1 Introduction

In the task of automatically segmenting and indexing objects in image and video content, the main difficulty is the diverse manifestation of an object in the image/video regardless of the object's inherent visual features such as colour, shape and texture. Due to factors such as different lighting conditions, different angles taken by the camera, and the degree and types of occlusions that often occur on objects, this makes the actual segmentation of an object as well as labeling the segmented object, for example a car, extremely difficult. This same problem of diverse manifestations of an object also occurs when a searcher has to give examples of an object during query formulation. This problem is alleviated through the use of semi-automatic segmentation.

As query formulation is the key element for getting feedback from the user in our approach, the framework we have built incorporates a user interaction strategy in which a user can interact with segmented objects by way of highlighting them, selecting them, and then using them in subsequent query formulation. The framework tested here does not rely on simple matching of an object from a query image against objects from a video keyframe, but uses a selection of a set of objects in a query as the basis for retrieval.

In this chapter we describe an investigation conducted using our retrieval framework. The purpose of this investigation is to evaluate object-based retrieval and to explore how real

users exploit object-based functionality in their searches. To this purpose an evaluation of object-based search against standard image-based search has been carried out in an interactive experiment with 24 search topics and 16 users each performing 12 search tasks on 50 hours of rushes video. A smaller scale experiment also described in this chapter, explores the retrieval performance of our proposed framework in terms of precision and recall.

7.2 Investigation of object-based retrieval performance

In this experiment we aimed to investigate the retrieval performance of object-based search. For this purpose we have developed an experimental system based on the object retrieval approach detailed in the previous chapter. A novel facility offered by this system is the automatic query branching as a means to provide the user with knowledge about the distribution of object features in the video collection. This is a two way feedback where the system is instructed about the relevance of retrieved objects and the user receives explicit indications about the mapping of the query into the feature space. By being aware of the ramifications that a query has on the collection space, the user can better adapt the query and their feedback to more accurately select query objects relative to their information need.

7.2.1 Experimental setup

Our system processes one object from each keyframe taken from each shot in the video and stores these in a database to be used in the retrieval process during an interactive search session. We use keyframes automatically extracted from the TRECVID 2003 test corpus, as well as images from the well known Corel test corpus. For each keyframe, the interactive object segmentation tool described in Section 6.4 was used to accurately segment one main object in the image. Once segmented, each object is automatically indexed by colour, shape and texture using the following descriptors: dominant colour descriptor, the compactness moment of the shape and the texture browsing descriptor as described in Section 6.4. This completes the off-line object segmentation and indexing process. Determining similarity among objects for retrieval purposes is done during interactive search without pre-computation as the system progressively receives more information from the user.

7. Retrieval experiments

Query formulation is the core user interaction required to achieve more accurate search through iterative refinement of object modeling. Relevance feedback occurs each time a user formulates a query to search for objects. After selecting an object, the user can then specify which low-level features (colour, shape or texture) of the specified object s/he is interested in. Each of the feature buttons toggles between positive, negative or neutral preferences for each feature as depicted in Figure 7.1. Collecting feedback on each feature independently allows for faster convergence on the search target.

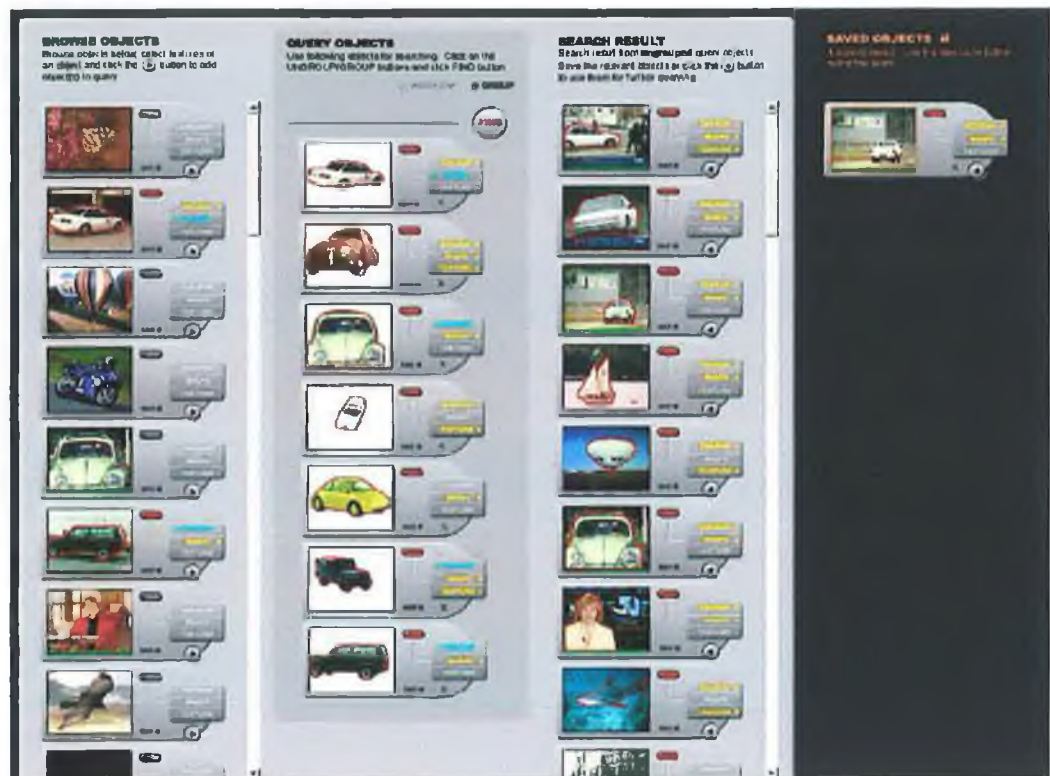


Figure 7.1: Graphical user interface screenshot

7.2.2 Experimental procedure

In order to evaluate the performance of the system we designed a retrieval experiment using 12 classes of objects, each class containing 50 objects, thus there were 600 images in the database. Although the number of images is small, it should be noted that each object requires interactive segmentation, thus the effort of building a ground-truth is higher than in the case of using whole images. The objects classes used are: balloon, boat, butterfly, car, eagle, flower, horse, motorcycle, people, plane, shark, tiger.

7. Retrieval experiments

Experiments were performed with an expert user selecting an initial query object and providing negative/positive feedback. For each query iteration a positive example was added in the query formulation, a negative example was added every second iteration. The query session for each object class was conducted for 5 iterations, therefore for each object class 5 positive examples and 2 negative examples were provided over 5 iterations. This can be considered a type of pseudo-feedback because the number of iterations and of positive/negative examples provided has been uniform across all classes of objects. However, since the experiment did not involve real users who did not have knowledge about the system internals, allowing the expert searcher to vary the amount of feedback for each class would have certainly biased the results.

The metrics used in the evaluation of the system performance were precision-recall graphs. The mean precision-recall curves obtained are shown in Figures 7.2.A, 7.2.B, 7.2.C and 7.2.D. Since representing 12 curves on the same graph becomes confusing, we present the precision-recall curves grouped on four sub-graphs for every three classes taken in alphabetical order. In order to provide an easy comparison between object classes, each sub-image contains the mean precision versus recall curve computed by averaging the results over the entire 12 classes.

7.2.3 Refinement with query branching

An important feature of our system is to allow the user to view how his/her relevance feedback and set of query objects are semantically consistent/inconsistent by showing the low-level clusters within the set of query objects. The clusters resulting from the clustering process described in section 6.5.2 are presented to the user who can branch the query into two or more sub-queries corresponding to each cluster and then focus on only one of the sub-queries at a time. This enables a model of retrieval where a user wishes to pursue two or more lines of enquiry as depicted in Figure 7.3. In this way, the user can see semantic clustering of query objects as s/he adds and specifies the features of objects, and can conduct a multi-threaded search by pursuing one of the clusters of query objects at a time. The user can enable or disable this system feature as s/he wishes.

In order to evaluate the usefulness of query branching functionality we compared the speed of convergence to the target ranking between query-branching and normal search modes. Comparison using precision-recall graphs of query-branching mode against normal mode

7. Retrieval experiments

would not be satisfactory since performance of query-branching depends on the specific query cluster activated by the user. Particular query-clusters have different retrieval performances. Also it is not obvious for a human evaluator which objects map into which cluster, therefore it is extremely difficult to create an accurate ground-truth.

Figure 7.4 shows the average increase in the number of relevant documents per iteration for two classes of objects: horse and car. These two classes have been selected because intuitively they should have the largest variation in low-level features, due to various colour and shape possibilities taking into account camera angle, car models and body motion for horses, among the 12 classes featured in our database. The average increase is computed over a number of 5 different search sessions. In query-branching mode, for every iteration results were generated for all available sub-queries, thus the results shown for query-branching mode are the combined results of all sub-queries taken separately. At each iteration a positive and a negative example were added to the query.

7.2.4 Results interpretation

The precision-recall curves show a relatively slow decay with increasing recall. The optimal values seem to be located around values of recall of 15-25 images/objects out of 50 objects per class, which seems to prove the effectiveness of the presented system. However, it is premature to generalise from results obtained on such a reduced dataset. More insight into the performance of object-based retrieval would be obtained by performing comparisons against other retrieval systems on a common test set and with multiple users in the retrieval loop.

The difficulty of conducting an extensive investigation on the current framework lies in the effort required to interactively segment objects from images and video. Although this is not an inconvenience in online searching where a user will outline very few objects, it becomes quite a significant workload when done for thousands of objects. The work presented here is not an exhaustive investigation of object-based retrieval, but one of a small number of early attempts, as far we are aware, to incorporate semantic objects in content-based retrieval.

In the query-branching mode the combination of sub-query results constantly outperformed the wide query where all user examples are employed together. This is because in the wide query the retrieved images compete for ranking among all Gaussian clusters in the query

model, and this may push a low scoring but relevant object retrieved by one such cluster in a ranked position below a high scoring but non-relevant object retrieved by an other Gaussian cluster.

7.2.5 Discussion

In its present form our system may not be suitable for a realistic video retrieval context, but the point of developing it was to demonstrate how an object-based query formulation mechanism could be realised to help dynamically refine the object model in the database and enhance retrieval. The experimental results indicate that object-based search provided a relatively good performance of retrieval. Although it is difficult to compare these results to a system performing retrieval on whole images, since the query formulation method is structurally different, the investigation reported here provides a measure of the potential of object-based retrieval.

7.3 Investigation of object-based searching

In these experiments we aim to investigate how real users make use of object-based search functionality in contrast with image-based search. For this purpose we asked the users to perform a set of video searches with our system using two identical looking search interfaces (like the one depicted at the bottom right in Figure 7.5). In one interface, allowing object-based search, users could use a combination of object and image searching, whereas in the other interface they were restricted (by disabling the object functionality) to using only whole image searching. For the remainder of this chapter we refer to these interfaces as the object-based interface and respectively, the image-based interface. The task given to the users is to find as many relevant shots for each predefined topic as possible. The effectiveness of a search interface is regarded as proportional to the number of relevant shots retrieved with that interface.

Each user was asked to perform a set of 6 separate search tasks with the object-based interface and a different set of 6 search tasks using the image-based interface. The users selected seed images from the Google image search engine¹ and could semi-automatically segment

¹<http://images.google.com>

7. Retrieval experiments

objects in these images if they considered them useful for their search. The segmentation step is not performed when using the image interface. Users were instructed to save all relevant shots retrieved. At any stage during the search the user can add or remove images from the query, either from the retrieved images or from the Google image search.

We allocated only a 5 minute period for task completion for each of the 12 searches completed by each user. The objective of the time limit is was to put participants under pressure to complete the task within the available time. Users were offered the chance to take a break at the session's halfway point should they feel fatigued.

7.3.1 Experimental setup

The test corpus used in this experiment is the TRECVID 2005 BBC Rushes collection [356]. The system begins by analysing raw video data in order to determine shots. From the 50 hours of BBC rushes video footage we detected 8,717 shots, or 174 keyframes per hour, much less than for post-produced video such as broadcast TV news. For each shot we extracted a single keyframe by examining the whole shot for levels of visual activity using features extracted directly from the video bitstream [357]. Rushes video is raw video footage, which is unedited and contains lots of redundancy, overlap and wasted material in which shots are generally much longer than in post-produced video. The regular approach of choosing the first, last or middle frames as the keyframe within a shot would be quite inappropriate given the amount of “dead” time that is in shots within rushes video. Thus an approach to keyframe selection based on choosing the frame where the greatest amount of action is happening seems reasonable, although this is not always true and is certainly a topic for further investigation.

Each of the 8,717 keyframes was then examined to determine if there was at least one significant object present in the frame. For such keyframes one or more objects were interactively segmented from the background using the segmentation tool described in Section 6.4.1. This process is very quick for a user to perform, requires no specialist skills and yielded 1,210 such objects since not all keyframes contained objects.

Once the segmentation process was completed, we proceed to extract visual features. The visual descriptors used in this experiment: Dominant Colour, Texture Browsing and Shape compactness, were introduced in Section 6.4. We extracted dominant colour and texture

browsing features for all keyframes and dominant colour, texture browsing, and shape compactness features for all segmented objects. This effectively resulted in two separate representations of each keyframe/shot. From the number of keyframes and objects extracted we pre-computed two 8,717 x 8,717 matrices of keyframe similarities using colour and texture for the whole keyframe and three 1,210 x 1,210 matrices of similarities between those keyframes with segmented objects using colour, texture and shape.

In order to kick-start a search we ask the user to locate one or more images from outside the system using some other image searching resource. In this experiment our users used Google image search² to locate such external images, but any image searching facility could be used. Once external images were found and downloaded they were analysed in the same way as the keyframes, and the user was allowed to semi-automatically segment one object (as described in Section 6.4.1) in the external image if they wished.

When these seed images were ingested into our system the user was asked to indicate which visual characteristics make each seed image a good query image - colour or texture in the case of the whole image and colour, shape or texture in the case of segmented objects in the image. Once this was done, the set of query images were used to perform retrieval and the user presented with a list of keyframes from the archive. For keyframes where there is a segmented object present (1,210 of our 8,717 keyframes) the object is highlighted when the keyframe is presented.

The user was asked to browse these keyframes and can either play back the video, save the shot, or add the keyframe (and its object, if present) to the query panel and the process of querying and browsing can continue until the user is satisfied. The overall architecture of our system is shown as Figure 7.5.

7.3.2 Search topic formulation

As described earlier in this section, running shot boundary detection on the rushes corpus returned 8,717 shots with one keyframe per shot. 1,200 representative objects were selected and subsequently extracted from these keyframes. For this experiment we required a set of realistic search topics. We based our formulation of the search topics on a set of over 1,000 real queries performed by professional TV editors at RTE, the Irish national broadcaster's

²<http://images.google.com/>

video archive. These queries had previously been collected for another research project in our research group.

The BBC rushes corpus consists of video recorded for a holiday program. We played through all the video and then eliminated queries which we knew could not be answered from the rushes collection. We then removed duplicate queries and similar, subsumed or narrow topics, ending with a set of 26 topics for which it is likely to find a reasonable number of relevant shots within this collection. Of these, 24 topics were used as search tasks and the other 2 as training during our users' familiarisation with the system. In the selection of search topics we did not consider whether they would be favorably inclined towards a particular search modality (object-based or image-based).

For each search topic the users were given a textual formulation of the query such as "find keyframes showing people walking on the beach" or "find keyframe depicting nightclub life". No visual examples (images) of the topics were provided to the users since we assume the content of the archive to be unknown, thus we cannot limit the user to a particular depiction of the topic. Moreover, whether a particular image represents a topic is a subjective matter. Examples of images employed by users as initial queries for the above topics are illustrated in Figure 7.6 and examples of retrieved images for different search topics are presented in Appendix B.

7.3.3 Experimental design methodology

In our experimental investigation we followed the guidelines for design of user experiments recommended by TRECVID³. These guidelines were developed in order to minimise the effect of user variability and possible noise in the experimental procedure. The guidelines outline the experimental process to be followed when measuring and comparing the effectiveness of two system variants (object/image based search versus image-only based search) using 24 topics and either 8, 16 or 24 searchers, each of whom searches 12 topics. The distribution of searchers against topics assumes a Latin-square configuration where a searcher performs a given topic only once and completes all work on one system variant before beginning any work on the other variant.

³TRECVID Evaluation, available at <http://www-nlpir.nist.gov/projects/trecvid>

7. Retrieval experiments

We chose to run the evaluation with 24 search topics and 16 users, with each user searching for 12 topics, 6 with the object/image based search and another 6 with the image-only based search. Our users were 16 postgraduate students and postdoctoral researchers: 8 people from within our research group with some prior exposure to video search interfaces and video retrieval experiments and another 8 people from other research fields with no exposure to video retrieval. Topics were assigned randomly to searchers. This design allows the estimation of the difference in performance between the two system variants free from the effects of searcher and topic.

7.3.4 Experimental procedure

In order to accommodate the schedules of users we ran experimental sessions with 4 users at a time. The search interface and segmentation tool were demonstrated to the users and we explained how the system worked and how to use all of its features. We then conducted a series of test searches until the users felt comfortable working with the retrieval system. Following these, the main search tasks began.

Users were handed a written description of the search topics. The topics were introduced one at a time at the beginning of each search task such that users would not be exposed to the next search topic in advance. This was done in order to reduce the influence that the current query and retrieved shots may have in revealing clues for the subsequent search topics. As previously stated, users were given 5 minutes for each topic and were offered the chance to take a break after completing 6 search topics.

Each individual's interactions were logged by the system and one member of our team was present for the duration of each of the sessions to answer questions or handle any unexpected system issues. The results of users' searching (i.e. saved shots) were collected and formed the ground-truth for evaluation. The rationale behind doing this is that the shots saved by a user are assumed to be relevant and in terms of retrieval effectiveness for each system we measure how many shots, all assumed to be relevant, the user managed to locate and to explicitly save as relevant. For each topic we collected a time-stamp log of the composition of each search at each iteration.

7.3.5 Evaluation metrics

Since we did not have a manual relevance ground-truth for our topics, we assumed the shots saved by users during the same topic search to be relevant and used them as our recall baseline. Although we do not have any independent third party validation of the relevance of the saved shots our users were under instruction to only save shots they felt were relevant to the search topic, so this is not an unreasonable assumption. Naturally there may be other relevant shots in the collection which were not retrieved by our users, but in the absence of exhaustive ground-truth we cannot know how many such shots are there. However our goal was to observe how real users make use of the object-based search functionality and that can be inferred even without an absolute ground-truth.

The shots saved by the users were checked by the experiment coordinator against inclusion of obvious irrelevant images in order to ensure that the relevance set is reasonable. This was done by visually inspecting all saved keyframes for each topic and such images were removed. The independent verification of each each topic against the entire video collection was not feasible given the amount of data, the number of topics and users, and the inherent subjectivity of such a relevance assessment on a collection of natural video.

From the logged data we derived the set of measures presented in Tables 7.1 and 7.2. The measures are shown for each search topic separately. The shots retrieved measure represents the total number of shots saved by all users for each search topic irrespective of the search interface used. The cumulative column gives the sum of shots saved by all users including the duplication of shots when saved by different users. The distinct value is obtained from the above cumulative number by removing duplicate shots. This value shows how many relevant shots were found for each topic. The distinct retrieved shots are then divided into shots saved with the object-based and with the image-based interface respectively. The unique retrieved value gives the number of distinct shots retrieved with only one of the search interfaces.

Table 7.2 shows the average values obtained during the 4 executions (by 4 users) of a search topic and each interface. The *average retrieved* shots gives the mean number of distinct shots saved. The *average query length* shows how many images/objects have been used for each query, and *average iterations* presents the number of iteration runs for each search task. The last distinct column of this table measures the *average utilisation of object functionality*

7. Retrieval experiments

in terms of average number of images for which object features and/or global image features have been used within the object-based search interface.

7.3.6 Results interpretation

As shown by the *shots retrieved* values in Table 7.1 from the comparison between the *cumulative* and *distinct* values the sets of shots saved by different users largely overlap, which means that most users were able to find the same relevant shots although they may have used a different combination of query images or features. However, during the experiments we observed that most users tended to initiate the search tasks from the same Google retrieved images, usually those found on the first page. Thus it is likely that most users have followed closely related search paths.

The number of *distinct retrieved* shots provides a measure of recall bound by the number of saved shots. By comparing the number of distinct shots retrieved with each search interface it can be observed that users found more relevant shots with the object-based interface. However that is not true for all search topics. For a few search topics such as *fish market*, *bridge*, *nightclub life* and *historic building* searching on the image-based interface seemed to provide better results. These topics seem to be more suited to global image feature searching and although such features were also available on the object-based interface, users made only limited use of them, focusing mostly on object features. Additionally it is clear that except for the *bridge* topic, for the other three topics it is relatively difficult to define what images/objects will provide a good initial query. The object-based retrieval seems to provide not only better recall, but also helps with locating shots that are not found by using image-only searching.

The average number of retrieved shots shows that object features provide better searching power than global features alone. The *average query length* and *average iterations* values are somehow correlated since performing an object-based search involves some time dedicated to segmenting objects which invariably reduces the time allocated to actually searching and therefore decreases the query length and the number of search iterations a user will be able to perform. The results show that although using shorter queries and less iterations, object-based search compensates through the additional discerning capacity provided by the object's features. The *average utilisation of object functionality* shows that searchers have largely employed object-based features when available.

7.3.7 Discussion

The experimental results indicate that object-based search consistently outperforms image-based search. However, there are search topics that are not well suited to object-based retrieval. Generally, the content of an image or a video can be described by one or more objects and the relationships between them. However, some content does not feature main objects, but rather contains multiple small objects, shape-less or variable shape materials such as: sand, grass, small seeds and beans, etc. This type of content does not benefit from object retrieval, the appropriate content for object-retrieval being where the object features are largely preserved across object instances.

While object extraction is not as much an inconvenience in terms of interaction effort required from users, defining a query in terms of objects is not always straightforward when the search target is not a specific object. For narrow topics (e.g. “find shots of red cars”) the query is an instance of that object class, however for broad topics (e.g. “find shots of fruits and/or vegetables”) the query is more complex to specify since the specific instance selected as query example (e.g. apple) may be missing in the content.

A drawback of searching by objects is that some of the search time is taken by extracting the query objects from images. However, objects seem to provide faster query convergence in most cases. The worst case scenario is when objects are used but the search topic is not suited to object search. Whether a search topic is suited to searching by objects may not always be obvious from the beginning.

Object search was largely employed by users for most of the query topics. This shows that when objects are available and suitable for the query topic, searchers will make use of them either alone or in combination with global image features. Combination of both objects and global features provides an enhanced approach to query formulation.

7.4 Experimental setting for further investigation of object-based retrieval

There are many ways to further investigate the utilisation of object-based functionalities in content-based retrieval starting from the approach proposed in this work. Some avenues

7. Retrieval experiments

which can be explored are: how users interact with objects, whether object-based retrieval improves the overall performance compared to global image retrieval on various type of content, the impact various object descriptors have on retrieval performance, or whether particular retrieval tasks are more suited to object-based or image-based retrieval. This section describes an experimental setup that could be used to compare the performance of objects against image in automatic retrieval tasks.

In order to run this comparison a test corpus containing various classes of image and objects should be first collected, preferably a commonly used corpus such as the TRECVID⁴ search collection. The objects present in the image collection would then be segmented. Both the images and the segmented objects are then indexed according to the colour, shape and texture features. The next step would be to develop of a set of search topics which match the content of the data without bias towards one of the retrieval modalities, object or image. TRECVID search topics would be a suitable set of topics when using the TRECVID collection since it is carefully chosen to accommodate the content of the data, and independently verified ground-truth is also available.

The experimentation can be divided in three search scenarios: object search, image search, and combined search. In the object search scenario the retrieval is performed only based on object descriptors, in the image search scenario it is only based on image descriptors, while in the combined search both object and image descriptors are used. For the actual search runs a ground-truth object or image for the searched topic is introduced as an initial query input to the system. The search phase for each scenario is fully automatic and this allows large numbers of runs since it is not constrained on the availability of human users. In order to maximise the quantity of experimental data collected the search can be run for all possible combinations of features and input queries for all search scenarios and all topics. In this way every ground-truth item for a topic will serve as query for that search topic. Precision and recall values are computed for each run and averaged over each topic on each search scenario. This allows the comparison of retrieval performance for each topic individually on all three scenarios.

The influence of relevance feedback can be simulated by making use of pseudo-feedback where the top retrieval documents are assumed to be relevant, such that each search run will be performed for a number of iterations. At each iteration the top ranking N retrieved

⁴<http://www-nlpir.nist.gov/projects/trecvid/>

documents assumed to be relevant can be used to update the query. However, this does not mimic adequately the way human users provide feedback since the top ranking items may not in fact be relevant documents. An improvement to pseudo-feedback can be done by updating the query with the first N retrieved documents that are also relevant documents in the topic's ground-truth.

7.5 Summary

In this chapter we described an investigation into the usability of semantic objects in content based retrieval. The first part of our investigation attempted an evaluation of our framework by using a single expert user in a set of retrieval tasks. The values measured in this experiment were precision and recall. The data collection used is relatively small and thus the results cannot be directly compared to other systems, however the investigation provided a quantitative measure on the potential use of objects in image and video retrieval.

The second part of our investigations was an empirical evaluation of object-based video search functionality in an interactive search experiment. This was done in an attempt to isolate the impact of object-based search taking as an experimental collection the BBC rushes video corpus where text from automatic speech recognition (ASR), from video OCR, and from closed captions is not available. Sixteen users each completed 12 different searches, each in a controlled and measured environment with a 5 minute time limit to complete each search.

The analysis of logged data corroborated observations of user's behaviour during the search show that object-based searching consistently outperforms the image-based search. This result goes some way towards validating the approach of allowing users to select objects as a basis for searching video archives when the search dictates it as appropriate, though the technology to do this, is still under development for larger scale video collections.

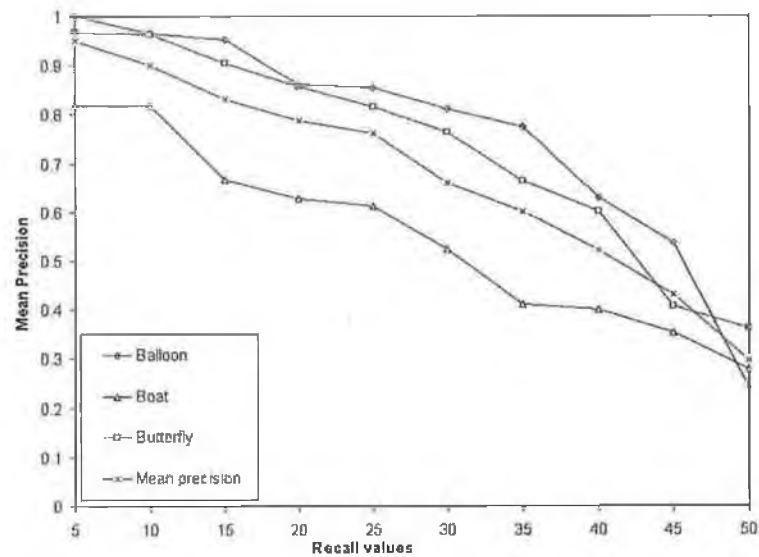


Figure 7.2.A: Mean precision vs recall curves for 12 object classes - subgraph A

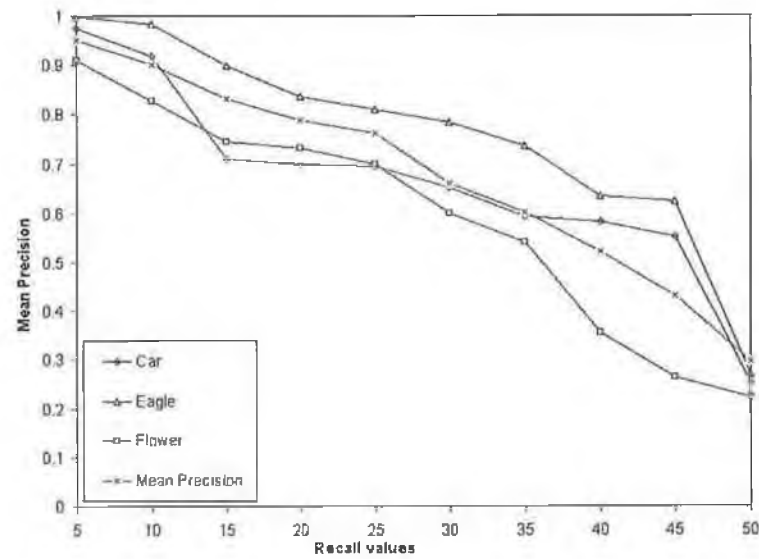


Figure 7.2.B: Mean precision vs recall curves for 12 object classes - subgraph B

7. Retrieval experiments

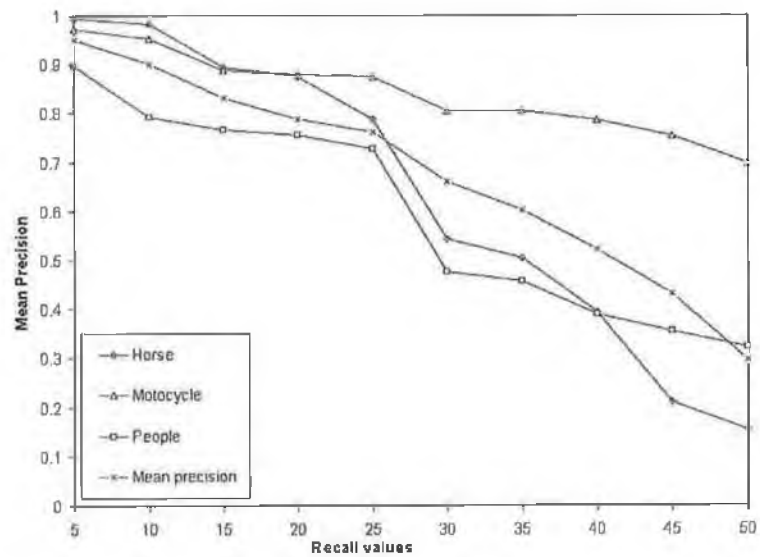


Figure 7.2.C: Mean precision vs recall curves for 12 object classes - subgraph C

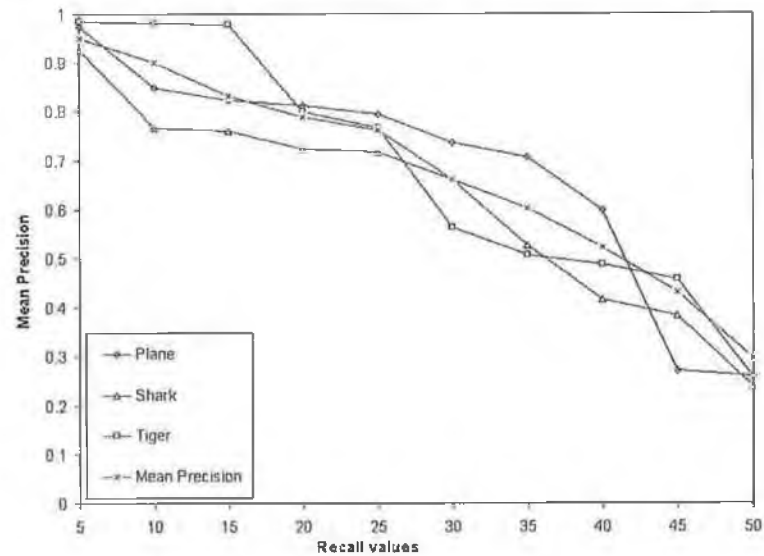


Figure 7.2.D: Mean precision vs recall curves for 12 object classes - subgraph D

7. Retrieval experiments



Figure 7.3: Graphical user interface with query branching enabled

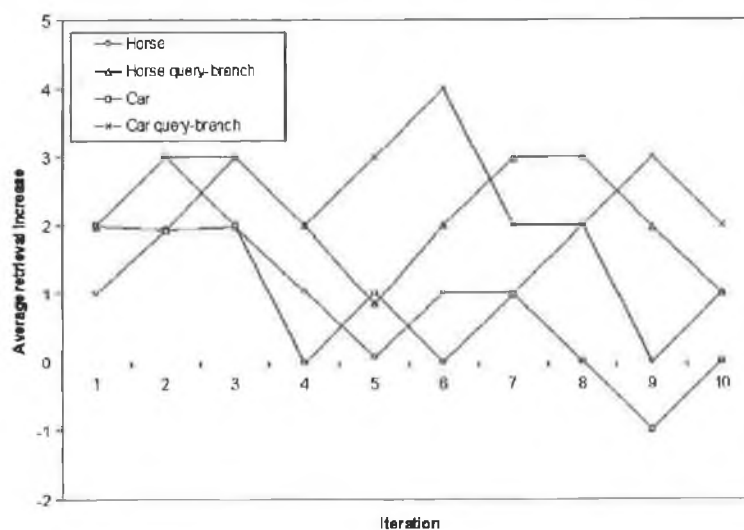


Figure 7.4: Differential modification in relevant documents retrieved per iteration

7. Retrieval experiments

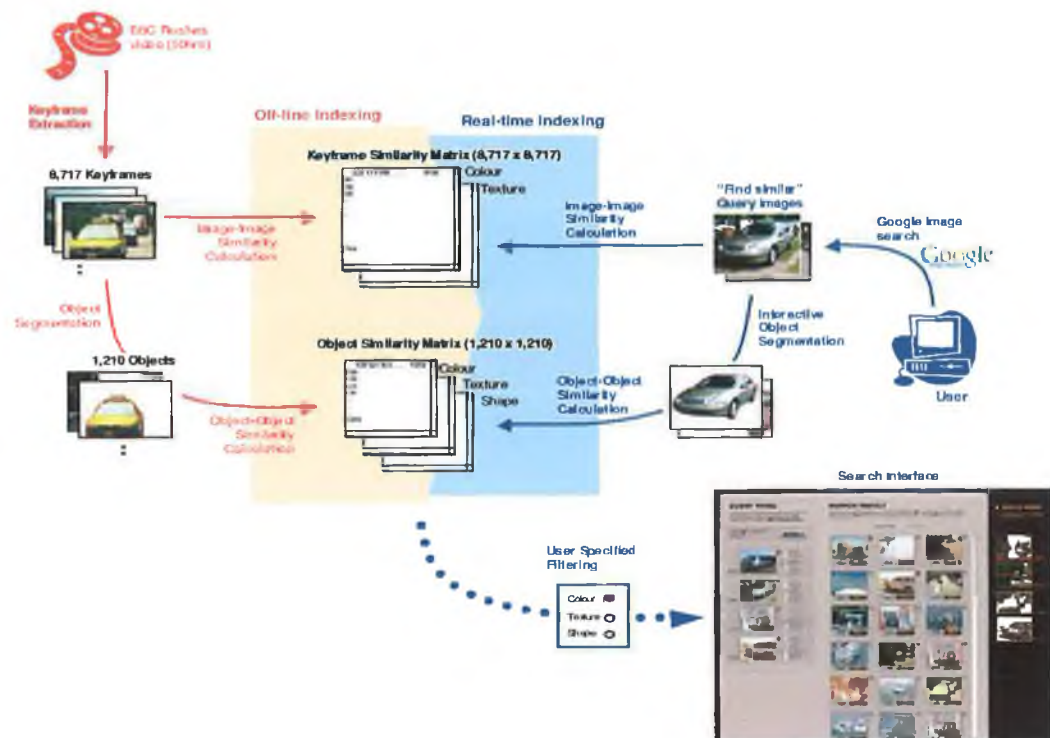


Figure 7.5: System architecture overview



Figure 7.6: Examples of images used as initial query for the topics: "people walking on the beach" and "nightclub life"

7. Retrieval experiments

Topic no #	Topics	Shots retrieved		Distinct retrieved		Unique retrieved	
		Cumulative	Distinct	Object interface	Image interface	Object interface	Image interface
1	helicopter	32	7	7	5	2	0
2	people walking on the beach	72	18	16	12	2	2
3	fish market	20	8	4	6	1	3
4	boats at sea or in harbour	124	29	27	19	11	2
5	fresh vegetables or fruits	28	9	5	7	1	3
6	bridge	16	5	4	3	2	1
7	farm animals	56	14	13	8	5	1
8	palm trees	108	23	21	15	10	2
9	people in urban settings	140	29	27	19	9	2
10	nightclub life	44	15	8	12	1	5
11	camels	44	12	11	7	5	1
12	people in traditional dress	52	16	13	10	6	2
13	flying birds	52	11	10	8	3	1
14	cars in urban settings	96	21	21	13	5	1
15	people in the pool or sea	68	17	14	11	5	1
16	historic buildings	60	14	11	12	2	3
17	people sunbathing	108	22	19	16	4	2
18	skyscrapers	40	9	8	7	2	1
19	people inside a restaurant/bar	24	8	6	5	2	2
20	pigeons in a plaza	40	9	8	6	2	1
21	shoes in a shop window	64	18	17	8	12	2
22	people wind-surfing	40	11	10	6	3	1
23	elephant	28	6	6	4	2	0
24	plane in flight	56	12	11	9	3	1
Average		58.8	14.3	12.4	9.5	4.2	1.7
		F-test		0.0581		0.0013	
		p-value		0.8106		0.9714	

Table 7.1: Size-bounded recall by search topic

Topic no #	Average retrieved		Average query length		Average iterations		Average utilisation of object functionality	
	Object interface	Image interface	Object interface	Image interface	Object interface	Image interface	Object features	Image features
1	5.1	3.1	1.5	2.4	3.7	6.8	2.3	0.2
2	10.9	6.7	2.2	2.8	6.1	9.1	2.1	1.2
3	1.6	3.1	3.2	2.3	5.9	6.7	2.9	2.3
4	20.0	10.7	2.1	3.4	7.1	8.9	2.4	1.1
5	3.1	4.0	2.7	2.9	2.9	5.8	3.1	2.4
6	3.2	0.9	2.1	2.3	5.9	9.2	2.4	1.3
7	9.8	4.0	1.6	2.9	4.3	5.8	2.0	1.0
8	18.2	9.0	2.3	3.3	5.9	8.2	2.3	1.2
9	23.0	12.0	2.5	4.3	8.2	8.9	3.1	1.4
10	4.9	6.3	2.3	3.4	3.8	6.0	2.4	1.9
11	7.7	3.2	2.1	2.4	4.7	8.0	1.4	1.3
12	8.2	4.9	3.2	2.2	6.9	8.6	2.8	1.1
13	7.7	5.3	3.0	3.2	6.5	8.2	2.2	1.7
14	17.6	6.0	1.6	2.3	6.8	9.1	2.1	1.3
15	11.1	5.9	2.7	1.8	8.2	8.7	3.1	1.2
16	7.3	7.9	2.0	2.4	6.1	8.2	2.4	2.2
17	15.7	11.1	3.1	3.2	6.8	9.3	3.1	0.9
18	6.0	4.0	1.7	2.3	3.6	5.7	2.0	1.2
19	3.8	2.0	2.6	3.4	7.9	9.4	2.9	1.4
20	7.2	2.8	2.3	2.3	4.7	8.8	2.4	1.7
21	14.3	2.1	2.1	2.7	5.9	9.2	2.1	1.1
22	8.4	2.0	3.0	3.8	4.0	6.9	3.3	1.4
23	4.8	2.0	1.5	2.4	5.7	9.2	2.3	0.3
24	9.3	4.9	0.7	1.8	4.3	6.9	1.4	1.1
Average	9.5	5.2	2.2	2.8	5.6	8.1	2.4	1.3
F-test	0.0039		0.9877		0.4380		0.9047	
p-value	0.9505		0.3254		0.5113		0.3464	

Table 7.2: Average size-bounded recall by search topic

Chapter 8

Conclusions and future work

The gap between the user's perception of what information is contained in data stored in image and video, and the capabilities of current data analysis technology for that media cannot be bridged by low-level features alone. Since users search for semantically meaningful events, queries exclusively based on low-level features alone have very limited success. There is an increasing demand to incorporate semantics in the data descriptors in order to enable image and video retrieval by concepts which are meaningful to users.

8.1 A Brief Review

Chapter two provided an overview on content based information retrieval. In this context search, browsing and user interaction for visual retrieval were introduced. The role of low-level features as the base layer in retrieval was then summarised. Semantic objects are retrieval units that have the potential of bridging the gap between low-level features and semantic concepts. Attempts at using objects for retrieval have been reported for a decade, but only with recent technology has object-based functionality been enabled for video. However, detection of all possible object types is generally considered a computationally infeasible problem. Here are briefly mentioned some of the object-based retrieval work reported in the research literature. This set the context for the investigation reported in the thesis. The chapter also presented aspects related to the evaluation of retrieval systems and a short overview of few major content based image and video retrieval systems.

Chapter three reviewed some low-level visual content descriptors commonly used in image and video retrieval. It covered descriptors associated with colour, shape and texture features. Although there are many other features or combinations of features which can be extracted from images and video, such as motion and audio, these were not described here since no features from these categories are used in the work reported in this dissertation. Although not discussed in detail in the chapter, every descriptor has its advantages and as well as shortcomings for specific application domains. The benefit of using a particular representation has to be determined for each scenario regarding on the type of discriminative power, robustness, invariance, storage capacity and computation complexity demanded. Finally we moved from image descriptors into the video domain by introducing the process of shot boundary segmentation. Shot segmentation is the first step in the video analysis chain by breaking the content into segments of related frames such that each segment can be analysed independently.

Chapter four covered the topic of image segmentation with a main focus on methods related to segmentation for content-based retrieval. Image segmentation is a fundamental step in object identification and recognition. The complexity of segmentation varies in practical applications, and even the most advanced methods require some form of user input in order to adapt to a large range of segmentation scenarios. In the context of image retrieval, segmentation provides a way to focus the search on the most relevant areas of the image, the semantic objects present in the scene. This chapter introduced the main techniques developed in this research area covering methods such as histogram thresholding, clustering, region-growing, split-merge algorithms, contour and motion based segmentation. Aspects related to interactive segmentation tools were also presented in the chapter.

Chapter five gave an overview of relevance feedback. Relevance feedback is an effective method for iteratively improving a query by collecting a user's feedback during a search session. In addition to improving the query, and as a consequence increasing the retrieval performance, the feedback mechanism structures the search process by breaking the search operation into sequences of iterative steps. This sequencing makes the entire retrieval process more controllable, and intuitive to the user. Conceptually, relevance feedback acts as an interface that hides the complexity of query modification from the user. The main challenge in the relevance feedback process is that users provide only limited and sometimes inadequate feedback information as they are influenced by many objective and subjective factors when assessing relevance of a given document.

Chapter six described an approach to using object-based searching functionalities in the retrieval of visual content. A framework for extraction of objects from images and video was developed and presented in this chapter. This framework contains the entire processing chain required to analyse, index and interactively retrieve images and video via object-to-object matching. The elements of the processing are as follows: shot boundary detection, an interactive object segmentation tool and visual feature extraction, a relevance feedback mechanism and a graphical user interface. The video sequence segmentation developed here takes advantage of the distribution of macroblock types in the MPEG compressed domain. The semi-automatic segmentation is designed around a scribble based interaction on pre-segmented keyframes obtained from an automatic region segmentation approach. The relevance feedback mechanism is modeled by a Gaussian mixture model incorporating positive and negative examples. The graphical user interface facilitates interaction with objects and object-based query formulation.

Chapter seven presented an investigation into the utilisation of objects in content-based visual information retrieval. An evaluation of object-based search against standard image-based search has been carried out in an interactive experiment with 24 search topics and 16 users each performing 12 search tasks on 50 hours of rushes video. This experiment attempts to measure the impact of object-based search on a corpus of video where textual annotation is not available. A second experiment explores the retrieval performance of the framework in terms of precision and recall.

8.2 Conclusions

Content-based retrieval of visual information is a complex process as various features associated with image and video data need to be analysed. There are two approaches to managing digital video: using manually inserted annotations and metadata, and automatically processing video by deriving content descriptors. In the years since digital technology has become ubiquitous, a variety of approaches and modalities for automatic content description have emerged although they are still far from achieving their full potential.

Automatic analysis of video is considerably more complicated and less feasible than the analysis of still images, although the present state of the art in the processing and retrieval of visual content is largely confined to automatic scene detection promising new directions

8. Conclusions and future work

have been opened. Object-based retrieval which allows users to manipulate video objects as part of their searching and browsing interaction is one of these directions.

The main aim of this work is to investigate an approach to using objects for video retrieval and a set of experiments was developed in order to explore some issues related to this. The focus of this thesis is on general ad hoc retrieval by a hypothetical professional user performing visual queries on a video and image archive. In this section are discussed the main findings of this research. Experimental results indicate that object-based search overall outperforms the image-based search for the set of topics used here. However, there are search topics that are not well suited to object-based retrieval.

While object extraction is not as much an inconvenience in terms of interaction effort required from users, defining a query in terms of objects is not always straightforward when the search target is not a specific object. For narrow topics (e.g. “find shots of red cars”) the query is an instance of that object class, however for broad topics (e.g. “find shots of fruits and/or vegetables”) the query is more complex to specify since the specific instance selected as a query example (e.g. apple) may be missing in the content.

A drawback of searching by objects is that some of the search time is taken by having to extract the query objects from images. However, objects seem to provide faster query convergence in most cases. The worst case scenario is when objects are used, but the search topic is not suited for object search. Whether a search topic is suited to searching by objects may not be always obvious from the beginning.

Object search was largely employed by users for most of the query topics in the experiments. This shows that when objects are available and suitable for the query topic searchers will make use of them either alone or in combination with global image features. A combination of both objects and global features provides an enhanced approach to query formulation.

8.3 Future work

The investigation of the proposed framework has provided interesting results and it would be useful to investigate object-based retrieval under various other conditions, for example a change of corpus content with a larger number of users and search topics. An extension to the current work would be an investigation of various other low-level features for objects

8. Conclusions and future work

such as various colour descriptors and shape templates. Adding motion features in the object representation would enable retrieval by motion trajectory.

A comprehensive investigation of various other low-level features would provide better understanding on which features are effective in object-based searching. Also domain-specific applications can be investigated in order to determine which features work best under specific conditions. Surveillance and related applications seem to be the most suitable domain for this approach.

Object extraction is a major drawback in the proposed framework. Future work should focus on fully automated segmentation, perhaps by designing object templates for specific classes, and on simplifying the interaction required in defining objects. Fully automated segmentation could be possible in constrained applications such as surveillance where a small number of object classes are expected to occur in the content.

Another interesting area of research would be a long-term learning approach to relevance judgements. The relevant items collected from different users can be used off-line in deriving global connections among objects. This can help in reducing the number of feedback iterations by using the connected objects as unlabeled but “probably-relevant” data in the query updating process. Also the global connections could serve as links in a retrieval by browsing approach.

References

- [1] A. Smeulders, M. Worring, S. Santini, A. Gupta and R. Jain, "Content based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1349–1380, December 2000.
- [2] D.H. Ballard and C.M Brown, *Computer Vision*. Prentice Hall, New Jersey, USA, 1982.
- [3] R.M. Haralick and L.G Shapiro, *Computer and Robot Vision*. Addison-Wesley, New York, USA, 1993.
- [4] A.F. Smeaton and P. Over, "Benchmarking the effectiveness of information retrieval tasks on digital video," *Proceedings of the 2nd International Conference on Image and Video Retrieval (CIVR 2003)*, Urbana, IL, pp. 19–27, July 2003.
- [5] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele and P. Yanker, "Query by image and video content: The QBIC system," *IEEE Computer*, pp. 23–32, September 1995.
- [6] J.R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jam, and C.F. Shu, "The VIRAGE search engine: An open framework for image management," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases IV*, pp. 77–87, February 1996.
- [7] A. Pentland, R.W. Picard, and S. Sclaroff, "Photobook: Content-based manipulation of image databases," *International Journal of Computer Vision*, pp. 233–254, June 1996.
- [8] J.R. Smith and S.F. Chang, "An image and video search engine for the world-wide web," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases V*, pp. 85–95, February 1997.

- [9] J.R. Smith and S.F. Chang, "Querying by color regions using the VisualSEEK content-based visual query system," *Intelligent Multimedia Information Retrieval*, MIT Press, pp. 159–173, 1997.
- [10] W.Y. Ma and B.S. Manjunath, "Netra: A toolbox for navigating large image databases," *Proceedings of IEEE International Conference on Image Processing (ICIP '97)*, Santa Barbara, CA, pp. 568–571, October 1997.
- [11] T.S. Huang, S. Mehrotra, and K. Ramchandran, "Multimedia analysis and retrieval system (MARS) project," *Proceedings of the 33rd Annual Clinic on Library Application of Data Processing - Digital Image Access and Retrieval*, Urbana, IL, March 1996.
- [12] H. Wactlar, "Informedia - search and sumarization in the video medium," *Proceedings of Imagina 2000 Conference*, Monaco, January-February 2000.
- [13] M. Pickering and S. Rüger, "ANSES: Summarisation of news video," *Proceedings of International Conference on Image and Video Retrieval, (CIVR 2003)*, Urbana-Champaign, IL, pp. 425–434, July 2003.
- [14] J.P. Eakins, K.J. Riley and J.D. Edwards, "Shape feature matching for trademark image retrieval," *Proceedings of the 2nd International Conference on Image and Video Retrieval (CIVR 2003)*, Urbana, IL, pp. 28–38, July 2003.
- [15] A. Bosson, G.C. Cawley, Y. Chan, and R. Harvey, "Non-retrieval: Blocking pornographic images," *Proceedings of the 2nd International Conference on Image and Video Retrieval (CIVR 2002)*, London, UK, pp. 50–60, July 2002.
- [16] Y. Rui, T.S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: A power tool for interactive content-based image retrieval," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, pp. 644–655, September 1998.
- [17] H. Rowley, S. Baluja and K. Kanade, "Human face detection in visual scenes," *Proceedings of Neural Information Processing Systems*, Denver, CO, vol. 8, pp. 875–881, November 1996.
- [18] M.S. Lew and N. Huijsmans, "Information theory and face detection," *Proceedings of International Conference on Pattern Recognition*, Vienna, Austria, pp. 601–605, 1996.

- [19] J.S. Boreczky and L.A. Rowe, "Comparison of video shot boundary detection techniques," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, pp. 170–179, February 1996.
- [20] A. Hanjalic, R.L. Lagendijk, and J. Biemond, "A new method for key frame based video content representation," *Image Databases and Multimedia Search, World Scientific*, pp. 97–107, 1997.
- [21] M. Haas, M.S. Lew, D.P. Huijsmans, "Shot break detection and camera motion classification," *Image Databases and Multimedia Search, World Scientific*, pp. 191–201, 1997.
- [22] R. Lienhart, "Reliable transition detection in videos: A survey and practitioner's guide," *International Journal of Image and Graphics*, vol. 1, pp. 469–486, July 2001.
- [23] N. O'Connor, C. Czirjek, S. Deasy, S. Marlow, N. Murphy, and A.F. Smeaton, "News story segmentation in the Físchlár video indexing system," *Proceedings of International Conference On Image Processing (ICIP'01), Thessaloniki, Greece*, pp. 418–421, October 2001.
- [24] A.F. Smeaton, "Indexing, browsing and searching of digital video," *ARIST Annual Review of Information Science and Technology*, vol. 38, pp. 371–407, September 2004.
- [25] S. Santini and R. Jain, "Integrated browsing and querying for image database," *IEEE Transactions on Multimedia*, vol. 7, pp. 26–39, September 2000.
- [26] J.P. Eakins, "Automatic image content retrieval - are we getting anywhere?," *Proceedings of the 3rd International Conference on Electronic Library and Visual Information Research (ELVIRA3), Milton Keynes, UK*, pp. 123–135, May 1996.
- [27] P.G.B. Enser, "Pictorial information retrieval," *Journal of Documentation*, vol. 51, pp. 126–170, February 1995.
- [28] P.C. Aigrain, H.C. Zhang, and D.C. Petkovic, "Content-based representation and retrieval of visual media - a state-of-the-art review," *Multimedia Tools and Applications*, vol. 3, pp. 179–202, March 1996.

- [29] K. Messer and J. Kittler, "Using feature selection to aid an iconic search through an image database," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97)*, Munich, Germany, pp. 2605–2608, April 1997.
- [30] Y. Rui, T.S. Huang, and S. Chang, "Image retrieval: Current techniques, promising directions and open issues," *Journal of Visual Communication and Image Representation*, pp. 39–62, April 1999.
- [31] D. Heesch, and S. Rüger, "Semantic analysis of NN^k networks," *International Conference on Image and Video Retrieval, Singapore*, pp. 609–618, July 2005.
- [32] D. Heesch, M. Pickering, A. Yavlinsky, and S. Rüger, "Video retrieval within a browsing framework using keyframes," *Proceedings of TRECVID 2003*, 2004.
- [33] J. Rocchio, "Relevance feedback in information retrieval," *The Smart System-Experiments In Automatic Document Processing, Englewood Cliffs, NJ: Prentice Hall*, pp. 313–323, 1971.
- [34] H. Lee and A.F. Smeaton, "Designing the user-interface for the Físchlár digital video library," *Journal of Digital Information, Special Issue on Interactivity in Digital Libraries*, vol. 2, May 2002.
- [35] D. Heesch and S. Rüger, "Three interfaces for content-based access to image collections," *Proceedings of International Conference on Image and Video Retrieval (CIVR'04)*, Dublin, Ireland, pp. 491–499, July 2004.
- [36] Y.A. Aslandogan and C.T. Yu, "Techniques and systems for image and video retrieval," *IEEE Transactions on Knowledge and Data Engineering*, vol. 11, pp. 155–164, February 1999.
- [37] M. Swain and D. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, pp. 11–32, 1991.
- [38] M. Stricker and M. Orengo, "Similarity of color images," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases III, San Jose, CA*, vol. 2185, pp. 381–392, February 1995.

- [39] M. Stricker and A. Dimai, "Color indexing with weak spatial constraint," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases IV, San Jose, CA*, pp. 29–40, February 1996.
- [40] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: Segmentation using expectation-maximization and its application to image querying," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1026–1038, August 2002.
- [41] G. Pass and R. Zabith, "Histogram refinement for content-based image retrieval," *IEEE Workshop on Applications of Computer Vision*, pp. 96–102, December 1996.
- [42] J. Huang, S. Kumar, M. Metra, W. Zhu, and R. Zabith, "Image indexing using color correlogram," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'97), Puerto Rico*, pp. 762–768, July 1997.
- [43] T. Gevers and A. Smeulders, "Pictoseek: Combining color and shape invariant features for image retrieval," *IEEE Transactions on Image Processing*, vol. 9, pp. 102–119, January 2000.
- [44] C.T. Zahn and R.Z. Roskies, "Fourier descriptors for plane close curves," *IEEE Transactions on Computers*, vol. C-21, pp. 269–281, 1972.
- [45] E. Persoon and K. Fu, "Shape discrimination using Fourier descriptors," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 7, pp. 170–179, 1977.
- [46] Y. Rui, T.S. Huang, S. Mehrotra, and M. Ortega, "Automatic matching tool selection via relevance feedback in MARS," *Proceedings of the 2nd International Conference on Visual Information Systems, San Diego, CA*, pp. 109–116, December 1997.
- [47] M.K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions in Information Theory*, vol. 8, pp. 179–187, 1962.
- [48] L. Yang and F. Algrejtsen, "Fast computation of invariant geometric moments: A new method giving correct results," *Proceedings of the 12th IAPR International Conference on Pattern Recognition, Jerusalem, Israel*, pp. 201–204, October 1994.
- [49] R. Barber W. Niblack, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin, "The QBIC project: Querying images by content using colour,

- texture, and shape," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases, San Jose, CA*, pp. 173–187, January 1993.
- [50] T. Gevers and A. Smeulders, "Content-based image retrieval by viewpoint-invariant image indexing," *Journal of Image and Vision Computing*, vol. 17, pp. 475–488, July 1999.
- [51] E.M. Arkin, L.P. Chew, D.P. Huttenlocher, K. Kedem, and J.S. Mitchell, "An efficiently computable metric for comparing polygonal shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, pp. 209–226, 1991.
- [52] D. Tegolo, "Shape analysis for image retrieval," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases II, San Jose, CA*, pp. 59–69, February 1994.
- [53] F. Mokhtarian and S. Abbasi, "Shape similarity retrieval under affine transforms," *Pattern Recognition*, vol. 35, pp. 31–41, 2002.
- [54] D. Zhang and G. Lu, "Content-based shape retrieval using different shape descriptors: A comparative study," *Proceedings of International Conference on Multimedia and Expo (ICME '01), Tokyo, Japan*, pp. 1139–1142, August 2001.
- [55] H. Tamura, S. Mori and T. Yamawaki, "Texture features corresponding to visual perception," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 8, pp. 460–473, June 1978.
- [56] F. Liu, and R.W. Picard, "Periodicity, directionality, and randomness: Wold features for image modeling and retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 722–733, 1996.
- [57] T. Chang and C.C.J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Transactions on Image Processing*, vol. 2, pp. 429–441, October 1993.
- [58] A. Laine and J. Fan, "Texture classification by wavelet packet signatures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 1186–1191, November 1993.

- [59] B.S. Manjunath and W.Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 837–842, August 1996.
- [60] A. Kankanhalli, H.J. Zhang and C.Y. Low, "Using texture for image retrieval," *Proceedings of 3rd International Conference on Automation, Robotics and Computer Vision, Singapore*, pp. 935–939, November 1994.
- [61] W.Y. Ma and B.S. Manjunath, "Image indexing using a texture dictionary," *Proceedings of SPIE Image Storage and Archiving System*, vol. 2606, pp. 288–298, October 1995.
- [62] W.Y. Ma and B.S. Manjunath, "A comparison of wavelet features for texture annotation," *Proceedings of IEEE International Conference on Image Processing, Washington D.C.*, vol. 2, pp. 256–259, October 1995.
- [63] P. Howarth and S. Rüger, "Evaluation of texture features for content-based image retrieval," *Proceedings of International Conference on Image and Video Retrieval (CIVR 2004), Dublin, Ireland*, pp. 326–334, July 2004.
- [64] A.F. Smeaton and P. Browne, "A usage study of retrieval modalities for video shot retrieval," *Information Processing and Management*, vol. 42, pp. 1330–1344, September 2006.
- [65] E. Oomoto and K. Tanaka, "OVID: design and implementation of a video-object database system," *IEEE Transactions on Knowledge and Data Engineering*, pp. 629–643, April 1993.
- [66] M. Weber, M. Welling, and P. Perona, "Towards automatic discovery of object categories," *Proceedings of IEEE Computer Vision and Pattern Recognition Conference (CVPR'00), Hilton Head Island, SC*, pp. 101–109, June 2000.
- [67] P. Hong, R. Wang, and T. Huang, "Learning patterns from images by combining soft decisions and hard decisions," *Proceedings of IEEE Computer Vision and Pattern Recognition Conference (CVPR'00), Hilton Head Island, SC*, pp. 78–83, June 2000.
- [68] L. Hohl, F. Souvannavong, B. Merialdo, and B. Huet, "Enhancing latent semantic analysis video object retrieval with structural information," *Proceedings of IEEE International Conference on Image Processing (ICIP'04), Singapore*, pp. 1609–1612, October 2004.

- [69] B. Erol and F. Kossentini, "Shape-based retrieval of video objects," *IEEE Transactions on Multimedia*, vol. 7, pp. 179–182, February 2005.
- [70] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," *Proceedings of IEEE International Conference On Computer Vision (ICCV'03)*, Nice, France, pp. 1470–1477, October 2003.
- [71] J. Sivic, F. Shaffalitzky, and A. Zisserman, "Efficient object retrieval from videos," *Proceedings of European Signal Processing Conference (EUSIPCO04)*, Vienna, Austria, pp. 159–165, September 2004.
- [72] C.B. Liu and N. Ahuja, "Motion based retrieval of dynamic objects in videos," *Proceedings of ACM Multimedia*, New York, NY, pp. 288–291, October 2004.
- [73] M. Smith and A. Khotanzad, "An object-based approach for digital video retrieval," *Proceedings of International Conference On Information Technology: Coding And Computing (ITCC 2004)*, Las Vegas, NV, pp. 456–459, April 2004.
- [74] S. Marchand-Maillet, "Construction of a formal multimedia benchmark," *Proceedings of European Signal Processing Conference (EUSIPCO2002)*, Toulouse, France, pp. 455–458, September 2002.
- [75] P. Over, C.H.C. Leung, H.H.S. Ip, and M. Grubinger, "Multimedia retrieval benchmarks," *IEEE Multimedia*, vol. 11, pp. 80–84, April 2004.
- [76] D. Martin, C. Fowlkes, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," *Proceedings of International Conference on Computer Vision (ICCV'01)*, Vancouver, Canada, pp. 416–425, July 2001.
- [77] R.C. Veltkamp and M. Tanase, "Content-based image retrieval systems: A survey," *Technical Report, Utrecht University, Department of Computer Science*, March 2001.
- [78] Y. Rui, T.S. Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in MARS," *Proceedings of IEEE International Conference on Image Processing (ICIP'97)*, Santa Barbara, CA, pp. 815–818, October 1997.

- [79] I. Kompatsiaris, E. Triantafillou, and M. G. Strintzis, "Region-based colour image indexing and retrieval," *Proceedings of International Conference on Image Processing (ICIP'01), Thessaloniki, Greece*, pp. 658–661, October 2001.
- [80] E. Izquierdo, J. Casas, R. Leonardi, P. Migliorati, N. O'Connor, I. Kompatsiaris, and M. Strintzis, "Advanced content-based semantic scene analysis and information retrieval: The SCHEMA project," *Proceedings of the 4th Workshop on Image Analysis for Multimedia Interactive Service (WIAMIS 2003), London, U.K.*, April 2003.
- [81] S. Sclaroff, L. Taycher, and M. La Cascia, "Image rover: A content-based image browser for the world wide web," *Proceedings of IEEE Workshop on Content-based Access of Image and Video Libraries, Puerto Rico*, pp. 140–152, June 1997.
- [82] M. Swain, C. Frankel, and V. Athitsos, "WebSeer: An image search engine for the world wide web," *Technical report tr-96-14, University of Chicago Department of Computer Science*, July 1996.
- [83] J. Laaksonen, M. Koskela, S. Laakso, and E. Oja, "Self-organizing maps as a relevance feedback technique in content-based image retrieval," *Pattern Analysis and Applications*, pp. 140–152, June 2001.
- [84] M. Koskela, "Interactive image retrieval using self-organizing maps," *PhD thesis, Helsinki University of Technology, Espoo, Finland*, November 2003.
- [85] N. O'Connor, S. Marlow, N. Murphy, A. Smeaton, P. Browne, S. Deasy, H. Lee, and K. McDonald, "Físchlár: An on-line system for indexing and browsing of broadcast television content," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '01), Salt Lake City, UT*, May 2001.
- [86] W. Niblack, S. Yue, R. Kraft, A. Amir and N. Sundaresan, "User interface design for keyframe-based browsing of digital video," *Proceedings of IEEE International Conference on Multimedia and Expo, New York*, July 2000.
- [87] J. Huang, S. Kumar, M. Metra, W. Zhu, and R. Zabith, "Spatial color indexing and applications," *International Journal of Computer Vision*, vol. 35, pp. 245–268, March 1999.
- [88] A. K. Jain, *Fundamental of Digital Image Processing*. Englewood Cliffs, Prentice Hall, 1989.

References

- [89] E. Mathias, "Comparing the influence of color spaces and metrics in content-based image retrieval," *Proceedings of International Symposium on Computer Graphics, Image Processing, and Vision*, pp. 371–378, 1998.
- [90] G. Pass and R. Zabith, "Comparing images using joint histograms," *ACM Journal of Multimedia Systems*, vol. 7, pp. 234–240, 1999.
- [91] H. Zhang, Y. Gong, C. Low and S. Smoliar, "Image retrieval based on color features: An evaluation study," *Proceedings of SPIE Digital Image Storage and Archiving Systems*, vol. 2606, pp. 212–220, November 1995.
- [92] C.S McCamy, H. Marcus, and J.G. Davidson, "A colour-rendition chart," *Journal of Applied Photographic Engineering*, vol. 2, pp. 95–99, 1976.
- [93] M. Miyahara and Y. Yoshida, "Mathematical transform of (R,G,B) color data to munsell (H,V,C) color data," *Proceedings of SPIE Visual Communications and Image Processing*, vol. 1001, pp. 650–657, 1988.
- [94] J. Wang, W.J. Yang, and R. Acharya, "Colour clustering techniques for colour-content-based image retrieval from image databases," *Proceedings of IEEE Conference on Multimedia Computing and Systems*, 1997.
- [95] D. LeGall, J.L. Mitchell, W.B. Pennbaker, C.E. Fogg, *MPEG video compression standard*. Chapman and Hall, New York, 1996.
- [96] J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes, *Computer graphics: principles and practice*. Reading, Mass, Addison-Wesley, 1990.
- [97] Y. Gong, H.J. Zhang, and T.C. Chua, "An image database system with content capturing and fast image indexing abilities," *Proceedings of IEEE International Conference on Multimedia Computing and Systems, Boston, MA*, pp. 121–130, May 1994.
- [98] G. Pass, R. Zabith, and J. Miller, "Comparing images using color coherence vectors," *Proceedings of ACM Multimedia, Boston, MA*, pp. 65–73, November 1996.
- [99] G.D. Finlayson, "Color in perspective," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, pp. 1034–1038, October 1996.

References

- [100] J.E. Gary and R. Mehrotran, "Shape similarity-based retrieval in image database systems," *Proceedings of SPIE Image Storage and Retrieval Systems*, vol. 1662, pp. 2–8, 1992.
- [101] W.I. Grosky and R. Mehrotra, "Index based object recognition in pictorial data management," *Journal of Computer Vision, Graphics, and Image Processing*, vol. 52, pp. 416–436, 1990.
- [102] H.V. Jagadish, "A retrieval technique for similar shapes," *Proceedings of International Conference on Management of Data (SIGMOID '91), Denver, CO*, pp. 208–217, May 1991.
- [103] R.C. Veltkamp and M. Hagedoorn, "State-of-the-art in shape matching," *Technical Report, Utrecht University, Department of Computer Science*, September 1999.
- [104] S. Sclaroff and A. Pentland, "Modal matching for correspondence and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 545–561, 1995.
- [105] H. Kauppinen, T. Seppnen, and M. Pietikinen, "An experimental comparison of autoregressive and Fourier-based descriptors in 2D shape classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 201–207, 1995.
- [106] K. Arbter, W.E. Snyder, H. Burkhardt, and G. Hirzinger, "Application of affine-invariant Fourier descriptors to recognition of 3D objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 640–647, 1990.
- [107] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. Academic Press, 1999.
- [108] T. Ojala, M. Pietikainen, and D. Harwood, "A comparative study of texture measures with classification based feature distributions," *Pattern Recognition*, vol. 29, pp. 51–59, January 1996.
- [109] H. Voorhees and T. Poggio, "Detecting textons and texture boundaries in natural images," *Proceedings of International Conference on Computer Vision (ICCV'87), London, UK*, pp. 250–258, June 1987.
- [110] H. Tamura and N. Yokoya, "Image database systems: A survey," *Pattern Recognition*, vol. 17, pp. 29–43, 1984.

- [111] J. Mao and A. K. Jain, "Texture classification and segmentation using multiresolution simultaneous autoregressive models," *Pattern Recognition*, vol. 25, pp. 173–188, February 1992.
- [112] A.K. Jain and F. Farroknia, "Unsupervised texture segmentation using gabor filters," *Pattern Recognition*, vol. 24, pp. 1167–1186, 1991.
- [113] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Transactions on Information Theory*, vol. 36, pp. 961–1005, September 1990.
- [114] S.G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, July 1989.
- [115] B. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7: Multimedia Content Description Standard*. Wiley, New York, USA, 2001.
- [116] T. Ojala, M. Aittola and E. Matinmikko, "Empirical evaluation of MPEG-7 XM color descriptors in content-based retrieval of semantic image categories," *Proceedings of the 16th International Conference on Pattern Recognition, Quebec, Canada*, pp. 1021–1024, 2002.
- [117] S. Santini and R. Jain, "Similarity measures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, pp. 871–883, September 1999.
- [118] N. Vasconcelos, "On the efficient evaluation of probabilistic similarity functions for image retrieval," *IEEE Transactions on Information Theory*, vol. 50, pp. 1482–1496, July 2004.
- [119] Y. Rubner, C. Tomasi and L.J. Guibas, "The earth movers distance as a metric for image retrieval," *International Journal on Computer Vision*, vol. 40, pp. 99–121, November 2000.
- [120] J. Puzicha, Y. Rubner, C. Tomasi, and J. Buhmann, "Empirical evaluation of dissimilarity measures for color and texture," *Proceedings of International Conference on Computer Vision (ICCV-1999), Corfu, Greece*, pp. 1165–1173, September 1999.

- [121] C. Aggarwal, A. Hinneburg, and D. Keim, "On the surprising behavior of distance metrics in high dimensional space," *Lecture Notes in Computer Science*, Springer, vol. 1973, pp. 420–434, 2001.
- [122] P. Howarth and S. Rüger, "Fractional distance measures for content-based image retrieval," *Proceedings of European Conference on IR Research (ECIR 2005)*, Santiago de Compostela, Spain, pp. 447–456, March 1999.
- [123] S. Antani, R. Kasturiand, and R. Jain, "A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video," *Pattern Recognition*, vol. 35, pp. 945–965, 2002.
- [124] S.B. Jun, K. Yoon, and H. Lee, "Dissolve transition detection algorithm using spatio-temporal distribution of MPEG macro-block types," *Proceedings of ACM Multimedia*, Marina del Rey, CA, pp. 391–394, October 2000.
- [125] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," *Proceedings of IFIP TC2/WG 2.6 Second Working Conference on Visual Database Systems*, Budapest, Hungary, pp. 113–127, September 1992.
- [126] R. Zabih, J. Miller and K. Mai, "A feature-based algorithm for detecting and classifying scene breaks," *Proceedings of ACM Multimedia*, Boston, MA, pp. 189–200, November 1995.
- [127] H.J. Zhang, A. Kankanhalli, and S.W. Smoliar, "Automatic partition of full-motion video," *Multimedia Systems*, vol. 1, pp. 10–28, January 1993.
- [128] F. Arman, R. DEMPommier, A. Hsu, and M.Y. Chiu, "Content-based browsing of video sequences," *Proceedings of ACM Multimedia*, San Francisco, CA, pp. 97–103, October 1994.
- [129] D. LeGall, J.L. Mitchell, W.B. Pennbaker, and C.E. Fogg, *MPEG video compression standard*. Chapman-Hall, New York, 1996.
- [130] V. Kobla, D.S. Doermann, K.I. Lin, and C. Faloutsos, "Compressed domain video indexing techniques using DCT and motion vector information in MPEG video," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, San Diego, CA, pp. 200–211, February 1997.

- [131] S.W. Lee, Y.M. Kim, and S.W. Choi, "Fast scene change detection using direct feature extraction from MPEG compressed videos," *IEEE Transactions on Multimedia*, vol. 2, pp. 240–254, December 2000.
- [132] B. Yeo and B. Liu, "Rapid scene analysis on compressed video," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 5, pp. 533–544, December 1995.
- [133] U. Gargi and R. Kasturi, "An evaluation of color histogram-based methods in video indexing," *Proceedings of International Workshop on Image Databases and Multimedia Search, Amsterdam, The Netherlands*, pp. 75–82, August 1996.
- [134] C. O'Toole, A.F. Smeaton, N. Murphy, and S. Marlow, "Evaluation of automatic shot boundary detection on a large video test suite," *Proceedings of Conference on Challenge of Information Retrieval, Newcastle, UK*, February 1999.
- [135] P. Chiu, A. Girgensohn, W. Polak, E. Rieffel, and L. Wilcox, "A genetic algorithm for video segmentation and summarization," *Proceedings of IEEE International Conference on Multimedia and Expo, New York, NY*, pp. 1329–1332, July 2000.
- [136] A. Dailianas, R.B. Allen, and P. England, "Comparison of automatic video segmentation algorithms," *Proceedings of SPIE Conference on Integration Issues in Large Commercial Media Delivery Systems, Philadelphia, PA*, pp. 2–16, October 1995.
- [137] W. Zhao, J. Wang, D. Bhat, K. Sakiewicz, N. Nandhakumar, and W. Chang, "Improving color based video shot detection," *Proceedings of IEEE International Conference on Multimedia Computing and Systems, Florence, Italy*, pp. 752–756, June 1999.
- [138] S. Kim and R.H. Park, "A novel approach to scene change detection using a cross entropy," *Proceedings of IEEE International Conference on Image Processing (ICIP'00), Vancouver, Canada*, pp. 937–940, September 2000.
- [139] W. Ren, M. Sharma, and S. Singh, "Comparison of automatic video segmentation algorithms," *Proceedings of International Conference on Information, Communication, and Signal Processing, Singapore*, pp. 2–16, October 2001.
- [140] R. Kasturi and R.C. Jain, *Dynamic vision. Computer Vision : Principles - IEEE Computer Society Press, Washington*, 1991.

- [141] M.S. Lee, Y.M. Yang, and S.W. Lee, "Automatic video parsing using shot boundary detection and camera operation analysis," *Pattern Recognition*, vol. 7, pp. 711–719, March 2001.
- [142] M. Bertini, A. Del Bimbo, and P. Pala, "Content based indexing and retrieval of TV news," *Pattern Recognition Letters*, vol. 22, pp. 503–516, April 2001.
- [143] D. Swanberg, C.F. Shu, and R. Jain, "Knowledge guided parsing and retrieval in video databases," *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases, San Jose, CA*, pp. 13–24, February 1993.
- [144] Y. Chahir and L. Chen, "Automatic video segmentation and indexing," *Proceedings of SPIE Conference on Intelligent Robots and Computer Vision, Boston, MA*, pp. 345–356, September 1999.
- [145] H. Yu, G. Bozdagi, and S. Harrington, "Feature-based hierarchical video segmentation," *Proceedings of IEEE International Conference on Image Processing (ICIP'97), Santa Barbara, CA*, pp. 498–501, October 1997.
- [146] A.F. Smeaton, G. Gormley J. Gilvarry, B. Tobin, S. Marlow, and N. Murphy, "An evaluation of alternative techniques for automatic detection of shot boundaries in digital video," *Proceedings of Machine Vision and Image Processing Conference, Dublin, Ireland*, pp. 45–62, September 1999.
- [147] W.J. Heng and K.N. Ngan, "Integrated shot boundary detection using object-based technique," *Proceedings of IEEE International Conference on Image Processing (ICIP'99), Kobe, Japan*, pp. 289–293, October 1999.
- [148] J. Nam and A.H. Tewfik, "Combined audio and visual streams analysis for video sequence segmentation," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97), Munich, Germany*, pp. 2665–2668, April 1997.
- [149] J. Nam and A.H. Tewfik, "Wipe transition detection using polynomial interpolation," *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases, San Jose, CA*, pp. 231–241, January 2001.
- [150] A. Hampapur, R.C. Jain, and T. Weymouth, "Production model based digital video segmentation," *Multimedia Tools and Applications*, vol. 1, pp. 9–46, March 1995.

- [151] A.M. Alattar, "Detecting fade regions in uncompressed video sequences," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97)*, Munich, Germany, pp. 3025–3028, April 1997.
- [152] A.M. Alattar, "Detecting and compressing dissolve regions in video sequences with Dvi multimedia image compression algorithm," *Proceedings of IEEE International Symposium on Circuits and Systems, Chicago, IL*, pp. 13–16, May 1993.
- [153] W.A.C. Fernando, C.N. Canagarajah, and D.R. Bull, "Fade and dissolve detection in uncompressed and compressed video sequences," *Proceedings of IEEE International Conference on Image Processing (ICIP'99)*, Kobe, Japan, pp. 24–28, October 1999.
- [154] B.T. Truong, C. Dorai, and S. Venkatesh, "New enhancements to cut, fade, and dissolve detection processes in video segmentation," *Proceedings of ACM Multimedia, Los Angeles, CA*, pp. 219–227, October 2000.
- [155] N. Vasconcelos and A. Lippman, "Statistical models of video structure for content analysis and characterization," *IEEE Transactions on Image Processing*, vol. 9, pp. 3–19, January 2001.
- [156] A. Hanjalic and H.J. Zhang, "Optimal shot boundary detection based on robust statistical models," *Proceedings of IEEE International Conference on Multimedia Computing and Systems, Florence, Italy*, vol. 9, pp. 710–714, June 1999.
- [157] K.J. Han and A.H. Tewfik, "Eigen-image based video segmentation and indexing," *Proceedings of IEEE International Conference on Image Processing (ICIP'97)*, Santa Barbara, CA, pp. 538–541, October 1997.
- [158] A. Yilmaz and M. Shah, "Shot detection using principle coordinate system," *International Conference on Internet and Multimedia Systems and Applications, Las Vegas, CA*, pp. 168–174, October 2000.
- [159] Y. Gong and X. Liu, "Video shot segmentation and classification," *Proceedings of IEEE International Conference on Pattern Recognition, Barcelona, Spain*, pp. 860–863, September 2000.
- [160] S. Eickeler and S. Müller, "Content-based video indexing of tv broadcast news using Hidden Markov Models," *Proceedings of IEEE International Conference on*

References

- Acoustics, Speech, and Signal Processing (ICASSP'99)*, Phoenix, AZ, pp. 2997–3000, March 1999.
- [161] J.S. Boreczky and L.D. Wilcox, “A Hidden Markov Model framework for video segmentation using audio and image features,” *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'98)*, Seattle, WA, pp. 2741–3744, May 1998.
- [162] Z.N. Li and J. Wei, “Spatio-temporal joint probability images for video segmentation,” *Proceedings of IEEE International Conference on Image Processing (ICIP'00)*, Vancouver, Canada, pp. 295–298, September 2000.
- [163] M. Cherfaoui and C. Bertin, “Temporal segmentation of videos: a new approach,” *Proceedings of SPIE Conference on Digital Video Compression: Algorithms and Technologies*, San Jose, CA, pp. 38–47, February 1995.
- [164] P. Bouthemy, M. Gelgon, and F. Ganansia, “A unified approach to shot change detection and camera motion characterization,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 1030–1044, October 1999.
- [165] A. Akutsu, Y. Tonomura, H. Hashimoto, and Y. Ohba, “Video indexing using motion vectors,” *Proceedings of SPIE Conference on Visual Communications and Image Processing*, Boston, MA, pp. 1522–1530, November 1992.
- [166] B. Shahraray, “Scene change detection and content-based sampling of video sequences,” *Proceedings of SPIE Conference on Digital Video Compression: Algorithms and Technologies*, San Jose, CA, pp. 2–13, February 1995.
- [167] S.V. Porter, M. Mirmehdi, and B.T. Thomas, “Video cut detection using frequency domain correlation,” *Proceedings of IEEE International Conference on Pattern Recognition*, Barcelona, Spain, pp. 413–416, September 2000.
- [168] G. Quénot and P. Mulhem, “Two systems for temporal video segmentation,” *Proceedings of European Workshop on Content Based Multimedia Indexing*, Toulouse, France, pp. 187–194, October 1999.
- [169] J.M. Gauch, S. Gauch, S. Bouix, and X. Zhu, “Real time video scene detection and classification,” *Information Processing and Management*, vol. 9, pp. 401–420, May 1999.

- [170] Y. Yusoff, J. Kittler, and W. Christmas, "Combining multiple experts for classifying shot changes in video sequences," *Proceedings of IEEE International Conference on Multimedia Computing and Systems, Florence, Italy*, pp. 700–704, June 1999.
- [171] A. Vellakal and C.C.J. Kuo, "Joint spatial-spectral indexing for image retrieval," *Proceedings of IEEE International Conference on Image Processing (ICIP'96), Lausanne, Switzerland*, pp. 867–870, September 1996.
- [172] M.R. Naphade, P. Mehrotra, A.M. Ferman, J. Warnick, T.S. Huang, and A.M. Tekalp, "A highperformance shot boundary detection algorithm using multiple cues," *Proceedings of IEEE International Conference on Image Processing (ICIP'98), Chicago, IL*, pp. 884–887, October 1998.
- [173] B. Gunsel, A.M. Ferman, and A.M. Tekalp, "Temporal video segmentation using unsupervised clustering and semantic object tracking," *Electronic Imaging*, vol. 7, pp. 592–604, July 1998.
- [174] A.M. Ferman and A.M. Tekalp, "Efficient filtering and clustering for temporal video segmentation and visual summarization," *Visual Communication and Image Representation*, vol. 9, pp. 336–351, December 1998.
- [175] H.C. Lee, C.W. Lee, and S.D. Kim, "Abrupt shot change detection using an unsupervised clustering of multiple features," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'00), Istanbul, Turkey*, pp. 2015–2018, June 2000.
- [176] J.H. Oh, K.A. Hua, and N. Liang, "A content-based scene change detection and classification technique using background tracking," *Proceedings of SPIE Conference on Multimedia Computing and Networking, San Jose, CA*, pp. 254–265, January 2000.
- [177] F. Arman, A. Hsu, and M.Y. Chiu, "Image processing on compressed data for large video databases," *Proceedings of ACM Multimedia, Anaheim, CA*, pp. 267–272, August 1993.
- [178] H. Zhang, C. Low, Y. Gong, and S. Smoliar, "Video parsing using compressed data," *Proceedings of SPIE Conference on Image and Video Processing, San Jose, CA*, pp. 142–149, February 1994.

- [179] N. Patel and I. Sethi, "Video shot detection and characterization for video databases," *Pattern Recognition*, vol. 30, pp. 583–592, April 1997.
- [180] K. Shen and E. Delp, "A fast algorithm for video parsing using MPEG compressed sequences," *Proceedings of IEEE International Conference on Image Processing (ICIP'95)*, Washington, DC, pp. 252–255, October 1995.
- [181] C. Taskiran and E. Delp, "Video scene change detection using the generalized sequence trace," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'98)*, Seattle, WA, pp. 2961–2964, May 1998.
- [182] I. Sethi and N. Patel, "A statistical approach to scene change detection," *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases*, San Diego, CA, pp. 329–338, February 1995.
- [183] E. Ardizzone, G. Gioiello, M. L. Cascia, and D. Molinelli, "A real-time neural approach to scene cut detection," *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases*, San Jose, CA, pp. 243–252, February 1996.
- [184] D. Lelescu and D. Schonfeld, "Statistical sequential analysis for realtime video scene change detection on compressed multimedia bitstream," *IEEE Transactions on Multimedia*, vol. 5, pp. 106–117, March 2003.
- [185] J. Meng, Y. Juan, and S.F. Chang, "Scene change detection in a MPEG compressed video sequence," *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases*, San Jose, CA, pp. 14–25, February 1995.
- [186] J. Feng, K.T. Lo, and H. Mehrpour, "Scene change detection algorithm for mpeg video sequence," *Proceedings of IEEE International Conference on Image Processing (ICIP'96)*, Lausanne, Switzerland, pp. 821–824, September 1996.
- [187] H. Zhang, C. Low, and S. Smoliar, "Video parsing and browsing using compressed data," *Multimedia Tools and Applications*, vol. 1, pp. 89–111, March 1995.
- [188] I. Koprinska and S. Carrato, "Video segmentation: A survey," *Signal Processing: Image Communication*, vol. 16, pp. 477–500, January 2001.

References

- [189] S. Eickeler and G. Rigoll, "A novel error measure for the evaluation of video indexing systems," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '00)*, Istanbul, Turkey, pp. 1991–1994, June 2000.
- [190] R. Ruiloba, P. Joly, S. Marchand-Maillet, and G. Quénot, "Towards a standard protocol for the evaluation of video-to-shots segmentation algorithms," *Proceedings of European Workshop on Content Based Multimedia Indexing*, Toulouse, France, pp. 41–48, October 1999.
- [191] G. Ahanger and T.D.C. Little, "A survey of technologies for parsing and indexing digital video," *Journal of Visual Communication and Image Representation*, vol. 7, pp. 28–43, March 1996.
- [192] J.S. Boreczky and L.A. Rowe, "A comparison of video shot boundary detection techniques," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, San Diego, CA, pp. 170–179, February 1996.
- [193] K.S. Fu and J.K. Mui, "A survey on image segmentation," *Pattern Recognition*, vol. 13, pp. 3–16, January 1981.
- [194] R.M. Haralick and L.G. Shapiro, "Survey on image segmentation techniques," *Computer Vision Graphics and Image Processing*, vol. 29, pp. 100–132, January 1985.
- [195] N.R. Pal and S.K. Pal, "A review on image segmentation techniques," *Pattern Recognition*, vol. 26, pp. 1277–1294, September 1993.
- [196] L. Lucchese and S.K. Mitra, "Advances in color image segmentation," *Proceedings of Global Telecommunications Conference Globecom*, pp. 2038–2044, Decembrie 1999.
- [197] W. Skarbek and A. Koschan, "Colour image segmentation: A survey," *Technical Report, Computer Science Department, Berlin Technical University*, October 1994.
- [198] H.D. Cheng, X.H. Jiang, Y. Sun, and J. Wang, "Color image segmentation: Advances and prospects," *Pattern Recognition*, vol. 34, pp. 2259–2281, Septembrie 2001.
- [199] L. Lucchese and S.K. Mitra, "Color image segmentation: A state of the art survey," *Proceedings of Indian National Science Academy*, vol. A, pp. 207–221, March 2001.

References

- [200] R. Ohlander, K. Price, and D.R. Reddy, "Picture segmentation using a recursive region splitting method," *Computer Graphics and Image Processing*, vol. 8, pp. 313–333, 1978.
- [201] M. Cheriet, J.N. Said, and C.Y. Suen, "A recursive thresholding technique for image segmentation," *IEEE Transactions on Image Processing*, vol. 7, pp. 918–920, June 1998.
- [202] B. Bhanu and O.D. Faugeras, "Segmentation of images having unimodal distributions," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 4, pp. 408–419, April 1982.
- [203] B. Bhanu and B.A. Parvin, "Segmentation of natural sceness," *Pattern Recognition*, vol. 20, pp. 487–496, May 1987.
- [204] L. Li, J. Gong and W. Chen, "Gray-level image thresholding based on fisher linear projection of two-dimensional histogram," *Pattern Recognition*, vol. 30, pp. 743–749, May 1997.
- [205] F.H.Y. Chan, F. K. Lam and H. Zhu, "Adaptive thresholding by variational method," *IEEE Transactions on Image Processing*, vol. 2, pp. 168–174, March 1998.
- [206] M. Celenk and M.U. de Haag, "Optimal thresholding for color images," *Proccedings of SPIE Nonlinear Image Processing, San Jose, CA*, pp. 250–259, January 1998.
- [207] L. Shafarenko, M. Petrou, and J. Kittler, "Automatic watershed segmentation of randomly textured color images," *IEEE Transactions On Image Processing*, vol. 6, pp. 1530–1544, November 1997.
- [208] D.C. Tseng, Y.F. Li, and C.T. Tung, "Circular histogram thresholding for color image segmentation," *Proccedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, Canada*, pp. 673–676, August 1995.
- [209] G. Guo, S. Yu, and S. Ma, "Unsupervised segmentation of color images," *Proccedings of IEEE International Conference on Image Processing (ICIP'98), Chicago, IL*, pp. 299–302, October 1998.

- [210] E. Saber, A.M. Tekalp, R. Eschbach, and K. Knox, "Annotation of natural scenes using adaptive color segmentation," *Proceedings of SPIE Image and Video Processing*, San Jose, CA, pp. 72–80, February 1995.
- [211] L.J. Liu, J.F. Lu, J.Y. Yang, K. Liu, Y.G. Wu, and S.J. Li, "Efficient segmentation of nuclei in different color spaces," *Proceedings of SPIE Applications of Digital Image Processing*, San Diego, CA, pp. 773–778, July 1994.
- [212] S.H. Park, I.D. Yun, and S.U. Lee, "Color image segmentation based on 3D clustering: Morphological approach," *Pattern Recognition*, vol. 31, pp. 1061–1076, August 1998.
- [213] K. Takahashi and K. Abe, "Color image segmentation using ISODATA clustering algorithm," *Transactions of the Institute of Electronics, Information and Communication Engineers of Japan*, vol. J82D-II, pp. 751–762, April 1999.
- [214] S. Ray, R. H. Turi, and P. E. Tischer, "Clustering-based colour image segmentation: An evaluation study," *Proceedings of Digital Image Computing: Technology and Applications (DICTA '95)*, Brisbane, Australia, pp. 86–92, December 1995.
- [215] D. Comaniciu and P. Meer, "Robust analysis of feature spaces: Color image segmentation," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'97)*, San Juan, Puerto Rico, pp. 750–755, June 1997.
- [216] D. Comaniciu and P. Meer, "Mean shift analysis and applications," *Proceedings of IEEE International Conference on Computer Vision (ICCV'99)*, Kerkyra, Greece, pp. 1197–1203, September 1999.
- [217] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 603–619, May 2002.
- [218] S. Belongie, C. Carson, H. Greenspan, and J. Malik, "Color- and texture-based image segmentation using EM and its application to content-based image retrieval," *Proceedings of IEEE International Conference on Computer Vision (ICCV'98)*, Mumbai, India, pp. 675–682, January 1998.

References

- [219] A.K. Jain, R. Dubes, and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 4–38, January 2000.
- [220] G. McLachlan and . Krishnan, *The EM Algorithm and Extensions*. John Wiley and Sons, 1997.
- [221] G. McLachlan and D. Peel, *Finite Mixture Models*. John Wiley and Sons, 2000.
- [222] N.H.C. Yung and H.S. Lai and H.S. Lai, "Segmentation of color images based on the gravitational clustering concept," *SPIE Optical Engineering*, vol. 37, pp. 989–1000, March 1998.
- [223] K. Uchimura, "Color images segmentation using tree representation," *Transactions of the Institute of Electrical Engineers of Japan*, vol. 114-C, pp. 1320–1321, December 1994.
- [224] W. Wang, C. Sun, and H. Chao, "Color image segmentation and understanding through connected components," *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, Orlando, Florida*, pp. 1089–1093, October 1997.
- [225] T.N. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Transactions on Signal Processing*, vol. 40, pp. 901–913, April 1992.
- [226] M.M. Chang, I. Sezan, and M. Tekalp, "Adaptive Bayesian segmentation of color images," *Journal of Electronic Imaging*, vol. 3, pp. 404–414, April 1994.
- [227] J.M. Keller and C.L. Carpenter, "Image segmentation in the presence of uncertainty," *International Journal of Intelligent Systems*, vol. 5, pp. 193–208, June 1990.
- [228] J.M. Keller, M.R. Gray, and J.A. Givens, "A fuzzy K-nearest neighbor algorithm," *IEEE Transactions on System Science and Cybernetics*, vol. SMC-15, pp. 580–585, August 1985.
- [229] T.L. Huntsberger, C.L. Jacobs, and R.L. Cannon, "Iterative fuzzy image segmentation," *Pattern Recognition*, vol. 18, pp. 131–138, February 1985.
- [230] A. Moghaddamzadeh and N. Bourbakis, "A fuzzy region growing approach for segmentation of color images," *Pattern Recognition*, vol. 30, pp. 867–881, June 1997.

- [231] Y.W. Lim and S.U. Lee, "On the color image segmentation algorithm based on the thresholding and the fuzzy C-means techniques," *Pattern Recognition*, vol. 23, pp. 935–952, September 1990.
- [232] H. D. Cheng and J. Li, "Fuzzy homogeneity and scale space approach to color image segmentation," *Pattern Recognition*, vol. 36, pp. 1545–1562, July 2003.
- [233] J. Freixenet, X. Muñoz, D. Raba, J. Martí, X. Cufí, "Yet another survey on image segmentation: Region and boundary information integration," *Proceedings of European Conference on Computer Vision, Copenhagen, Denmark*, pp. 408–418, May 2002.
- [234] A. Tremeau and N. Borel, "A region growing and merging algorithm to color segmentation," *Pattern Recognition*, vol. 30, pp. 1191–1204, July 1997.
- [235] Y. Kanai, "Image segmentation using intensity and color information," *Proceedings of SPIE Visual Communications and Image Processing, San Jose, CA*, pp. 709–720, January 1998.
- [236] B. Cramariuc, M. Gabbouj, and J. Astola, "Clustering based region growing algorithm for color image segmentation," *Proceedings of International Conference on Digital Signal Processing, San Jose, CA*, pp. 857–860, July 1997.
- [237] Y. Deng, B.S. Manjunath, and H. Shin, "Color image segmentation," *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR'99), Fort Collins, CO*, pp. 2446–2451, June 1999.
- [238] V. Rehrmann and L. Priese, "Fast and robust segmentation of natural color scenes," *Proceedings of Asian Conference on Computer Vision (ACCV'98), Hong Kong*, pp. 598–606, January 1998.
- [239] P. Colantoni and B. Laget, "Color image segmentation using region adjacency graphs," *Proceedings of International Conference on Image Processing and Applications, Dublin, Ireland*, pp. 698–702, July 1997.
- [240] D.K. Panjwani and G. Healey, "Markov random field models for unsupervised segmentation of textured color images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 939–954, October 1995.

- [241] M. Celenk, "Hierarchical color clustering for segmentation of textured images," *Proceedings of Southeastern Symposium on System Theory (SSST'97)*, Cookeville, TN, pp. 483–487, March 1997.
- [242] S. Ji and H.W. Park, "Image segmentation of color image based on region coherency," *Proceedings of IEEE International Conference on Image Processing (ICIP'98)*, Chicago, IL, pp. 80–83, October 1998.
- [243] K. Saarinen, "Color image segmentation by a watershed algorithm and region adjacency graph processing," *Proceedings of IEEE International Conference on Image Processing (ICIP'94)*, Austin, TX, pp. 1021–1025, November 1994.
- [244] T. Gevers and A.W.M. Smeulders, "Combining region splitting and edge detection through guided Delaunay image subdivision," *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR'97)*, Puerto Rico, pp. 1021–1026, June 1997.
- [245] R. Schettini and M. Suardi, "A low-level segmentation procedure for color images," *Proceedings of European Signal Processing Conference (EUSIPCO-94)*, Edinburgh, UK, pp. 26–29, September 1994.
- [246] C.T. Zahn, "Graph-theoretical methods for detecting and describing gestalt clusters," *IEEE Transactions on Computers*, vol. 20, pp. 68–86, October 1971.
- [247] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its applications to image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 1101–1113, November 1993.
- [248] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR'97)*, Puerto Rico, pp. 731–737, June 1997.
- [249] J. Shi, S. Belongie, T. Leung, and J. Malik, "Image and video segmentation: The normalized cut framework," *Proceedings of IEEE International Conference on Image Processing (ICIP'98)*, Chicago, IL, pp. 943–947, October 1998.
- [250] R. Urquhart, "Graph theoretical clustering based on limited neighborhood sets," *Pattern Recognition*, vol. 13, pp. 173–187, March 1982.

- [251] P.F. Felzenszwalb and D.P. Huttenlocher, "Image and video segmentation: The normalized cut framework," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'98)*, Santa Barbara, CA, pp. 98–104, June 1998.
- [252] J.P. Wang, "Stochastic relaxation on partitions with connected components and its application to image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 619–635, June 1998.
- [253] T. Vlachos and A.G. Constantinides, "Graph-theoretical approach to colour picture segmentation and contour classification," *IEE Proceedings Communications, Speech and Vision*, vol. 140, pp. 36–45, February 1993.
- [254] Y. Xu and E.C. Uberbacher, "2D image segmentation using minimum spanning trees," *Image and Vision Computing*, vol. 15, pp. 47–57, January 1997.
- [255] M. Chapron, "A new chromatic edge detector used for color image segmentation," *International Conference on Pattern Recognition, The Hague, Netherlands*, pp. 311–314, August 1992.
- [256] M. Chapron, "A chromatic contour detector based on abrupt change techniques," *Proceedings of IEEE International Conference on Image Processing (ICIP'97)*, Santa Barbara, CA, pp. 18–21, October 1997.
- [257] A. Cumani, "Edge detection in multispectral images," *Computer Vision, Graphics and Image Processing: Graphical Models and Image Processing*, vol. 53, pp. 40–51, October 1991.
- [258] J.M. Prager, "Extracting and labeling boundary segments in natural scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, pp. 16–27, January 1980.
- [259] W.A. Perkins, "Area segmentation of images using edge points," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, pp. 8–15, January 1980.
- [260] W.Y. Ma and B.S. Manjunath, "Edge flow: A framework of boundary detection and image segmentation," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'97)*, Puerto Rico, pp. 744–749, June 1997.

- [261] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, pp. 321–331, April 1987.
- [262] G. Sapiro, "Vector (self) snakes: A geometric framework for color, texture and multiscale image segmentation," *Proceedings of IEEE International Conference on Image Processing (ICIP'96)*, Lausanne, Switzerland, pp. 817–820, October 1997.
- [263] G. Sapiro, "Color snakes," *Computer Vision and Image Understanding*, vol. 68, pp. 247–253, February 1997.
- [264] T. Gevers, S. Ghebreab, and A.W.M. Smeulders, "Color invariant snakes," *Proceedings of the 9th British Machine Vision Conference (BMVC'98)*, Southampton, UK, pp. 578–588, September 1998.
- [265] G. Sapiro, "Color and illuminant voting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, pp. 1210–1215, November 1999.
- [266] P. Campadelli, D. Medici, and R. Schettini, "Color image segmentation using hop-field networks," *Image and Vision Computing*, vol. 15, pp. 161–166, March 1997.
- [267] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 23–38, January 1998.
- [268] H.C. Fu, P.S. Lai, R.S. Lou, and H.T. Pao, "Face detection and eye localization by neural network based color segmentation," *Proceedings of IEEE Signal Processing Society Workshop, Neural Networks for Signal Processing*, Sydney, Australia, pp. 507–516, December 2000.
- [269] H. Okii, N. Kaneki, H. Hara, and K. Ono, "Automatic color segmentation method using a neural network model for stained images," *IEICE Transactions on Information and Systems, Japan*, vol. 3, pp. 343–350, March 1994.
- [270] S.N. Krjukov, T.O. Semenkova, V.A. Pavlova, and B.I. Arnt, "Backpropagation neural network for adaptive color image segmentation," *Proceedings of SPIE Applications of Artificial Neural Networks in Image Processing*, San Jose, CA, pp. 70–74, February 1997.

- [271] M. Sammouda, R. Sammouda, N. Niki, and K. Mukai, "Segmentation and analysis of liver cancer pathological color images based on artificial neural networks," *Proceedings of IEEE International Conference on Image Processing (ICIP'99), Kobe, Japan*, pp. 392–396, October 1999.
- [272] D. Goldman, M. Yang, and N. Bourbakis, "A neural network-based segmentation tool for color images," *Proceedings of IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2002), Washington, DC*, pp. 500–511, November 2002.
- [273] S.A. Shafer, "Using color to separate reflection components," *Color Research and Application*, vol. 10, pp. 210–218, April 1985.
- [274] R. Bajcsy, S.W. Lee, and A. Leonardis, "Detection of diffuse and specular interface reflections and interreflections by color image segmentation," *International Journal of Computer Vision*, vol. 17, pp. 241–272, March 1996.
- [275] G.E. Healey, *Color image segmentation. Physics-Based Vision Principles and Practice Color*. Jones and Bartlett Publishers, Boston, 1992.
- [276] G.E. Healey, *Segmenting images using normalized color. Physics-Based Vision Principles and Practice Color*. Jones and Bartlett Publishers, Boston, 1992.
- [277] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, "Performance of optical flow techniques," *International Journal of Computer Vision*, vol. 12, pp. 43–77, January 1994.
- [278] A. Mitiche and P. Bouthemy, "Computation and analysis of image motion: A synopsis of current problems and methods," *International Journal of Computer Vision*, vol. 19, pp. 29–55, January 1996.
- [279] R. Mech and M. Wollborn, "A noise robust method for segmentation of moving objects in video sequences," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'97), Munich, Germany*, pp. 2657–2660, April 1997.
- [280] C. Kim and J.N. Hwang, "A fast and robust moving object segmentation in video sequences," *Proceedings of IEEE International Conference on Image Processing (ICIP'99), Kobe, Japan*, pp. 131–134, October 1999.

References

- [281] T. Meier and K. N. Ngan, "Extraction of moving objects for content-based video coding," *Proceedings of SPIE Visual Communications and Image Processing, San Jose, CA*, pp. 1178–1189, January 1999.
- [282] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision*, vol. 22, pp. 61–79, January 1997.
- [283] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Fast geodesic active contours," *IEEE Transactions on Image Processing*, vol. 10, pp. 1467–1475, October 2001.
- [284] N. Paragios and R. Deriche, "Geodesic active contours and level sets for the detection and tracking of moving objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 266–280, March 2000.
- [285] E. Sifakis, C. Garcia, and G. Tziritas, "Bayesian level sets for image segmentation," *Journal of Visual Communication and Image Representation*, vol. 13, pp. 44–64, March 2002.
- [286] E. Sifakis, I. Grinias, and G. Tziritas, "Video segmentation using fast marching and region growing algorithms," *EURASIP Journal On Applied Signal Processing*, vol. 2002, pp. 379–388, April 2002.
- [287] A.R. Mansouri, B. Sirivong, and J. Konrad, "Multiple motion segmentation with level sets," *Proceedings of SPIE Visual Communications and Image Processing, Perth, Australia*, pp. 584–595, June 2000.
- [288] G.M. Nielson and B. Hamann, "Techniques for the interactive visualization of volumetric data," *Proceedings of IEEE Conference on Visualization, San Francisco, CA*, pp. 45–50, October 1990.
- [289] E.N. Mortensen and W.A. Barrett, "Intelligent scissors for image composition," *Proceedings of ACM SIGGRAPH'95, Los Angeles, CA*, pp. 191–198, August 1995.
- [290] Y.Y. Boykov and M.P. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images," *Proceedings of IEEE International Conference on Computer Vision, Vancouver, Canada*, pp. 105–112, July 2001.

References

- [291] A. Blake, A. Rother, M. Brown, P. Perez, and P. Torr, "Interactive image segmentation using an adaptive GMMRF model," *Proceedings of European Conference on Computer Vision (ECCV 2004)*, Prague, Czech Republic, pp. 428–441, May 2004.
- [292] N. O'Connor and S. Marlow, "Supervised semantic object segmentation and tracking via EM-based estimation of mixture density parameters," *Proceedings of Noblesse Workshop on Non-Linear Model Based Image Analysis*, Glasgow, UK, pp. 165–174, July 1998.
- [293] F. Meyer, "Morphological multiscale and interactive segmentation," *Proceedings of IEEE-EURASIP Workshop on Non-Linear Signal and Image Processing (NSIP99)*, Antalya, Turkey, pp. 369–377, June 1999.
- [294] F. Marqués, B. Marcotegui, F. Zanoguera, P. Correia, R. Mech, and M. Wollborn, "Partition-based image representation as basis for user-assisted segmentation," *Proceedings of IEEE International Conference on Image Processing (ICIP'00)*, Vancouver, Canada, pp. 312–315, September 2000.
- [295] B. Marcotegui and F. Zanoguera, "Image editing tools based on multi-scale segmentation," *Proceedings of International Symposium On Mathematical Morphology*, Sydney, Australia, pp. 127–135, April 2002.
- [296] S. Cooray, N. O'Connor, S. Marlow, N. Murphy and T. Curran, "Hierarchical semi-automatic video object segmentation for multimedia applications," *Proceedings of SPIE ITCOM, Denver, CO*, pp. 86–91, August 2001.
- [297] N. O'Connor, T. Adamek, S. Sav, N. Murphy and S. Marlow, "QIMERA: A software platform for video object segmentation and tracking," *Proceedings of the 4th Workshop on Image Analysis for Multimedia Interactive Service (WIAMIS 2003)*, London, UK, pp. 43–50, April 2003.
- [298] J.P. Schober, T. Hermes and O. Herzog, "Picturefinder: Description logics for semantic image retrieval," *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2005)*, Amsterdam, The Netherlands, pp. 1571–1574, July 2005.
- [299] J.S. Cardoso and L. Corte-Real, "Toward a generic evaluation of image segmentation," *IEEE Transactions on Image Processing*, vol. 14, pp. 1773–1782, November 1998.

References

- [300] B. McCane, "On the evaluation of image segmentation algorithms," *Proceedings of Digital Image Computing: Techniques and Applications (DICTA'97)*, Auckland, New Zealand, pp. 455–460, December 1997.
- [301] P. Villegas and X. Marichal, "Perceptually-weighted evaluation criteria for segmentation masks in video sequences," *IEEE Transactions on Image Processing*, vol. 13, pp. 1092–1103, August 2004.
- [302] Y.J. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognition*, vol. 29, pp. 1335–1346, August 1996.
- [303] M. Borsotti, P. Campdelli, and P. Schettini, "Quantitative evaluation of color image segmentation results," *Pattern Recognition Letters*, vol. 19, pp. 741–747, June 1998.
- [304] M.D. Levine and A. Nazif, "Dynamic measurement of computer generated image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, pp. 155–164, February 1985.
- [305] R. Mech and F. Marqués, "Objective evaluation criteria for 2-d shape estimation results of moving objects," *Proceedings of the 2nd Workshop on Image Analysis for Multimedia Interactive Service (WIAMIS 2001)*, Tampere, Finland, pp. 23–28, May 2001.
- [306] G. Salton and C. Buckley, "Improving retrieval performance by relevance feedback," *Journal of the American Society for Information Science*, vol. 41, pp. 288–297, 1990.
- [307] E. Ide, "New experiments in relevance feedback," *The Smart System-Experiments In Automatic Document Processing*, Englewood Cliffs, NJ: Prentice Hall, pp. 337–354, 1971.
- [308] G. Salton, "Relevance feedback and the optimization of retrieval effectiveness," *The Smart System-Experiments In Automatic Document Processing*, Englewood Cliffs, NJ: Prentice Hall, pp. 337–354, 1971.
- [309] S.E. Robertson and K. Sparck Jones, "Relevance weighting of search terms," *Journal of the American Society for Information Science*, vol. 27, pp. 129–146, 1976.
- [310] C.J. van Rijsbergen, *Information retrieval*. (2nd ed.) London: Butterworths, UK, 1979.

References

- [311] S.E. Robertson, C.J. van Rijsbergen, and M.F. Porter, "Probabilistic models of indexing and searching," *Information retrieval research, London:Butterworths, UK*, pp. 35–56, 1981.
- [312] J.G. Dy, C.E. Brodley, A. Kak, C. Shyu, and L.S. Broderick, "The customized-queries approach to CBIR," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases VII*, pp. 22–32, January 1999.
- [313] J. Peng, B. Bhanu, and S. Qing, "Probabilistic feature relevance learning for content-based image retrieval," *Computer Vision and Image Understanding*, vol. 75, pp. 150–164, July 1999.
- [314] I.J. Cox, M.L. Miller, T.P. Minka, and P.N. Yianilos, "An optimized interaction strategy for Bayesian relevance feedback," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'98), Santa Barbara, CA*, pp. 553–558, June 1998.
- [315] M. Crucianu, M. Ferecatu, and N. Boujemaa, "Relevance feedback for image retrieval: A short review," *State Of The Art In Audiovisual Content-Based Retrieval, Information Universal Access And Interaction Including Data Models And Languages, DELOS2 Report (FP6 NoE)*, 2004.
- [316] J. Wang, A.P. de Vries, and M.J.T. Reinders, "A user-item relevance model for log-based collaborative filtering," *Proceedings of European Conference on Information Retrieval (ECIR 2006), London, UK*, pp. 37–48, April 2006.
- [317] A.F. Smeaton, H. Lee, C. Foley, S. McGivney, and C. Gurrin, "Físchlár-DiamondTouch: Collaborative video searching on a table," *Proceedings of SPIE Electronic Imaging - Multimedia Content Analysis, Management, and Retrieval, San Jose, CA*, pp. 37–48, January 2006.
- [318] Y. Ishikawa, R. Subramanya, and C. Faloutsos, "Mindreader: Query databases through multiple examples," *Proceedings of the 24th International Conference on Very Large Data Bases, New York*, pp. 218–227, August 1998.
- [319] Y. Rui and T.S. Huang, "Optimizing learning in image retrieval," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'00), Hilton Head Island, SC*, pp. 236–245, June 2000.

- [320] C. Nastar , M. Mitschke and C. Meilhac, "Efficient query refinement for image retrieval," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'98)*, Santa Barbara, CA, pp. 547–552, June 1998.
- [321] K. Tieu and P. Viola, "Boosting image retrieval," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'00)*, Head Island, SC, pp. 228–235, June 2000.
- [322] X.S. Zhou and T.S. Huang, "Relevance feedback in image retrieval: A comprehensive review," *ACM Multimedia Systems*, vol. 8, pp. 536–544, April 2000.
- [323] Y. Wu, Q. Tian, and T.S. Huang, "Discriminant-EM algorithm with application to image retrieval," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'00)*, Head Island, SC, pp. 222–227, June 2000.
- [324] B. Moghaddam, Q. Tian, N. Lesh, C. Shen, and T.S. Huang, "Visualization and layout for personal photo libraries," *International Workshop on Content-based Multimedia Indexing (CBMI 2001)*, University of Brescia, Italy, September 2001.
- [325] Y. Chen, X.S. Zhou, and T.S. Huang, "One-class SVM for learning in image retrieval," *Proceedings of IEEE International Conference on Image Processing (ICIP '01)*, Thessaloniki, Greece, pp. 34–37, October 2001.
- [326] X.S. Zhou and T.S. Huang, "Small sample learning during multimedia retrieval using BiasMap," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'01)*, Hawaii, pp. 11–17, December 2001.
- [327] S.D. MacArthur, C.E. Brodley and C.R. Shyu, "Relevance feedback decision trees in content-based image retrieval," *Proceedings of IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'00)*, Hilton Head, SC, pp. 68–72, June 2000.
- [328] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," *Proceedings of ACM Multimedia*, Ottawa, Canada, pp. 107–118, October 2001.
- [329] N. Vasconcelos and A. Lippman, "Bayesian relevance feedback for content-based image retrieval," *Proceedings of IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'00)*, Hilton Head, SC, pp. 63–67, June 2000.

- [330] M. Wood, N. Campbell, and B. Thomas, "Iterative refinement by relevance feedback in content-based digital image retrieval," *Proceedings of ACM Multimedia, Bristol, UK*, pp. 13–20, September 1998.
- [331] R. Schettini, G. Ciocca, and I. Gagliardi, "Content-based color image retrieval with relevance feedback," *Proceedings of IEEE International Conference on Image Processing (ICIP'99), Kobe, Japan*, pp. 75–79, October 1999.
- [332] N. Cristianini and J. Shawe-Taylor, *An Introduction To Support Vector Machines (and other kernel-based learning methods)*. Cambridge University Press, UK, 2000.
- [333] P. Hong, Q. Tian, and T.S. Huang, "Incorporate support vector machines to content-based image retrieval with relevant feedback," *Proceedings of IEEE International Conference on Image Processing (ICIP 2000), Vancouver, Canada*, pp. 750–753, September 2000.
- [334] F. Jing, M. Li, H.J. Zhang, and B. Zhang, "Learning region weighting from relevance feedback in image retrieval," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '02), Orlando, Florida*, pp. 4088–4091, May 2002.
- [335] T.P. Minka and R.W. Picard, "Interactive learning using a society of models," *Pattern Recognition*, vol. 30, pp. 565–581, April 1997.
- [336] Y. Xu, E. Saber, and A.M. Tekalp, "Hierarchical content description and object formation by learning," *Proceedings of IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'99), Fort Collins, CO*, pp. 84–88, June 1999.
- [337] A.L. Ratan, O. Maron, W.E.L. Grimson, and T. Lozano-Pérez, "A framework for learning query concepts in image classification," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'99), Fort Collins, CO*, pp. 1423–1433, June 1999.
- [338] D.A. Forsyth and M.M. Fleck, "Finding people and animals by guided assembly," *Proceedings of IEEE International Conference on Image Processing (ICIP'97), Santa Barbara, CA*, pp. 5–8, October 1997.

- [339] P. Hong and T.S. Huang, "Spatial pattern discovering by learning the isomorphic sub-graph from multiple attributed relation graphs," *Proceedings of the 8th International Workshop on Combinatorial Image Analysis, Philadelphia, PA*, pp. 19–26, October 2001.
- [340] F. Jing, M. Li, H.J. Zhang, and B. Zhang, "Relevance feedback in region-based image retrieval," *IEEE Transactions On Circuits And Systems For Video Technology*, vol. 14, pp. 672–681, May 2004.
- [341] V. Mezaris, I. Kompatsiaris, and M.G. Strintzis, "Region-based image retrieval using an object ontology and relevance feedback," *EURASIP Journal On Applied Signal Processing*, vol. 2004, pp. 886–901, June 2004.
- [342] Z. Xu, X. Xu, K. Yu, and V. Tresp, "A hybrid relevance feedback approach to text retrieval," *Proceedings of the 25th European Conference on Information Retrieval (ECIR'03), Pisa, Italy*, pp. 281–293, April 2003.
- [343] B. Chawda, B. Craft, P. Cairns, D. Heesch and S. Rüger, "Do "attractive things work better"? an exploration of search tool visualisations," *Proceedings of International Conference on Human-Computer Interaction (HCI 2005 The Bigger Picture), Edinburgh, UK*, pp. 46–51, September 2005.
- [344] Y.K. Chang, C. Cirillo and J. Razon, "Evaluation of feedback retrieval using modified freezing, residual collection & test and control groups," *The Smart System-Experiments In Automatic Document Processing, Englewood Cliffs, NJ: Prentice Hall*, pp. 355–370, 1971.
- [345] N. O'Connor, E. Cooke, H. Le Borgne, M. Blighe, and T. Adamek, "The Ace-Toolbox: Low-level audiovisual feature extraction for retrieval and classification," *Proceedings of IEE European Workshop on the Integration of Knowledge, Semantic and Digital Media Technologies, London, UK*, pp. 229–232, November 2005.
- [346] C.W. Ngo, T.C. Pong, and R. T. Chin, "Video partitioning by temporal slice coherency," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, pp. 941–953, August 2001.
- [347] C.W. Ngo, "A robust dissolve detector by support vector machine," *Proceedings of ACM Multimedia, Berkeley, CA*, pp. 283–286, November 2003.

- [348] M. Cooper, "Video segmentation combining similarity analysis and classification," *Proceedings of ACM Multimedia, New York, NY*, pp. 252–255, October 2004.
- [349] R. Ewerth and B. Freisleben, "Video cut detection without thresholds," *Workshop on Signals, Systems and Image Processing, Poznan, Poland*, pp. 227–230, September 2004.
- [350] G. Boccignone, A. Chianese, and V. Moscato, "Foveated shot detection for video segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, pp. 365–377, March 2005.
- [351] J. Bescos, G. Cisneros, and J.M. Martinez, "A unified model for techniques on video-shot transition detection," *IEEE Transactions on Multimedia*, vol. 7, pp. 293–307, April 2005.
- [352] S. Pei and Y.Z. Chou, "Efficient MPEG compressed video analysis using macroblock type information," *IEEE Multimedia*, vol. 1, pp. 321–333, December 1999.
- [353] J. Calic, S. Sav, E. Izquierdo, S. Marlow, N. Murphy and N. O'Connor, "Temporal video segmentation for real-time key frame extraction," *Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP'02), Orlando, FL*, pp. 3632–3635, May 2002.
- [354] S.H. Kwok and A.G. Constantinides, "A fast recursive shortest spanning tree for image segmentation and edge detection," *IEEE Transactions on Image Processing*, vol. 6, pp. 328–332, February 1997.
- [355] F. Qian, M. Li, L. Zhang, H.J. Zhang, and B. Zhang, "Gaussian mixture model for relevance feedback in image retrieval," *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2002), Lausanne, Switzerland*, pp. 229–232, August 2002.
- [356] P. Over, T. Ianeva, W. Kraaij, and A.F. Smeaton, "Trecvid 2005 - an overview," *Proceedings of Text REtrieval Conference TRECVID Worksho (TRECVID 2005), Gaithersburg, MD*, pp. 229–232, November 2005.
- [357] S. Sav, G. Jones, H. Lee, N. O'Connor, and A.F. Smeaton, "Interactive experiments in object-based retrieval," *Proceedings of International Conference on Image and Video Retrieval (CIVR 2006), Tempe, AZ*, pp. 1–310, July 2006.

Appendix A

List of author's publications

S. Sav, G. Jones, H. Lee, N. O'Connor and A.F. Smeaton, "Interactive Experiments in Object-Based Retrieval", CIVR 2006 - International Conference on Image and Video Retrieval, Tempe, AZ, July 2006.

A.F. Smeaton, G. Jones, H. Lee, N. O'Connor and S. Sav, "Object-Based Access to TV Rushes Video", ECIR 2006 - European Conference on Information Retrieval, London, U.K., April 2006.

C. Foley, C. Gurrin, G. Jones, H. Lee, S. Mc Givney, N. O'Connor, S. Sav, A.F. Smeaton and P. Wilkins, "TRECVID 2005 Experiments at Dublin City University", TRECVID 2005 - Text REtrieval Conference TRECVID, Gaithersburg, ML, November 2005.

S. Sav, H. Lee, A. F. Smeaton, N. E. O'Connor and N. Murphy, "Using Video Objects and Relevance Feedback in Video Retrieval", SPIE 2005 - Multimedia Systems and Applications VIII, Boston, MA, October 2005.

S. Sav, H. Lee, N. O'Connor and A.F. Smeaton, "Interactive Object-based Retrieval Using Relevance Feedback", ACIVS 2005 - Advanced Concepts for Intelligent Vision Systems, Antwerp, Belgium, September 2005.

S. Sav, H. Lee, A.F. Smeaton and N. O'Connor, "Using Segmented Objects in Ostensive Video Shot Retrieval", AMR 2005 - International Workshop on Adaptive Multimedia Retrieval, Glasgow, U.K., July 2005.

S. Sav, N. O'Connor, A.F. Smeaton and N. Murphy, "Associating Low-level Features with Semantic Concepts using Video Objects and Relevance Feedback", WIAMIS 2005 - Workshop on Image Analysis for Multimedia Interactive Service, Montreux, Switzerland, April 2005.

N. O'Connor, S. Sav, T. Adamek, V. Mezaris, I. Kompatsiaris, T.Y. Lui, E. Izquierdo, C.F. Bennstrom and J.R. Casas, "Region and Object Segmentation Algorithms in the QIMERA Segmentation Platform", CBMI 2003 - International Workshop on Content-Based Multimedia Indexing, Rennes, France, September 2003.

N. O'Connor, T. Adamek, S. Sav, N. Murphy and S. Marlow, "QIMERA: A Software Platform for Video Object Segmentation and Tracking", WIAMIS 2003 - Workshop on Image Analysis for Multimedia Interactive Service, London, U.K., April 2003.

J. Calic, S. Sav, E. Izquierdo, S. Marlow, N. Murphy and N. O'Connor, "Temporal Video Segmentation for Real-Time Key Frame Extraction", ICASSP 2002 - International Conference on Acoustics, Speech and Signal Processing, Orlando, FL, May 2002.

P. Browne, C. Gurrin, H. Lee, K. Mc Donald, S. Sav, A.F. Smeaton and J. Ye, "Dublin City University Video Track Experiments for TREC 2001", TREC 2001 - Text REtrieval Conference, Gaithersburg, ML, November 2001.

Appendix B

Examples of retrieved images



Search topic: people walking on the beach



Search topic: boats at sea or in harbour



Search topic: fresh vegetables or fruits



Search topic: bridge



Search topic: farm animals

Appendix B. Examples of retrieved images



Search topic: people in urban settings



Search topic: nightclub life



Search topic: people in traditional dress



Search topic: cars in urban settings



Search topic: historic buildings



Search topic: skyscrapers



Search topic: planes in flight