Syntactic and Semantic Concepts in Audio-Visual Media

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der technischen Wissenschaften

by

Dalibor Mitrović
Registration Number 9925385

Matthias Zeppelzauer
Registration Number 9926063

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Univ. Prof. Dr. Christian Breiteneder

The dissertation has been reviewed by:

(Univ. Prof. Dr. Christian Breiteneder) (Prof. Dr. Harald Kosch)

Vienna, 3.11.2011

(Dalibor Mitrović) (Matthias Zeppelzauer)

Vienna University of Technology
A-1040 Vienna ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

Erklärung zur Verfassung der Arbeit

Dalibor Mitrović, Matthias Zeppelzauer
Favoritenstrasse 9-11, 1040 Wien

Hiermit erklären wir, dass wir diese Arbeit selbständig verfasst haben, dass wir die verwendeten Quellen und Hilfsmittel vollständig angegeben haben und dass wir die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht haben.

(Ort, Datum)	(Dalibor Mitrović)	(Matthias Zeppelzauer)

Acknowledgements

I would like to thank all people who supported me during my PhD studies. First of all, I sincerely thank my parents and my sister who facilitated my studies. I am especially grateful to my girlfriend for being at my side throughout the last years. Additionally, I would like to thank all my friends for their support.

I want to thank my advisor Professor Christian Breiteneder for his support and guidance during my studies. I want to express my special gratitude to my colleague Dalibor Mitrović for the excellent cooperation during the years of working together on this thesis. I further thank my colleague Hofrat Horst Eidenberger, who encouraged me to start my PhD studies. My special thanks go to Maia Zaharieva who particularly enriched our research team. Many thanks go to all members of the institute for their ongoing help and collegiality. I especially thank Ingrid Lissa for taking a lot of administrative work out of my hands. Special thanks further go to the tutors and student assistants who helped me in organizing and holding my lectures. Finally, I thank my second advisor Professor Harald Kosch for his valuable input while writing this thesis.

During my studies, I had the pleasure to supervise the master thesis of Markus Seidl. His work significantly contributed to the results presented in Chapter 6. Further thanks go to my colleagues from the Austrian Film Museum Adelheid Heftberger, Barbara Vockenhuber, and Michael Löbenstein for the good cooperation during our joint research project.

Financial support for this work was obtained by the Vienna Science and Technology Fund (WWTF) and the Austrian Science Fund (FWF) in the research projects CI06024, P23099, and P16111.

Abstract

In recent years tremendous amounts of audio-visual media have become available to the public and increased the demand for efficient access and retrieval methods. The research community working in content-based media retrieval has mainly focused on specific media types, such as sports videos, news broadcasts, and commercials. A widely neglected media type is *historic film* provided by film archives and museums. In the context of historic film, film scientists and archivists have research questions and requirements that are novel for content-based retrieval.

In this thesis, we investigate novel requirements for content-based retrieval of archive film stated by film experts. From the abstract requirements of film experts we first derive specific lower- and higher-level, syntactic and semantic concepts to be retrieved automatically. Next, we develop techniques for the automatic retrieval of these concepts from archive film. The investigated films are challenging for retrieval due to their sophisticated editing and their low material quality which results in numerous artifacts.

The contribution of this thesis are novel techniques and investigations for the retrieval of syntactic and semantic concepts in archive film. We develop detectors for lower-level concepts such as black frames and intertitles and perform comprehensive investigations of shot boundary detection in archive film material. We further analyze higher-level concepts: We propose methods for the extraction of semantically coherent scenes and synchronous audio-visual montage sequences and investigate the retrieval of motion composition and visual composition.

The developed techniques are successfully applied to archive and contemporary films and enable efficient access to the film material. Additionally the methods assist film experts in their investigations and enable them to gain novel insights into the films.

Kurzfassung

In den vergangenen Jahren wurden große Mengen audiovisueller Medien öffentlich zugänglich gemacht. Die so entstandenen Mediensammlungen haben die Nachfrage nach neuen effizienten Methoden für das Informationretrieval erhöht. Forschung, die sich mit Informationretrieval in diesen Mediensammlungen beschäftigt, hat sich überwiegend auf Sport- und Nachrichtenbeiträge sowie Werbefilme konzentriert. Ein bisher weitgehend vernachlässigter Medientyp sind Filme aus Filmarchiven und Filmmuseen. Archivare und Filmwissenschaftler bearbeiten Forschungsfragen und stellen Anforderungen, die im Kontext des Informationretrievals neu sind.

In dieser Dissertation untersuchen wir Anforderungen der Filmexperten an das inhaltsbasierte Informationretrieval von Archivfilmen. Aus den abstrakten Anforderungen der Filmexperten leiten wir syntaktische und semantische Konzepte unterschiedlicher Komplexität ab, welche automatisch aus den Filmen extrahiert werden sollen. Die untersuchten Filme stellen durch ihre komplexe Montage und durch ihren schlechten Materialzustand die automatische Analyse vor neue Herausforderungen.

Der wissenschaftliche Beitrag dieser Arbeit umfasst Untersuchungen und Analysetechniken für die Extraktion von syntaktischen und semantischen Konzepten aus Archivfilmen. Wir entwickeln Detektoren für Konzepte niedriger Komplexität, wie zum Beispiel Schwarzkader und Zwischentitel und untersuchen die Erkennung von Einstellungsgrenzen. Darüber hinaus beschäftigen wir uns mit Konzepten höherer Komplexität: Wir stellen Methoden für die Erkennung von Filmszenen sowie Sequenzen mit synchroner audiovisueller Montage vor. Weiters untersuchen wir die automatisch Analyse von Bewegungskomposition und Bildkomposition.

Die präsentierten Methoden wurden erfolgreich auf Archiv- und zeitgenössische Filme angewendet. Die Methoden ermöglichen einen effizienten Zugriff auf das Material und unterstützen Filmexperten bei ihren Untersuchungen. Weiters zeigt sich, dass die Methoden den Filmexperten zum Teil neue, das Filmmaterial betreffende, Einsichten ermöglichen.

Contents

Chapter 1

Introduction

1.1 Motivation

The field of content-based video retrieval has significantly grown in the last 20 years due to the availability of large amounts of digital media, for example in archives of broadcasting companies, museums, and social media platforms. At the same time research related to the field has grown tremendously. This growth can be observed from the increasing number of publications indexed with the terms "content-based video retrieval" over time. A search in Google scholar for the years 1991 to 1995 returns approximately 9.500 publications. This number doubles in the period from 1996 to 2000 and doubles again from 2001 to 2005. With increasing importance and maturity of research in the field, benchmarks such as TRECVID [198] and VideOlympics [200] have been established that provide large amounts of video material and enable the objective evaluation of retrieval systems.

Most research in content-based video retrieval focuses on special types of video such as news broadcasts, sports videos, and commercials. Frequently researched tasks include automatic video summarization [136], story segmentation of news broadcasts [87], highlight detection in sports video [12], and the detection of commercials [124].

Compared to the retrieval of these special types of content, the retrieval of *film* has received little attention by the research community. Existing approaches focus on the analysis of contemporary Hollywood films [2, 56, 150, 217] whereas *archive* films represent a widely neglected type of film. Present investigations targeting archive

films include query-based retrieval and browsing of historic film archives [164], shot cut detection [225] and automatic summarization [113].

Recently, film scientists have raised novel research questions and have formulated requirements in the context of archive films which have not been analyzed automatically so far. Film scientists focus on stylistic aspects of the films, such as their montage and composition. While many requirements of film scientists are highly abstract and consequently out of the scope for retrieval, there are demands that can be formalized well and that represent challenging tasks for content-based retrieval. In this thesis, we investigate such requirements and present novel retrieval methods for archive film.

This thesis has been performed in the context of an interdisciplinary research project involving film scientists, archivists, and computer scientists on the analysis of archive films. The goal of the project was to gain insights into the highly formalized style of filmmaking of the Soviet filmmaker Dziga Vertov (1896-1954). The interdisciplinary research project offers several opportunities to the domain of content-based retrieval: *First*, the project provides a novel type of film material that differs in composition and quality from the material traditionally employed in content-based retrieval. The films represent a particular challenge for automatic analysis due to their complex and sophisticated stylistic attributes and due to the low quality of the related film material, see Sections 3.1 and 3.2. The material has not been subject to automatic analysis and retrieval so far. *Second*, the requirements stated by the film scientists represent real world problems in the domain of content-based retrieval that have not been researched so far. *Third*, the cooperation with film experts, enables the generation of high-quality annotations and expert ground truths for the objective evaluation of the developed methods.

The basic idea behind the performed investigations in this thesis is to provide retrieval techniques that enable efficient access to the material and that support film scientists and archivists in their work. In a first step we have asked the film scientists to formulate their demands and requirements to automatic film analysis. In a second step, we have derived *concepts*[1] that should be retrieved automatically from the archive film material. In the context of content-based retrieval concepts are represented by instances of *spatio-temporal patterns* in signals. Generally, such patterns may reside in

[1] We regard a concept as "an abstract or generic idea generalized from particular instances" [144].

different modalities: in the visual domain, in the auditory domain, or in both. Consequently, we distinguish between *visual concepts*, *auditory concepts* and *audio-visual concepts*. Each concept has a *syntactic* dimension and a *semantic* dimension where both dimensions are usually differently accentuated. The syntactic dimension describes to which degree a concept represents structural information in a film. Similarly, the semantic dimension describes the amount of meaning associated to a concept. Usually the complexity and diversity of a concept increases as the semantic dimension receives more importance.

The set of identified concepts represents the basis for the development of novel analysis and retrieval methods. An overview of the concepts investigated in this thesis is given in Figure 1.1. We distinguish between two general classes of concepts: *lower-level* concepts and *higher-level* concepts. Lower-level concepts have a strong syntactic dimension while the semantic dimension plays a secondary role. This does however not mean that such concepts lack in semantic meaning. Intertitles, for example are important structuring elements but at the same time contain a certain amount of semantic meaning since they provide context information about the film. Higher-level concepts (the second class) have a stronger semantic component and less syntactic importance. Connecting lines between concepts in Figure 1.1 represent relationships between different concepts.

This thesis presents novel methods for the retrieval of the identified concepts. The organization of this thesis is similar to the organization of the concepts in Figure 1.1. We first analyze the lower-level concepts to extract syntactic information from the films. Based on the extracted syntactic information we perform retrieval of higher-level concepts. The detailed organization of this thesis is provided in Section 1.4.

1.2 Contributions

The major goal of this thesis is the development of novel analysis techniques for the retrieval of concepts from archive films. Due to the low-quality of the archive film material, state-of-the-art techniques developed for high-quality video are not applicable to archive film. Additionally, the requirements and demands for retrieval in the context of archive film are different than those for video analysis. This thesis presents novel solutions to real world requirements posed by film scientists and archivists. We develop

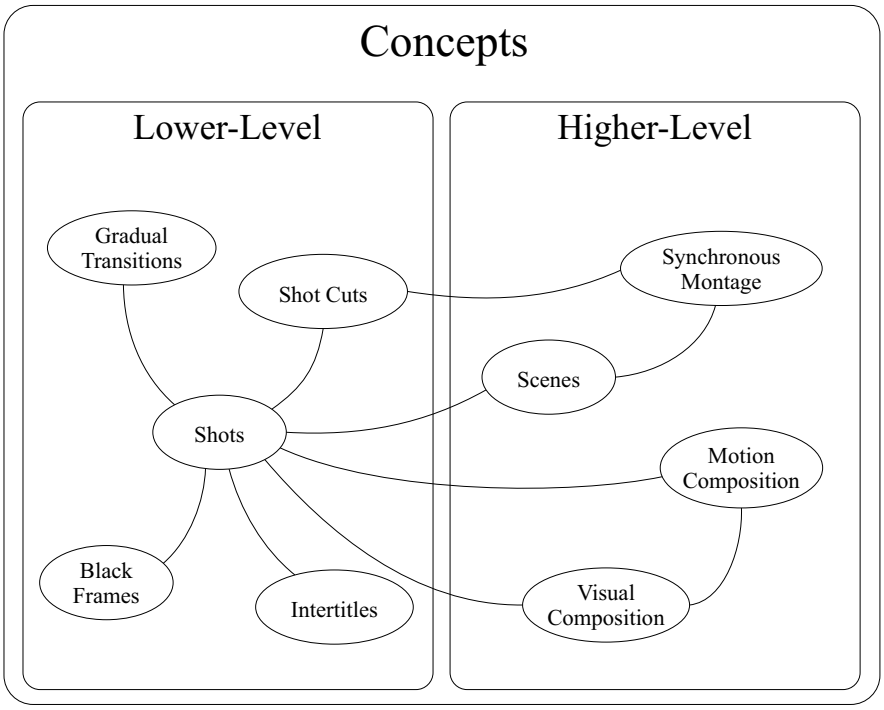


Figure 1.1: Investigated concepts and their relationships.

analysis techniques for the retrieval of different syntactic and semantic concepts from archive film.

We first perform two exemplary investigations on the retrieval of *black frames* and *intertitles* in Chapter 4. For black frames, we combine existing image features and show the complexity of the generally simple task in the context of archive film material. For intertitle detection, we propose novel content-based features that are capable of reliably separating intertitles from other types of content in a film.

In a next step, we investigate shot boundary detection in archive film. Since existing methods suboptimally perform on archive film material, we adapt and extend an existing approach to meet the requirements of archive film material. We first perform *shot cut detection* and introduce color-independent features that are robust to the artifacts in the archive film material (see Chapter 5). Additionally, we propose a novel feature fusion scheme for the effective combination of several features. The fusion scheme better exploits complementary information of different features than the original one. Additionally to shot cut detection we perform *gradual transition detection* in archive film material (see Chapter 6). We identify specific requirements of gradual

transition detection in archive film material. Based on the identified requirements, we extend an existing approach by more robust features and a more robust classifier. Our main contribution is a first systematic evaluation of gradual transition detection for archive and contemporary material.

The segmentation of a film into shots is the basis for the retrieval of higher-level concepts. We focus on the segmentation of *scenes* which represent the next higher structural layer of a film in Chapter 7. We propose a framework for multimodal scene segmentation that is purely based on visual and auditory similarities. The framework allows the combination and fusion of arbitrary visual and auditory features for scene segmentation. We propose a refinement step based on simple heuristics to improve the raw segmentation obtained from visual and auditory similarities. Additionally, we present a novel scheme for the aggregation of auditory features that enables the compact description of the audio content of a shot. We systematically evaluate the framework's components for archive and contemporary material.

An important stylistic device originally employed in early archive sound film is *synchronous montage*. Synchronous montage sequences are parts of scenes where the audio track and the visual cutting of a film are synchronized with each other for stylistic reasons. We propose a cross-modal method for the extraction of synchronous montage sequences from a film (see Chapter 8). The technique is based on automatically extracted shot cuts and auditory onsets. We propose a cross-modal correlation function that simulates human synchrony perception. Based on the detection of cross-modal correlations between the auditory and visual track, we introduce a tolerant segmentation scheme for the automatic extraction of entire sequences with synchronous montage.

Apart from the temporal composition of a film, reflected among others by shots, scenes and synchronous montage sequences, we investigate the retrieval of *motion composition* in a film (see Chapter 9). First, we introduce a novel clustering scheme for motion trajectories that is able to robustly segment highly fragmented and noisy motion fields into meaningful motion components. The clustering scheme allows trajectories to have different length and to break off frequently. The scheme is computationally efficient and allows the integration of arbitrary trajectory features and similarity measures and thus enables different types of clusterings and applications. Based on the extracted motion segments, we investigate two scenarios for the retrieval of motion composition. The basis for both scenarios is a simple and intuitive query, that enables

the user to abstractly sketch desired motion compositions. The first retrieval scenario represents a system for the retrieval of arbitrary motion compositions in a film based on a user-defined query. The second retrieval scenario addresses the search and retrieval of motion continuity (and motion discontinuity) between successive shots. For both retrieval scenarios we introduce tolerant matching schemes that compare the abstract motion queries with the previously extracted motion segments and retrieve relevant instances.

Additionally to motion composition, we investigate the retrieval of *visual composition* in film. We evaluate the applicability of low-level content-based features and similarity measures for this task in a user study (see Chapter 10). The user study reveals to which degree content-based features capture visual composition as it is understood by film scientists. Furthermore, we perform statistical data analysis of the features and propose a novel measure for expressiveness of features that is based on factor loadings obtained by Principal Component Analysis.

This thesis is a joint effort of Dalibor Mitrović and Matthias Zeppelzauer. The work and the resulting contributions originate from the close cooperation of the authors in a research project. The thesis has been composed together since the single investigations of both authors are highly related and partly build upon each other. A joint presentation allows a more comprehensive presentation of existing relationships and the performed research. Nevertheless the individual contributions of the authors can be clearly delineated and can be assessed separately from each other. In Table 1.1 we summarize each author's contribution to the chapters of this thesis. We distinguish between scientific contribution and the contribution to the composition of the thesis' text.

1.3 Resulting Publications

The work presented in this thesis has appeared in the following peer-reviewed publications:

- D. Mitrović, M. Zeppelzauer and H. Eidenberger. 2007. Analysis of the Data Quality of Audio Descriptions of Environmental Sounds. *Journal of Digital Information Management*, 5(2):48-55.

	Mitrović		Zeppelzauer	
Chapter	idea	text	idea	text
Detection of Black Frames and Intertitles	50%	70%	50%	30%
Detection of Shot Cuts	70%	70%	30%	30%
Detection of Gradual Transitions	20%	30%	80%	70%
Segmentation of Scenes	80%	60%	20%	40%
Extraction of Synchronous Montage Sequences	20%	20%	80%	80%
Retrieval of Motion Composition	30%	20%	70%	80%
Retrieval of Visual Composition - A User Study	80%	80%	20%	20%

Table 1.1: Contributions of the authors, separated by scientific contribution (idea) and textual contribution (text) for each chapter.

- M. Zeppelzauer, D. Mitrović and C. Breiteneder. 2008. Analysis of Historical Artistic Documentaries. *Proceedings of the 9th International Workshop on Image Analysis for Multimedia Interactive Services*, May 7-9, 2008, Klagenfurt, Austria. pp. 201-206.

- M. Zeppelzauer, M. Zaharieva, D. Mitrović and C. Breiteneder. 2010. A Novel Trajectory Clustering Approach for Motion Segmentation. *Proceedings of Multimedia Modeling Conference*, Jan 6-8, 2010, Chongqing, China, pp. 433-443.

- M. Zaharieva, M. Zeppelzauer, D. Mitrović and C. Breiteneder. 2010. Archive film comparison. *International Journal of Multimedia Data Engineering and Management*, 1(3):41-56.

- D. Mitrović, M. Zeppelzauer and C. Breiteneder. 2010. Features for Content-Based Audio Retrieval. *Advances of Computers*, vol. 78, editor: Marvin V. Zelkowitz, Academic Press, pp. 71-150.

- M. Seidl, M. Zeppelzauer and C. Breiteneder. 2010. A Study Of Gradual Transition Detection in Historic Film Material. *Proceedings of the ACM Multimedia 2010, Workshop - Electronic Heritage and Digital Art Preservation (eHeritage)*, October 25-29, 2010, Firenze, Italy, pp. 13-18.

- M. Zaharieva, D. Mitrović, M. Zeppelzauer and C. Breiteneder. 2011. Film Analysis of Archive Documentaries. *IEEE Multimedia*, 18(2):38-47, February, 2011.

- D. Mitrović, S. Hartlieb, M. Zeppelzauer and M. Zaharieva. 2010. Scene Segmentation in Artistic Archive Documentaries. *HCI in Work and Learning, Life and Leisure*, LNCS, vol. 6389, Springer, Berlin/Heidelberg, pp. 400-410.

- D. Mitrović, M. Zeppelzauer, M. Zaharieva and C. Breiteneder. 2011. Retrieval of Visual Composition in Film. *Proceedings of the 12th International Workshop on Image Analysis for Multimedia Interactive Services*, April 13-15, 2011, Delft, The Netherlands.

- M. Zeppelzauer, D. Mitrović and C. Breiteneder, 2011. Cross-Modal Analysis of Audio-Visual Film Montage. *Proceedings of 20th International Conference on Computer Communications and Networks*, July 31 - August 4, 2011, Maui, Hawaii.

- M. Seidl, M. Zeppelzauer, D. Mitrović and C. Breiteneder. 2011. Gradual Transition Detection in Historic Film Material – A Systematic Study. *To appear in ACM Journal on Computing and Cultural Heritage.*

- M. Zeppelzauer, M. Zaharieva, D. Mitrović and C. Breiteneder. 2011. Retrieval of Motion Composition in Film. *To appear in Digital Creativity*, vol. 22, issue 4.

1.4 Organization

The organization of this thesis principally follows the identified concepts shown in Figure 1.1. Prior to the work on the retrieval of the concepts, we review the principles of media retrieval in Chapter 2. We first present the basics of content-based retrieval and summarize important auditory and visual features. Next, we discuss similarity measurement in the context of retrieval and review classification techniques relevant to this thesis. Finally, we describe how retrieval systems are evaluated and how their performance is measured.

In Chapter 3 we present the investigated archive film material. We first overview the stylistic properties typical for the films. Next, we present the artifacts present in

the films and discuss their impact on the automatic analysis. Finally, we discuss the effects of automatic restoration of the films.

The first investigated concepts are black frames and intertitles. Both are structuring elements of a film, however at different scales. *Black frames* are provide structure at a small scale for the creation of rhythmic patterns in sequences. *Intertitles* structure a film at a larger scale, as they usually separate different broader topics. We address the detection and retrieval of black frames and intertitles in Chapter 4.

A central concept that gives a movie structure are *shots*. Successive shots are separated by shot boundaries which can either be gradual (*gradual transitions*) or abrupt (*shot cuts*). Both types of shot boundaries are analyzed in this thesis in Chapters 5 and 6.

The composition of a film has different aspects. One aspect is the temporal composition of a film. A film is usually temporally composed of several scenes. Scenes are made of several consecutive semantically related shots. Scenes have both, a semantic and a structural component. On the one hand, they represent high-level information about a film with an associated semantic meaning. On the other hand, scenes contribute to the structuring of a film, however at a coarser level than the single shots. The segmentation of films into semantically related scenes is presented in Chapter 7.

A concept related to scenes are *synchronous montage sequences*. Synchronous montage is an editing technique where the visual cutting (the shot cuts) and the audio track of a film are synchronized to increase tension and tempo in a sequence. Synchronous montage sequences often represent highlights in a scene. They have a strong semantic component and are highly characteristic for a film and a filmmaker's style. We focus on the extraction of synchronous montage sequences in Chapter 8.

Additionally to the temporal composition of a film by scenes and synchronous montage sequences, *visual composition* and *motion composition* are two concepts that play an important role in the investigated films. Visual composition refers to the spatial arrangement of objects and motion composition describes object motions, camera motions, and their interactions. The repeated use of particular compositions is a stylistic device employed frequently in the archive films. We investigate the retrieval of motion compositions in Chapter 9 and the retrieval of visual composition in Chapter 10.

Finally, we summarize the thesis and discuss open topics in the context of the identified concepts in Chapter 11.

Chapter 2

Principles of Media Retrieval

The methods presented in this thesis belong to the field of content-based media retrieval. In the following we briefly introduce content-based retrieval and discuss the different processing steps of a content-based retrieval task. In Sections 2.2 and 2.3 we survey auditory and visual content-based features which represent the basis of a content-based retrieval system. Next, we discuss the principles of similarity measurement in retrieval and review measures for distance and similarity comparison in Section 2.4. Section 2.5 targets pattern classification and presents the classifiers employed in this thesis. Finally, Section 2.6 focuses on the evaluation of retrieval systems, challenges of ground truth generation, and performance measures for retrieval systems.

2.1 Content-based Retrieval

In the last decades the number of available digital media has grown considerably. Additionally to textual information, image, audio, and video data have become ubiquitous due to the development of efficient compression and transmission techniques and the establishment of large (publicly accessible) databases in the Internet. The access to these large amounts of multimedia data is more difficult than to text documents. In text retrieval a search request is usually a set of terms that describe the desired information (textual query). A text retrieval system takes such a query as input and matches the query terms with the documents in a database [184, 226]. Thereby, the comparison of query terms and the content of a document is performed by exact matching (by testing the identity between two strings). In content-based media retrieval exact matching is

inappropriate. Given a media object as query (e.g. an image) the search for similar media objects in a database with exact matching would not succeed because similar objects (images) are usually not identical. Content-based retrieval requires *tolerant* matching for the comparison of media objects. For this purpose, traditional text-based information retrieval methods are not appropriate.

A straight-forward approach is to manually create textual metadata (annotations) for each multimedia object and to perform text retrieval on these annotations. A shortcoming of this approach is that the generation of annotations is a time-consuming and error-prone task. Additionally, annotations should be generated by domain experts to assure an adequate level of quality which makes the process expensive.

The general idea behind content-based retrieval is the extraction of information directly from the *raw* content of media objects (the pixels of images or the samples of an audio signal). The result of information extraction are numeric descriptions (features) that describe the media objects. Features, in the visual domain may be for example histograms that represent the color distribution of still and moving images. In the audio domain features may represent the pitch or the frequency distribution of a signal. Based on such numeric descriptions media objects can be compared to each other and assigned to a class of objects.

While a retrieval task is usually defined at a high level, e.g. "classify images as showing either indoor scenes or outdoor scenes", the numeric descriptions of the media objects (e.g. color histograms and texture descriptors) usually reside at a rather low level of abstraction. The gap that exists between the low-level numeric descriptions of the media content and the semantic meaning of the content to a human observer is referred to as the *semantic gap*. The semantic gap is omnipresent in all fields of content-based retrieval. For an audio retrieval system for example, a recording of Beethoven's symphony no. 9 is basically a series of numeric values (raw samples or feature vectors extracted from these samples, see Section 2.2). For a human however, the symphony is a sequence of notes with specific durations and pitches. A human may describe the symphony by even more abstract semantic concepts like musical entities (motifs, themes, movements), musical genres, and elicited emotions (excitement, euphoria).

On the technical side, the semantic gap is a direct consequence of the fact that content-based retrieval is an *inverse problem*. An inverse problem aims at the estimation of *model parameters* from *observed data* (measurements) [210]. In the case of

content-based retrieval, model parameters are terms, properties and concepts that may represent class labels (e.g. terms like "car" and "cat," properties like "male" and "female," and concepts like "outdoors" and "indoors"). The exact estimation of the model parameters is not possible in general, since the set of observed data is incomplete (not all possible data samples that make up a class or concept are available). This means that the model can only be approximated. Consequently, content-based retrieval is an ill-posed problem [78].

Humans bridge the semantic gap based on prior knowledge and (cultural) context. Due to the ill-posed nature of content-based retrieval, computers are usually not able to complete this task. Consequently, the goal in content-based retrieval is to *narrow* the semantic gap as far as possible. For this purpose, most approaches employ numeric models like the *vector space model* [185]. In the vector space model documents (originally text, later arbitrary media objects) are represented by numeric vectors, the *feature vectors*. The feature vectors should provide a compact and expressive representation of the media objects and capture information relevant for the given retrieval task. In text retrieval for example a feature vector may contain the probability of occurrence of representative terms that characterize the content documents. In the case of visual retrieval a feature vector may represent the color distribution of an image in the form of a color histogram. The representation of media objects by feature vectors additionally reduces the amount of data that has to be processed by several orders of magnitude. This is necessary, since the dimension of raw data, e.g. all pixels of an image is too high for direct processing and thus inadequate for retrieval. Furthermore, information such as shapes and texture is not directly apparent from the raw pixels. Hence, the extraction of feature vectors is an important preprocessing step to capture the required information from raw data and at the same time to neglect information that is not important for a given task.

Feature vectors of all documents in a repository always have the same dimension d and may be regarded as vectors in a d-dimensional vector space, usually $\mathbb{R}^d$, the *feature space*. Each feature vector denotes one position in this vector space. The basic assumption of the vector space model is that similar documents or media objects are represented by feature vectors that are spatially close in the vector space while dissimilar objects are spatially separated. Distances between feature vectors in the feature space are measured by different *metrics*, such as the Euclidean metric or the more

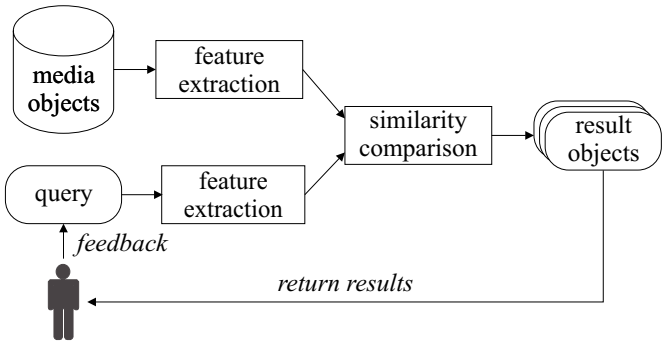


Figure 2.1: The workflow of a typical query-by-example retrieval system.

general Minkowski metric (see Section 2.4). Similarity judgements are obtained by mapping distances in the vector space to measures that approximate similarity.

Based on estimated similarities, the vector space model enables for example the comparison of a query object with objects in a database and allows for the retrieval of objects from a database that are similar to the query (similarity retrieval). The architecture of a typical query-based retrieval system is presented in Figure 2.1. The input to the system is a database with media objects (e.g. an image database) and a query object provided by the user (for example an image). In a first processing step features are extracted for each object in the database (*feature extraction*). The features for the objects in the database are usually stored in a feature database and are reused for all future search requests. The result of feature extraction are numerical descriptions that characterize particular aspects of the media objects (e.g. color distribution, texture information, faces, etc.). Next, the same features are extracted for the query object provided by the user.

After feature extraction, the feature vectors of the query and the media objects in the database are compared based on an adequate distance metric (*similarity comparison*). After similarity comparison the media objects that are most similar to the query object are returned to the user. This retrieval scenario is also called *query-by-example* [66].

Usually, not all returned media objects match the query and satisfy the user's expectations. Consequently, most retrieval systems offer the user the opportunity to give feedback for the returned media objects. The user may specify which of the returned objects meet her expectations and which do not (relevance feedback) [118]. This in-

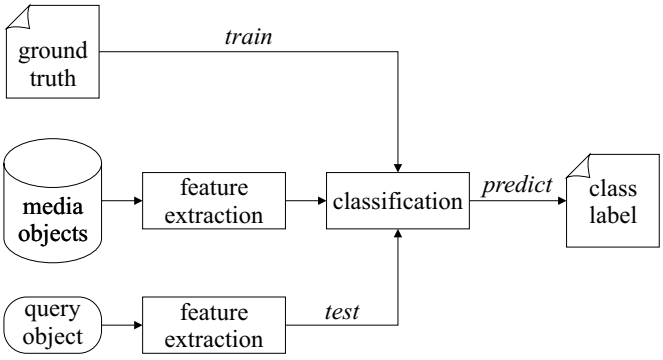
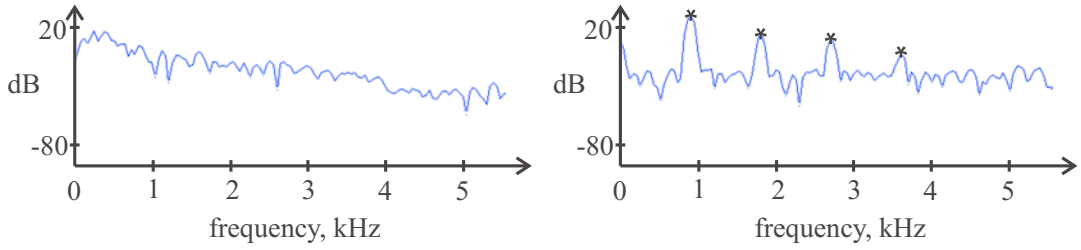


Figure 2.2: The workflow of a typical classification task.

formation may be used to iteratively refine the original query. Relevance feedback and iterative refinement enable the system to improve the quality of retrieval by incorporating the user's knowledge and intentions.

Additionally to query-based retrieval as described above, the vector space model supports the *clustering* and *classification* of media objects. Clustering groups similar objects represented as points in the feature space based on a particular distance metric and returns a representative for each cluster [79]. In classification each media object (represented as a point in the feature space) is assigned a distinct class label. A classifier first learns the properties of each class from training examples and later tries to predict the class labels of previously unseen media objects, see Section 2.5. The workflow of a typical classification task in content-based retrieval is illustrated in Figure 2.2.

The input is again a database of media objects (for example images, videos, or pieces of audio). Additionally, a *ground truth* (see Section 2.6) is available that contains for each object in the database a class label (e.g. "male" or "female" for a database of human face images). First, feature extraction is performed as in query-based retrieval for each media object in the database. Next, feature vectors are used to train the classifier. During training the classifier builds a *model* for each object class represented by the features (see Section 2.5 for details). Based on these models the classifier is able to later predict the class labels of previously unseen media objects (query objects). Training the classifier is usually an iterative process that includes repeated evaluations of the classifier by the ground truth. See Section 2.6 for details on the training and evaluation of classifiers.



(a) noise (b) harmonic

Figure 2.3: (a) The spectrum of a noise-like sound (thunder). (b) The spectrum of a harmonic sound (siren). The harmonic sound has peaks at multiples of the fundamental frequency (marked by asterisks), while the noise-like sound has a flat spectrum and consequently no pitch.

2.2 Auditory Features

In an audio retrieval system features should provide a compact and expressive representation of the underlying signals that captures the information that is most meaningful to a particular retrieval task. The type of auditory feature primarily depends on the required task and the characteristics of the given audio signals. Before we give an overview of different types of auditory features, we briefly present different characteristics of audio signals and basic audio attributes.

2.2.1 Attributes of Auditory Signals

Generally, we distinguish between tones and noise. Tones are characterized by the fact that they are "capable of exciting an auditory sensation having pitch" [9] while noise not necessarily has a pitch (see Figure 2.3(a)). Tones may be *pure tones* or *complex tones*. A pure tone is a sound wave where "the instantaneous sound pressure of which is a simple sinusoidal function in time" while a complex tone contains "sinusoidal components of different frequencies" [9].

Complex tones may be further distinguished into *harmonic complex tones* and *inharmonic complex tones*. Harmonic complex tones comprise of partials with frequencies at integer multiples of the fundamental frequency (so called harmonics, see Figure 2.3(b)). Inharmonic complex tones consist of partials whose frequencies significantly differ from integer multiples of the fundamental frequency.

From a psychoacoustic point of view, all types of audio signals may be described in terms of the following attributes: duration, loudness, pitch, and timbre.

Duration is the time between the start and the end of the audio signal of interest. The temporal extent of a sound may be divided into attack, decay, sustain, and release depending on the envelope of the sound. Not all sounds necessarily have all four components. Note that in certain cases silence (absence of audio signals) may be of interest as well.

Loudness is an auditory sensation mainly related to sound pressure level changes induced by the producing signal. Loudness is commonly defined as "that attribute of auditory sensation in terms of which sounds can be ordered on a scale extending from soft to loud" with the unit *sone* [9].

Pitch is defined by the American Standards Association as "that attribute of auditory sensation in terms of which sounds may be ordered on a scale extending from low to high" with the unit *mel* [9]. However, pitch has several meanings in literature. It is often used synonymously with the fundamental frequency. In speech processing pitch is linked to the glottis, the source in the source and filter model of speech production [172]. In psychoacoustics, pitch mainly relates to the frequency of a sound but also depends on duration, loudness, and timbre.

An attribute related to pitch is *pitch strength*. Pitch strength is the "subjective magnitude of the auditory sensation related to pitch" [9]. For example, a pure tone produces a stronger pitch sensation than high-pass noise [253]. Generally, the spectral shape determines the pitch strength. Sounds with line spectra and narrow-band noise evoke larger pitch strength than signals with broader spectral distributions.

Timbre is the most complex attribute of sounds. According to the ANSI standard timbre is "that attribute of auditory sensation which enables a listener to judge that two non-identical sounds, similarly presented and having the same loudness and pitch, are dissimilar." [9]. For example, timbre reflects the difference between hearing sensations evoked by different musical instruments playing the same musical note (e.g. piano and violin). In contrast to the above mentioned attributes, it has no single determining physical counterpart [4]. Due to the multidimensionality of timbre, objective measurements are difficult. Terasawa et al. propose a method to compare model representations of timbre with human perception [213].

Property	Values
Signal representation	linear-coded, lossily compressed
Domain	temporal, frequency, correlation, cepstral, modulation frequency
Temporal scale	intraframe, interframe, global
Semantic interpretation	perceptual, physical

Table 2.1: Formal properties of auditory features and their possible values.

Timbre is a high-dimensional audio attribute and is influenced by both stationary and non-stationary patterns. It takes the distribution of energy in the critical bands into account (e.g. the tonal or noise-like character of sound and its harmonics structure). Furthermore, timbre perception involves any aspect of sound that changes over time (changes of the spectral envelope and temporal characteristics, such as rhythmic patterns). Preceding and following sounds influence timbre as well.

Generally, auditory features describe aspects of the above mentioned audio attributes. For example there is a variety of features that aim at representing pitch and loudness. Other features capture particular aspects of timbre, such as sharpness, tonality and frequency modulations. See Sections 2.2.3 to 2.2.6 for an overview of auditory features organized by the aspect they represent.

2.2.2 Properties of Auditory Features

Additionally to the psychoacoustic attributes described in the previous section, auditory features may be characterized on a technical level by a number of (formal) properties. In Table 2.1, we summarize the most important properties.

A basic property of a feature is the *audio representation* it is specified for. We distinguish between two groups of features: features based on linear-coded signals and features that operate on lossily compressed (subband-coded) audio signals. Most feature extraction methods operate on linear-coded signals. However, there has been some research on lossily compressed domain auditory features, especially for MPEG audio encoded signals due to their wide distribution. Lossy audio compression transforms the signal into a frequency representation by employing psychoacoustic models which remove information from the signal that is not perceptible to human listeners (e.g. due to masking effects) [253]. Although lossy compression has different goals than feature

extraction, features may benefit from the psychoacoustically preprocessed signal representation, especially for tasks in which the human perception is modeled. Furthermore, compressed domain features may reduce computation time significantly if the source material is already compressed. Wang et al. provide a survey of compressed domain auditory features in [231]. In the following, we focus on features for linear-coded audio signals, since they often form the basis for lossily compressed domain auditory features.

Another property is the *domain* of an auditory feature. This is the representation a feature resides in after feature extraction. The domain allows for the interpretation of the feature data and provides information about the extraction process and the computational complexity. For example, a feature in *temporal* domain directly describes the waveform while a feature in *frequency* domain represents spectral characteristics of the signal. Another popular domain is the *cepstral* domain which can be considered as the spectrum of a spectrum and is used for example for the computation of MFCCs (see Section 2.2.6). Features in the *correlation* domain are derived from the auto-correlation of a signal and are frequently used for the estimation of fundamental frequency (see Section 2.2.5). Finally, the *modulation* domain reveals spectral variations over time in a signal which characterize for example musical rhythm.

Another property is the *temporal scale* of a feature. In general, audio is a non-stationary time-dependent signal. Hence, most feature extraction methods operate on short frames of audio where the signal is considered to be locally stationary (usually in the range of milliseconds). Each frame is processed separately which results in one feature vector for each frame. We call such features *intraframe* features because they operate on independent frames. Intraframe features are also called frame-level, short-time, and steady features [236]. A well known example for an intraframe feature are MFCCs which are frequently extracted for frames of 10-30 ms length.

In contrast, *interframe* features describe the temporal change of an audio signal. They operate on a larger temporal scale than intraframe features in order to capture the dynamics of a signal. In practice, interframe features are often computed from intraframe representations. Examples for interframe features are features that represent rhythm and modulation information. Interframe features are often called long-time features, global features, dynamic features, and clip-level features [221, 236].

In addition to interframe and intraframe features, there are *global* features. According to Peeters a global feature is computed for the entire audio signal. An example is

the attack duration of a sound. A global feature does not necessarily take the entire signal into account [162].

The *semantic interpretation* of a feature indicates whether or not the feature represents aspects of human perception. *Perceptual* features approximate semantic properties known by human listeners, e.g. pitch, loudness, rhythm, and harmonicity [250]. For this purpose, perceptual features usually incorporate psychoacoustic models [176]. Psychoacoustic models for example are filter banks that simulate the frequency resolution of the human auditory system [203]. Furthermore, models are integrated that take psychoacoustic properties into account, such as masking, specific loudness sensation, and equal-loudness contours [151, 253]. Investigations show that retrieval results often benefit from features that model psychoacoustic properties [73, 89, 176, 205].

Additionally to perceptual features, there are *physical* features. Physical features describe audio signals in terms of mathematical, statistical, and physical properties without emphasizing human perception in the first place (e.g. Fourier transform coefficients and the signal energy).

In the following, we give an overview of the most important auditory features in literature. We organize the features according to the auditory attribute they represent. A more comprehensive survey of auditory features can be found in [147].

2.2.3 Features related to Duration

Auditory features related to duration represent temporal characteristics of the waveform, for example particular points in time, such as the attack time of the sound. Popular features are *log attack time* and *temporal centroid*.

Log attack time. The log attack time characterizes the attack of a sound. According to the MPEG-7 standard, log attack time is the logarithm of the time it takes from the beginning of a sound signal to the point in time where the amplitude reaches a first significant maximum [101]. The attack characterizes the beginning of a sound, which can be either smooth or sudden. Log attack time is for example employed for classification of musical instruments by their onsets [91].

Temporal centroid. The temporal centroid (according to the MPEG-7 standard) is the time average over the envelope of a signal in seconds [101]. It is the point in time where most of the energy of the signal is located in average. The computation of temporal centroid is equivalent to that of spectral centroid (Section 2.2.6) in the frequency domain.

2.2.4 Features related to Loudness

We distinguish between two types of auditory features for loudness: physical loudness features and psychoacoustic loudness features. Physical loudness features coarsely estimate loudness from the energy in the signal (*short-time energy*, *volume*). Psychoacoustic loudness features aim at imitating the human sensation of loudness by taking into consideration psychoacoustic properties, such as critical bands, masking effects and equal loudness contours [253]. Advanced psychoacoustic loudness features are *specific loudness sensation* and *integral loudness*.

Short-time energy. Short-time energy is one of the most frequently used auditory features [37, 38, 202, 239]. Short-time energy is usually defined as the mean energy per frame (which actually is a measure for power) [250]. The same definition is used for the *MPEG-7 audio power descriptor* [101]. Additionally, there are definitions for short-time energy that take the *spectral* power into account [43, 134].

Volume. Volume is sometimes also called loudness, as in [238]. Volume is usually defined as the root-mean-square of the signal magnitude within a frame [130]. Consequently, volume is the square root of short-time energy. Both, volume and short-time energy reveal the magnitude variation over time. Volume is for example employed in silence detection and speech/music segmentation [104, 160].

Specific loudness sensation. Pampalk et al. propose a feature that approximates the specific loudness sensation per critical band of the human auditory system [159]. The authors first compute a Bark-scaled spectrogram [252] and then apply spectral masking and equal-loudness contours (expressed in phon) [253]. Finally, the spectrum is transformed to specific loudness sensation (in sone) which takes the logarithmic behavior of human loudness perception into account. Specific loudness sensation is

applied to audio retrieval for example in [152, 153]. We employ specific loudness for silence detection for scene segmentation in Chapter 7.

Integral loudness. The specific loudness sensation principally estimates the loudness of a single sine tone. A spectral integration of loudness over several frequencies enables the estimation of the loudness of more complex tones [253]. Pfeiffer proposes an approach to compute the integral loudness by summing up the loudness in different frequency bands [165]. The author empirically shows that the proposed method closely approximates the human sensation of loudness. The integral loudness feature is applied to foreground/background segmentation in [166].

2.2.5 Features related to Pitch

Similarly to loudness, we distinguish between physical pitch features and psychoacoustic pitch features. Physical pitch features mainly represent the fundamental frequency. The fundamental frequency is the lowest frequency of a harmonic series and is a coarse approximation of the psychoacoustic pitch. Fundamental frequency estimation employs a wide range of techniques, such as temporal autocorrelation, spectral, and cepstral methods and combinations of these techniques. An overview of techniques is given in [94].

Zero crossing rate (ZCR). One of the simplest and fastest methods for the coarse estimation of the fundamental frequency is the zero crossing rate, which is defined as the number of zero crossings in the temporal domain within one second. According to Kedem the ZCR is a measure for the dominant frequency in a signal [106]. ZCR is a popular feature for speech/music discrimination due to its simplicity [160, 187].

MPEG-7 audio fundamental frequency. The MPEG-7 standard proposes a more robust descriptor for the fundamental frequency which is defined as the first peak of the local normalized spectro-temporal autocorrelation function [42, 101]. Fundamental frequency is employed for example in [44, 222, 238].

Psychoacoustic pitch. Meddis and O'Mard propose a method to model human pitch perception in [141]. First, the authors apply a band-pass filter to the input signal to emphasize the frequencies relevant for pitch perception. Then the signal is decomposed with a gammatone filter bank that models the frequency selectivity of the cochlea. For each subband an inner hair-cell model transforms the instantaneous amplitudes into continuous firing probabilities. Next, a running autocorrelation function is computed from the firing probabilities in each subband. The resulting autocorrelation functions are summed across the channels in order to obtain a psychoacoustic pitch estimate.

2.2.6 Features related to Timbre

Timbre is the most complex attribute of audio. Consequently, a large number of features exist that describe different aspects of timbre. Since timbre comprises of stationary (short-time) and non-stationary (long-time) properties (as mentioned in Section 2.2.1) we distinguish between *stationary timbre features* and *non-stationary timbre features*.

Non-stationary timbre features

Non-stationary timbre features capture low-frequency modulations in audio signals. Modulation is a long-term signal variation of amplitude or frequency that is usually captured by a temporal (interframe) analysis of the spectrogram. Aspects of sound related to long-time modulations are rhythm and tempo which are especially important in music retrieval. In the domain of music information retrieval a number of modulation-based features have been introduced for the description of rhythmic structures, e.g. rhythm patterns [205] and pulse metric [189].

A widely used feature is the *beat spectrum* which represents the self-similarity of a signal for different time lags (similarly to autocorrelation) [68, 69]. In music, the peaks in the beat spectrum indicate strong beats with a specific repetition rate. Hence, this representation allows a description of the rhythm content of a piece of music. In non-musical signals, peaks in the beat spectrum indicate abrupt changes, such as speech onsets and the attacks of sudden noises). The beat spectrum is of particular interest for this thesis, as it is universal and can be applied to arbitrary time series. The concept of the beat spectrum can easily be applied to the detection of abrupt temporal changes in

a visual signal. We employ the beat spectrogram for the detection of shot boundaries in Chapters 5 and 6 as well as for cross-modal film analysis in Chapter 8.

Stationary timbre features

Features that capture stationary aspects of timbre are usually computed by a spectral (intraframe) analysis of short signal frames. In these short signal frames (usually 20-50 ms) the signal can be considered to be stationary. In a first step, the signal spectrum of a frame is computed by a Short Time Fourier Transform. Next, different information is extracted from the short-time spectrum. A first group of features extracts *particular properties* from the spectrum, such as the spectral centroid, bandwidth, flatness, and the harmonic structure. A second group of features aims at representing the *entire spectrum* in a compact and expressive way, for example by extracting the spectral shape. Features of the first group include:

Spectral centroid. The spectral centroid is defined as the center of gravity of the magnitude spectrum (first moment) [121, 222]. The spectral centroid determines the point in the spectrum where most of the energy is concentrated and is correlated with the dominant frequency of the signal. A definition of spectral centroid in logarithmic frequency can be found in [201]. Furthermore, spectral centroid may be computed separately for several frequency bands as in [174].

The spectral centroid is an approximation of the brightness of a sound. Brightness characterizes the spectral distribution of frequencies and describes whether a signal is dominated by low or high frequencies, respectively [253]. A sound becomes brighter as the high-frequency content becomes more dominant and the low-frequency content becomes less dominant.

Bandwidth. Bandwidth (often also referred to as spectral spread) is the magnitude-weighted average of the differences between the spectral components and the spectral centroid [238]. This means that the bandwidth is the second-order statistic of the spectrum. Tonal sounds usually have a low bandwidth (single peak in the spectrum) while noise-like sounds have high bandwidth. Bandwidth may be defined in the logarithmized spectrum as well as in the power spectrum [129, 134, 201]. Additionally, it may be computed within one or more subbands of the spectrum [5, 174]. Measures for

bandwidth are often combined with that of spectral centroid in literature since they represent complementary information [5, 134, 174].

Spectral rolloff point. The spectral rolloff point is the $N\%$ percentile of the power spectral distribution, where N is usually 85% or 95% [189]. The rolloff point is the frequency below which $N\%$ of the magnitude distribution is concentrated. It increases with the bandwidth of a signal. Spectral rolloff is used for example in music information retrieval [120, 152] and speech/music segmentation [189].

Spectral flatness. Spectral flatness estimates to which degree the frequencies in a spectrum are uniformly distributed [103]. The spectral flatness is the ratio of the geometric and the arithmetic mean of a subband in the power spectrum [174]. The same definition is used by the MPEG-7 standard for the *audio spectrum flatness* descriptor [101]. Spectral flatness may be further computed in decibel scale as in [77, 116]. Noise-like sounds have a higher flatness value (flat spectrum) while tonal sounds have lower flatness values. Spectral flatness is often used for audio fingerprinting [90, 116].

Harmonicity. Harmonicity is a property that distinguishes periodic signals (harmonic sounds) from non-periodic signals (inharmonic and noise-like sounds). Harmonics are frequencies at integer multiples of the fundamental frequency. Figure 2.3 presents the spectra of a noise-like (inharmonic) and a harmonic sound. Numerous features exist that describe the harmonic structure of a sound. An example is the *audio harmonicity* descriptor defined in the MPEG-7 standard. The audio harmonicity descriptor comprises two measures. The *harmonic ratio* is the ratio of the fundamental frequency's power to the total power in an audio frame [101, 110]. It is a measure for the degree of harmonicity in a signal. The computation of harmonic ratio is similar to that of MPEG-7 audio fundamental frequency, except for the used autocorrelation function [101]. The second measure of the audio harmonicity descriptor is the *upper limit of harmonicity*. The upper limit of harmonicity is the frequency beyond which the spectrum no longer has any significant harmonic structure. It may be regarded as the bandwidth of the harmonic components. The audio harmonicity descriptor is well-suited for the distinction of periodic (e.g. musical instruments, voiced speech) and non-periodic (e.g. noise, unvoiced speech) sounds.

The timbral features mentioned so far represent specific properties of the spectrum. The second group of stationary timbre features represents the shape of the entire short-time spectrum. Features from this group are the most frequently used features in audio retrieval. They comprise Mel-frequency cepstral coefficients (MFCC) and Bark-frequency cepstral coefficients (BFCC). Additionally linear predictive coding is employed to obtain the spectral shape.

Features that represent the spectral shape usually reside in the *cepstral* domain. The concept of the "cepstrum" has been originally introduced by Bogert et al. in [26] for the detection of echoes in seismic signals. In the audio domain, cepstral features have first been employed for speech analysis [32, 51, 158]. Cepstral features are frequency smoothed representations of the logarithmized magnitude spectrum. Furthermore, cepstral features allow for the application of the Euclidean metric as distance measure due to their orthogonal basis which facilitates similarity comparisons [51].

Bogert et al. define the cepstrum as the Fourier Transform (FT) of the logarithm (log) of the magnitude (mag) of the spectrum of the original signal [26].

$$signal \rightarrow FT \rightarrow mag \rightarrow log \rightarrow FT \rightarrow cepstrum$$

This processing chain is the basis for most cepstral features. However, in practice the computation slightly differs from this definition. For example, the second Fourier transform is often replaced by a discrete Cosine transform (DCT) due to its ability to better decorrelate the cepstral coefficients and due to its real-valued output.

Mel-frequency cepstral coefficients. MFCCs originate from automatic speech recognition but evolved into one of the standard techniques in most domains of audio retrieval. MFCCs have been successfully applied to timbre measurements by Terasawa et al. in [213]. MFCCs are computed from short-time spectra obtained by a short-time Fourier transform. First, the amplitude (magnitude) spectrum is logarithmized. This coarsely models the logarithmic perception of loudness in the human ear [204]. Next, the spectrum is transformed into Mel-scale by the application of a psychoacoustically scaled filter bank of (overlapping) triangle filters [203]. The Mel-scale gives more importance to low frequencies than for high frequencies, which corresponds to the frequency selectivity of the human ear. The output of the filter bank is a smoothed spectrum

with a lower number of frequency bands than the original spectrum. Finally, this Mel-scaled spectrum is input to a DCT which decorrelates the spectrum and allows a more compact representation. The resulting Cosine coefficients are referred to as cepstral coefficients. The following sequence illustrates the computation of MFCCs:

$$signal \rightarrow FT \rightarrow mag \rightarrow log \rightarrow Mel \rightarrow DCT \rightarrow MFCC\ coefficients$$

The first DCT coefficient represents the average power in the spectrum. The second coefficient approximates the broad shape of the spectrum and is related to the spectral centroid. The higher-order coefficients represent finer spectral details. In practice, the first 8-20 MFCC coefficients are used to represent the shape of the spectrum.

A variation of MFCCs are *Bark-frequency cepstral coefficients*, which differ from MFCC only in the applied psychoacoustic scale. Instead of the Mel-scale, the Bark-scale is employed [252]. The computation is as follows:

$$signal \rightarrow FT \rightarrow mag \rightarrow log \rightarrow Bark \rightarrow DCT \rightarrow BFCC\ coefficients$$

Cepstral coefficients based on the Mel-scale are the most popular variant used today, even if there is no theoretical reason that the Mel-scale is superior to other scales.

Autoregression-based features. Another group of timbral features for the description of the spectral envelope are autoregression-based features. In Autoregression analysis a linear predictor estimates the value of each sample of a signal by a linear combination of previous samples.

Linear predictive coding (LPC) estimates the coefficients of a filter of order p (the autoregressive filter) that predicts the value of the input signal at time x from the past p samples. The filter response of the estimated autoregressive filter is obtained by computing the Fourier transform of the filter coefficients. The resulting magnitude spectrum (the *linear prediction spectrum*) is an approximation of the spectral envelope of the signal. The extraction process is as follows:

$$signal \rightarrow LPC \rightarrow FT \rightarrow mag \rightarrow linear\ prediction\ spectrum$$

In speech recognition the linear prediction spectrum is used to estimate basic parameters of a speech signal, such as formant frequencies and the vocal tract transfer function [173]. The linear prediction spectrum is employed in other domains, such as audio segmentation and general purpose audio retrieval, as well [107, 108, 127].

Alternatively, the cepstral representation of the linear prediction spectrum (LPCCs) is frequently used due to their higher retrieval efficiency [239]. LPCCs are the inverse Fourier transform (iFT) of the logarithmized magnitude frequency response of the autoregressive filter. They are an alternative representation for linear prediction spectrum and thus capture equivalent information. The following processing chain illustrates the computation of LPCCs:

$$signal \rightarrow LPC \rightarrow FT \rightarrow mag \rightarrow log \rightarrow iFT \rightarrow LPCC\ coefficients$$

Alternatively, LPCCs may be directly derived from the linear prediction coefficients with a recursion formula [13].

LPCCs have shown to perform better than linear prediction coefficients, e.g. in automatic speech recognition, since they are a more compact and robust representation of the spectral envelope [1]. Furthermore, they allow for the application of the Euclidean distance metric due to their orthogonal basis.

2.3 Visual Features

Visual features should provide a compact and expressive representation of visual signals. Similarly to Section 2.2, we first present characteristics of visual signals and investigate properties of features prior to discussing visual features.

2.3.1 Attributes of Visual Signals

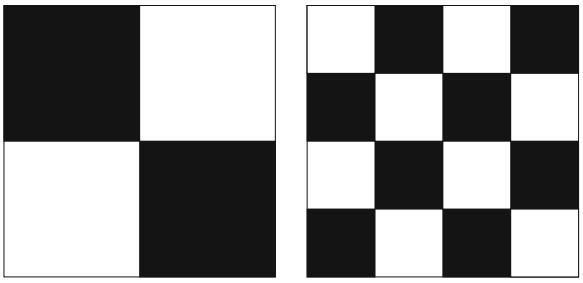
The visual signals considered in this thesis are digitized moving and still images recorded with analog cameras. The images can be seen as the results of perspective projections of physical, mostly opaque, objects embedded in a, mostly transparent, medium onto two-dimensional (image) planes [96]. These images have a number of attributes that may be used to describe the depicted real world objects. In the context of content-based retrieval, the attributes intensity, color, texture, shape, salient points and motion have gained importance. These attributes are suitable to serve as an organizing principle for the discussion of content-based features in Sections 2.3.3-2.3.7.

Image *intensity* corresponds to the amount of light that is reflected by the depicted objects in the direction of the observer and captured by the camera. The intensity is high (the image is bright) if much light is reflected and low (the image is dark) if little

light is reflected. Image intensity is also known as *gray value*, *gray tone*, *image value*, *luminance*, and *brightness* [85].

Color is interrelated to intensity, additionally to the amount of reflected light color takes its wavelengths into account. Color values are composed of intensities recorded separately in three frequency bands representing the biologically motivated primary colors red, green and blue [85]. The three primary colors span the three-dimensional RGB color space. Each color represents a distinct point in this space. The RGB space does not explicitly provide information such as brightness and saturation which impedes the interpretation of colors specified in the RGB space. A more intuitive interpretation of colors is provided by the HSV color space. The three axes represent hue, saturation and value, where value represents the brightness component. RGB and HSV are perceptually non-uniform color spaces. This means, that perceptually similar colors are not necessarily located near to each other in the color space. A perceptually uniform space is the LAB color space [45, 199]. The LAB color space has been defined in a way that the perceptual similarity between two colors is approximated by their Euclidean distance in the space. This (relative) perceptual uniformity of the LAB space facilitates similarity comparisons in retrieval. A comparison of different color spaces in content-based image retrieval is provided in [140]. Note that transformations between different color spaces is possible if the retrieval task requires this.

Texture is related to the structure of the objects' surface and its influence on the image intensity. Hawkins writes about texture [88, 169]: "The notion of texture appears to depend upon three ingredients: (1) some local 'order' is repeated over a region which is large in comparison to the order's size, (2) the order consists in the nonrandom arrangement of elementary parts and (3) the parts are roughly uniform entities having approximately the same dimensions everywhere within the textured region." Texture is often described in terms of intensity, density, dimensions of uniformity, coarseness, roughness, regularity, and directionality [85]. Figure 2.4 shows two images of checkerboard-like areas as examples of textured surfaces. Although, both images have the same gray value distribution (50% black and 50% white) they differ in texture and hence in appearance. The main difference between the two textures is that, the elementary parts (tiles) of Figure 2.4(a) are larger than the ones in Figure 2.4(b) making the texture of Figure 2.4(a) coarser.



(a) (b)

Figure 2.4: Examples of textures of two checkerboard-like areas. Although, both images have the same intensity distribution, the structure of their surfaces makes them easily distinguishable.

Shape relates to the depicted objects and regions in the scene. Features are used to describe their *boundary*, *covered area*, and *topology* in the image. The boundary can be described in terms of local extrema in curvature, inflection points, convexity, and concavity. The area covered by an object can be described in terms of eccentricity, surface area, bounding box, and roundness. Topology refers to the number of holes of an object and the number of disconnected parts that make up an object. The description of shape requires a segmentation of objects or interest regions in an image.

Salient points refer to particular points in images that are not influenced by geometric and radiometric distortions and that are distinct in their spatial neighborhood [85, 191]. This distinctiveness is often based on the attributes intensity, color, and texture. Ideally, salient points describe the visually most important points, edge elements (e.g. corners, T-junctions), and (small) patches of an image [218].

Motion is an attribute of moving images, it may refer to the motion of depicted objects, the motion of the camera, and camera operations like zoom-in and zoom-out. Motion may be described in terms of direction, magnitude, and acceleration.

2.3.2 Properties of Visual Features

Similarly to auditory features, visual features reside in different *domains* after extraction. For visual features the spatial (or image) domain is represented by the Cartesian coordinate system of the raw image. This domain is the native domain for images, similarly to the time domain of auditory features. From the native domain we obtain the

frequency domain by the application of (two-dimensional) frequency transforms, such as discrete Fourier transform and discrete Cosine transform. Frequency transforms decompose the input image into its spatial frequencies and yield two-dimensional frequency representations which are a well-suited basis for the extraction of content-based features.

We can distinguish between features by their *spatial structure*. There are two groups of features. The first group describes the image in terms of salient points and the second group of features accumulates information over the entire image. Features based on salient points are often extracted in two passes. In the first pass the salient points are identified and in the second pass descriptions for the point based on its neighborhood are computed. These features represent local information and are also referred to as *local features* and *local descriptors*. In contrast to local features, accumulated features describe the entire image. Accumulated features consist of *global* features and *block-based* features. The global features usually do not contain spatial information and accumulate information globally about the entire image, for example the distribution of gray values in an image (intensity histogram, see Section 2.3.3) is a global feature. Earlier, in Figure 2.4 we saw two images with the same distribution of gray values but different appearance. In order to distinguish between the two images using accumulated features some kind of spatial information is necessary. Such spatial information is provided by block-based features. Block-based features are computed for blocks of fixed size of the image and the resulting values are concatenated. To stay with the above example, assume that we divide the images in 2×2 blocks and then analyze the intensity distribution inside each block: For Figure 2.4(a) we obtain two distributions with 100% white and two distributions with 100% black pixels while for the image in Figure 2.4(a) we obtain four distributions with 50% white and 50% black pixels.

Invariance is an often desired property of features that is related to their sensitivity towards distortions and transformations of the underlying visual signals. The desired invariance depends on the retrieval task. For example, the description of object shapes requires rotation-invariant features to enable the matching of differently oriented objects. However, rotation invariance may be undesired and impeding for the extraction of the horizon in natural images.

2.3.3 Features related to Intensity

Intensity features represent the distribution of intensity (brightness) in an image. Features either capture the *occurrence frequency* of intensity values (intensity histograms) or the *spatial distribution* of intensity (global DCT coefficients) or both (block-based histograms and block-based DCT coefficients). Intensity features are structurally similar to color features (see Section 2.3.4) with the difference that only the intensity channel is processed instead of several color channels. Intensity features are frequently used in this thesis for image representation since archive film material is monochromatic and color features cannot be applied.

Intensity histograms A standard feature that characterizes the intensity distribution in an image is the intensity histogram (often also referred to as luminance histogram). The intensity histogram principally represents the occurrence frequency of each possible gray value. In practice, the gray values are aggregated into a smaller number of histogram bins. By using a smaller number of bins the robustness of the histogram to illumination changes increases. The global intensity histogram completely neglects spatial information and is thus invariant to rotation. If the histogram is additionally normalized (divided by the number of pixels in the input image), it becomes also invariant to scaling. Alternatively, the intensity histogram can be computed separately for a number of image blocks in order to capture spatial information. For this purpose, the histograms of each image block are concatenated into one vector. The result is a block-based intensity histogram. Global and block-based intensity histograms are used for example for frame-to-frame comparisons in shot cut detection [28, 211]. We employ intensity histograms for example in Chapter 6 for gradual transition detection and for intertitle detection in Chapter 4.

DCT coefficients A compact description of the spatial intensity distribution in an image can be obtained from the coefficients of the discrete Cosine transform (DCT) [7]. The discrete Cosine transform decomposes an image into its horizontal, vertical and diagonal spatial frequencies. The first 64 basis functions of the two-dimensional DCT are shown in Figure 2.5. The top left component is the zero-frequency DC coefficient. It represents the mean gray value of the image. The remaining entries are the AC components which represent horizontal frequencies (first row), vertical frequencies (first

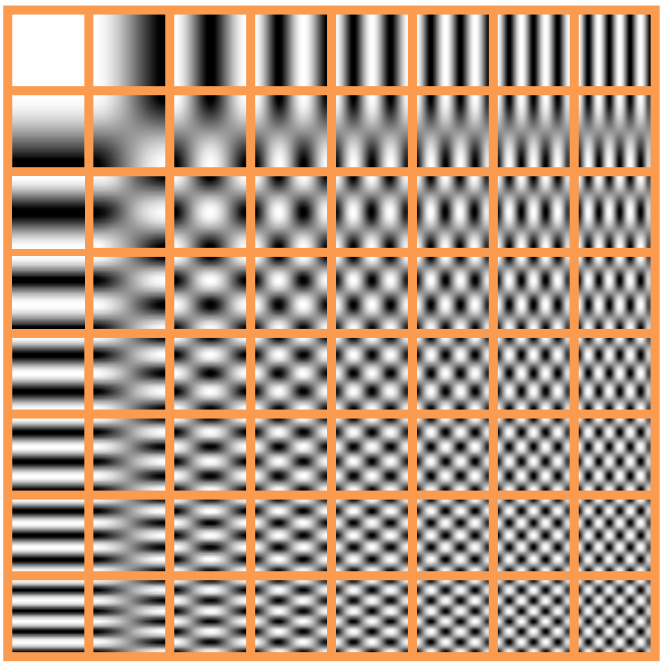


Figure 2.5: The first 8×8 basis functions of the two dimensional DCT. Bright pixels represent high amplitudes and dark pixels low amplitudes.

column), and diagonal components (all other rows and columns). The spatial frequency increases towards the right and the bottom of the figure. For each component the corresponding DCT coefficient expresses to which degree the underlying image matches the spatial frequency of the respective component. Most information about the image is concentrated in the low-frequency DCT coefficients. In practice, a few DCT coefficients are sufficient to represent the coarse spatial intensity distribution of an image.

DCT coefficients can be computed globally for an image or for image blocks. The global DCT coefficients represent the spatial distribution of intensity values over the entire image. Block-based DCT coefficients are extracted for individual image blocks and represent the intensity distribution in each image block separately and thus capture more spatial information than global DCT coefficients. We employ global and block-based DCT coefficients for shot cut detection and gradual transition detection in Chapters 5 and 6.

2.3.4 Features related to Color

A number of features exist for the description of color in an image. Color features represent different aspects of color, such as the *representative colors* of an image (MPEG-7 dominant color), the global *occurrence frequency* of colors (color histogram), the *spatial distribution* of colors (MPEG-7 color layout), and the *spatial coherency* of colors (MPEG-7 color structure). An additional aspect is the *co-occurrence* between different colors in an image (color correlogram).

MPEG-7 dominant color. The dominant color descriptor extracts the representative colors of an image [101]. In a first step, an image is clustered into a small number of representative colors. The cluster centroids represent the dominant colors. The descriptor contains the dominant colors together with the percentage of pixels that reside in the corresponding cluster. Additionally, the descriptor stores the variance of the colors in the corresponding cluster for each dominant color. Finally, a coherency measure is computed, that measures the spatial coherency of the dominant colors. The dominant color descriptors of two images may contain different (numbers of) dominant colors, and thus cannot be compared directly. An appropriate distance measure is proposed in [52]. The measure compares all dominant colors from one image with all dominant colors from another image, computes their Euclidean distances and sums up all distances that are below a threshold.

Color histograms. Color histograms are among the most widely used color features in content-based retrieval. They represent the occurrence frequency (distribution) of colors in an image. For the computation of a color histogram we (uniformly or non-uniformly) subsample the axes of the underlying color space. This subsampling yields a partition of the color space into subspaces where each subspace corresponds to one histogram bin. All colors that reside in one of the subspaces are considered similar. A bin of the color histogram contains the number of pixels whose colors reside in the corresponding subspace. Similarly to the intensity histogram in Section 2.3.3, the global color histogram is invariant to rotation. If the histogram is normalized it additionally becomes invariant to scaling. Global histograms do not capture the location of the colors, only their occurrence frequencies. An example of a global color histogram is the MPEG-7 scalable color descriptor [101, 138]. The scalable color descriptor is based on

the HSV color space and employs a uniform quantization of the space into 256 bins (subspaces). The HSV histogram is quantized and encoded by a Haar transform. The transform makes the descriptor scalable which enables the comparison of descriptors with different numbers of bins. Alternatively, each global histogram can be computed for blocks of an image separately. The individual histograms are then concatenated into one compound histogram. The result is a block-based histogram that captures spatial information about the image as well.

MPEG-7 color layout. An orthogonal feature to histograms is the MPEG-7 color layout descriptor [101]. While histograms principally represent distributional (but no spatial) information the color layout descriptor captures solely spatial (but no distributional) information. The color layout descriptor is computed by dividing an image into 8×8 blocks. For each block the average color is computed. The resulting 64 values (for each color channel) are transformed into frequency space by a two-dimensional DCT. The low-frequency DCT coefficients for each color channel are concatenated and together form the final feature vector.

MPEG-7 color structure. The MPEG-7 color structure descriptor is a combination of a histogram and a spatial color descriptor [101]. In contrast to a histogram, where each bin represents the occurrence frequency of a color, the bins of the color structure descriptor represent the degree of *spatial coherency* of the pixels with a corresponding color. The spatial coherency is high if all pixels of a color appear in one coherent region while the spatial coherency is low if the pixels of a color are evenly distributed across the image. The spatial coherency for a color is estimated by moving a sliding window over the image. The coherency of the color is inversely proportional to the number of locations where the sliding window at least contains one pixel with the current color [138]. The color structure descriptor is able to distinguish two images with identical occurrence frequencies of colors (identical color histograms) but different spatial distributions of colors.

Color correlogram. Additionally to the spatial distribution and coherency of colors, a further spatial aspect is the co-occurrence of *different* colors. The color correlogram is a feature that captures the spatial relationships between different colors [99]. For the

computation the color space is first subsampled into a lower number of colors. Next, all colors are compared in a pair-wise manner with each other. The color correlogram of two arbitrary colors i and j represents the probability that a pixel with color j can be found at distance d from a pixel with color i. The color correlogram is computed for all pairs of colors and contains probabilities for different distances d. Huang et al. present efficient methods for the computation of color correlograms and show that color correlograms outperform histogram-based techniques [99].

2.3.5 Features related to Shape

Shape features describe the regions covered and contours of pre-segmented objects and interest regions in an image. A broad survey and comparison of shape features is provided in [142]. According to the authors there are two basic types of shape features: *boundary-based features* which operate on the outline (contour) and neglect the interior of the objects and *region-based* approaches which capture properties of the region including properties of the interior such as holes. We present representative features for both types of features: the MPEG-7 region shape descriptor captures structural properties of an object region and the MPEG-7 contour descriptor robustly represents an object's boundary.

MPEG-7 region shape. The MPEG-7 region shape descriptor is able to represent arbitrary object regions, containing holes and disconnected parts [101]. The descriptor comprises the coefficients of the Angular Radial Transform (ART). The ART is a 2-D complex transform which decomposes the input image by angular and radial basis functions. The region shape descriptor employs twelve angular and three radial functions to describe a region, see Figure 2.6. The angular frequency of the basis functions increases from left to right in Figure 2.6. The radial frequency increases from top to bottom. The region shape descriptor contains the coefficients of all radial basis functions in Figure 2.6 except for the top left function which represents the zero-frequency component. The region-based shape description is invariant to rotation and robust to scaling [179]. A comparative study of different region descriptors has shown, that the MPEG-7 region shape descriptor outperforms other techniques in most experiments [248].

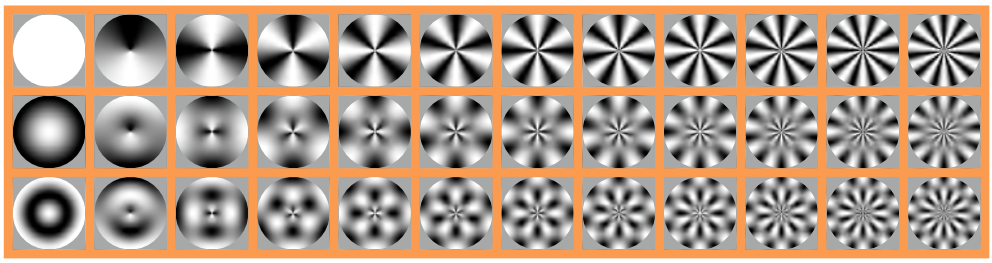


Figure 2.6: The basis functions of the Angular Radial Transform employed for the computation of the MPEG-7 region shape descriptor. Bright pixels represent high amplitudes and dark pixels low amplitudes.

(a) (b) (c)

Figure 2.7: Three objects with similar region properties but different contours.

MPEG-7 region contour. While the region shape descriptor captures properties of the area covered by an object, the MPEG-7 region contour descriptor describes the outline of an object or region of interest. Contour and region descriptors are complementary to each other. Contour descriptors are able to distinguish between objects with similar spatial pixel distributions (region shapes), see for example Figure 2.7.

A contour descriptor should be invariant to a large number of transformations such as affine transformations, non-uniform scaling, and perspective transformations in order to allow robust similarity comparisons between objects. The MPEG-7 region shape descriptor achieves invariance to a number of transformations by the underlying Curvature Scale-Space (CSS) representation [25, 148]. The CSS representation is obtained by first sampling the object boundary with a fixed number of equidistant points. Next, the peaks of the second derivative (curvature) of the sampled boundary are computed. The peaks in the second derivative indicate salient points of the contour. In the CSS representation the boundary is represented at differently smoothed scales (the smoother the boundary the more convex it becomes). For each scale the peaks in the second derivative are extracted and sorted by decreasing level of scale. The scale represents the salience of the corresponding point in the contour (peaks in a highly smoothed contour

are more salient than peaks in the original contour). The position of the peaks is represented relative to the highest peak to obtain invariance to rotation [101]. Additionally, the peaks are normalized to obtain invariance to uniform scaling [148]. The MPEG-7 contour shape descriptor contains the ordered series of normalized peaks. Additionally, the descriptor contains two global contour parameters (eccentricity and circularity) of the original contour and the smoothest employed contour [101]. MPEG-7 contour shape is a compact descriptor for closed object boundaries. The dimension of the descriptor varies with the complexity of the underlying boundary. For convex boundaries no peaks are extracted at all. Due to the varying dimension of the descriptor special matching schemes are required. A tolerant matching scheme for the contour descriptor is described in [248].

2.3.6 Features related to Texture

Texture has multiple properties, such as coarseness (from coarse to fine), contrast (from high to low), directionality (from directional to non-directional), and roughness (from rough to smooth) [209]. Content-based features for texture usually describe one particular property of texture, e.g. the individual Tamura features. Other features represent texture in a more holistic way, such as the MPEG-7 edge histogram and MPEG-7 homogeneous texture. The edge histogram captures the distribution of differently oriented edges in an image. Homogeneous texture quantitatively represents the intensity variation of a texture along different scales and directions. Since texture is independent of color, texture features are extracted from the image intensity [199].

MPEG-7 edge histogram. Edges are building blocks of texture. The MPEG-7 edge histogram represents the distribution of edge directions in an image [101]. For the computation of the edge histogram five types of edges are distinguished: horizontal edges, vertical edges, diagonal edges with a slope of 45°, diagonal edges with a slope of 135° and non-directional edges. The edge histogram is computed by dividing the input image into 16 non-overlapping sub-images. In each sub-image a separate edge histogram is computed. For this purpose, a sub-image is divided into a fixed number of image blocks. Each image block is matched with templates of the five edge types. If the score for the best matching template exceeds a certain threshold, the image block

is assigned to the corresponding edge type and contributes to the histogram of the current sub-image. The final edge histogram is the concatenation of the edge histograms of all 16 sub-images. The bins of the histogram are normalized and non-linearly quantized according to the MPEG-7 standard [101]. Alternatively, to this block-based edge histogram (based on sub-images) a *global* edge histogram for an entire image can be computed analogously [138]. Edge histograms robustly represent salient information in an image and have been successfully applied in several investigations presented in this thesis. We employ global and block-based edge histograms for intertitle detection (Chapter 4), shot boundary detection (Chapters 5 and 6), scene segmentation (Chapter 7), and visual composition retrieval (Chapter 10).

MPEG-7 homogeneous texture. While the MPEG-7 edge histogram is extracted from distinct local information in an image, namely edges, the MPEG-7 homogeneous texture descriptor takes all pixels into account and provides a global quantitative representation of texture [101]. Homogeneous texture is extracted in frequency domain from filter responses of differently scaled and oriented two-dimensional Gabor functions. First, the frequency domain is partitioned into 6 frequency channels along the angular direction (each channel captures 30°). Next, the 6 frequency channels are partitioned along the radial dimension into 5 octave scaled channels. The result are 30 frequency channels with different scales and orientations. For each frequency channel a two-dimensional Gabor function with the corresponding scale and orientation is generated and applied to filter the image. The resulting filter response (energy) is summed and scaled logarithmically. Additionally, the logarithmically scaled standard deviation of the energy is computed. The homogeneous texture descriptor contains the summed energy and the energy deviation for each frequency channel. Furthermore, the mean and standard deviation of the image intensity are added. The homogeneous texture descriptor represents global directional and spatial information and thus is related to directionality and coarseness of texture. The descriptor has successfully been applied to texture retrieval in large databases [138]. We apply the descriptor for visual composition retrieval in Chapter 10.

Tamura features In contrast to the above features, Tamura features represent distinct properties of a texture [209]. According to the authors especially three properties,

namely coarseness, contrast, and directionality are of special relevance to human perception.

Coarseness is related to the size of the texture elements and can be captured by a multi-scale analysis [182]. In a preprocessing step, the average of differently sized square neighborhoods is computed at each pixel. Next, for each pixel the difference between the averages of a neighborhood positioned to the left and a neighborhood positioned to the right of the pixel is computed. These differences are computed for differently sized neighborhoods to detect differently sized (coarse) structures. The same is performed in the vertical direction. The size that yields the maximum difference along the horizontal and vertical direction is assigned to the pixel. If several sizes yield a maximum, the largest size is selected. The final coarseness measure is the mean of the maximum sizes over all pixels of an image.

Contrast is related to several factors, such as the range of gray values, the ratio of black and white areas, the sharpness of edges, and the period of repeating patterns [209]. Tamura et al. define a contrast measure as the quotient of the standard deviation and the standardized kurtosis (fourth moment) of the intensity distribution of an image. The standard deviation reflects the dynamic range of the gray values and kurtosis represents the pointedness of the gray value distribution. The standardized kurtosis is non-linearly scaled by taking the power of $1/4$. The contrast feature with this exponent has shown to approximate human perception best in the experiments of [209].

Directionality measures to which degree a texture has a dominant direction. For this purpose Tamura et al. compute a histogram of gradient directions that incorporates all gradients from the input image with a significant magnitude (magnitude above a threshold). Sharp peaks in the histogram indicate dominant directions in the texture. The sharpness of a peak is expressed by the variance of the histogram in the neighborhood of a peak. Additionally, this variance is scaled by the peak height. The sharpness of all peaks is accumulated to obtain a feature for directionality.

2.3.7 Features related to Salient Points

In the context of salient points, we distinguish between *detectors* and *descriptors*. While detectors identify the salient points and give their position, the descriptors are needed for retrieval and matching. Descriptors are usually centered around the salient point and compactly represent the salient point's neighborhood.

Harris corner detector. The Harris corner detector is a widespread detection method for salient points. Devised by Harris and Stephens [86] in 1988, the Harris corner detector is still considered a state-of-the-art technique [74]. The detector is based on the gradient distribution in the neighborhood of a pixel. The Hessian matrix is built from the smoothed derivatives for the neighborhood of each pixel in the original image. Corners are those points in the image where the visual signal strongly changes in vertical and horizontal direction. Corners are detected based on the *cornerness* measure that reflects the amount of change in both directions. The cornerness can be expressed using the eigenvalues of the Hessian matrix. However, Harris proposed to compute the cornerness from the determinant and the trace of the Hessian in order to reduce the computation time. This is a valid approach, because the determinant is the product and the trace is the sum of the eigenvalues. Local maxima of the cornerness measure are identified using non-maximum suppression. These local maxima are the detected salient points. The points are invariant to rotation, translation and (to a certain degree) radiometric distortions. However, they are sensitive to scale changes [133]. Note that the salient points detected by Harris's technique are not only true corner points but include for example T-junctions and points with high curvature [218].

Maximally stable extremal regions. Maximally stable extremal regions (MSER) is a region detector related to regions that are stable with respect to thresholding [139]. Given an intensity image we create binary versions of the image by thresholding with successively increasing intensity thresholds [208]. We arrive at an image sequence that starts with an entirely white image and transforms into an entirely black image. First, some black pixels appear which represent local intensity minima. Next, the intensity minima grow, and for some threshold values begin to merge. Eventually, all intensity minima are merged and we arrive at the entirely black image. Regions that remain stable (in shape) for a large number of threshold values correspond to maximal regions. Analogously, the minimal regions are identified. Tuytelaars and Mikolajczyk list the following four properties of MSER: (i) MSER are invariant to monotonic changes of the image's intensity values. (ii) MSER are preserved under a number of geometric changes. (iii) The absolute number of pixels in the image is the maximum possible number of MSER. (iv) MSER are computationally cheap. However, they are sensitive

to image blur. MSER's properties make them suitable for the recognition of specific objects while they do not perform that well for object class recognition [218].

Scale invariant feature transform. Scale invariant feature transform (SIFT) is a widely used algorithm for the detection and description of salient points (also referred to as SIFT keypoints). It has been devised by Lowe [133] for image matching in various application domains such as object recognition, motion tracking and segmentation, and stereo correspondence matching. SIFT keypoints are extracted for different scales, for each scale the original image is iteratively smoothed with Gaussian filters with increasing σ. The result is a set of differently smoothed images. Next, difference-of-Gaussian images are computed by subtraction of adjacent images of the set. Then, local extrema are identified by comparing each pixel with its neighbors in the current and the adjacent difference-of-Gaussian images and selecting only the ones that are larger (for maxima) and smaller (for minima) than all the neighbors. Finally, local extrema that have low contrast or are located at edges are removed and location, scale, and orientation are assigned to each salient point. The SIFT algorithm identifies large numbers of salient points at different scales which is beneficial for the identification of comparatively small and occluded objects in cluttered images. Salient points with large scales introduce robustness to image noise and blur. Additionally, SIFT keypoints are robust to affine transformations (rotation, scaling, translation) and illumination changes of the original image.

Additionally to the detector, Lowe proposes a descriptor for the salient points. The descriptor is based on gradient magnitudes and orientations that are sampled in the neighborhood of the keypoint. The neighborhood is aligned to the orientation of the keypoint to obtain rotation invariance of the descriptor. Similarly, the neighborhood's size is aligned with the keypoint's scale to obtain invariance to scaling. The descriptor contains 8-bin orientation histograms for different blocks in the neighborhood of the salient point. These histograms form the feature vector (descriptor) which is made more robust against illumination changes through normalization to unit length.

Intensity-domain spin images. Intensity-domain spin images are a two-dimensional intensity histogram introduced by Lazebnik et al. [117]. The two-dimensions of the histogram are (i) the spatial distance from the salient point's center and (ii) the

intensity value. The authors propose the use of ten distance bins with ten intensity bins respectively, resulting in a 100-dimensional descriptor. Intensity-domain spin images achieve invariance to affine transformations by normalization of the intensities in the descriptor.

Cross-correlation. Cross-correlation is a very basic descriptor based on intensity. The original image is smoothed and uniformly sampled around the salient point. The sampled values are the descriptor. The descriptors of different salient points are matched using the cross-correlation of the descriptors. The computation of the descriptor is easy, however cross-correlation is more sensitive to geometric distortions than more sophisticated descriptors [145].

2.3.8 Features related to Motion

The visual features presented so far are defined on single images and thus represent static information. An important aspect introduced by film and video is motion. A number of features exist for the representation of the motion content in a sequence of frames. Features that quantitatively capture the amount of motion in a sequence are obtained from pixel differences and histogram differences. Other features represent the spatio-temporal distribution of motion (spatio-temporal slices). More advanced features first explicitly estimate the motion of the pixels between successive frames and then accumulate the resulting motion field to obtain a compact representation of the motion content (MPEG-7 motion activity). Additionally, motion information can be obtained from feature trackers that compute motion trajectories for distinct points in a sequence. Trajectory features capture invariant and characteristic information from the trajectories.

Pixel differences. A straight forward approach to estimate the amount of motion between two frames is to sum the intensity differences of all pixels of the two frames (sum of absolute differences). The more motion is present between the two frames the higher is the accumulated pixel difference. Unfortunately, the opposite is not true in general. Illumination changes may also produce high frame-to-frame differences. Pixel differences between successive frames are further employed for the detection of moving objects in surveillance applications, for example in [232].

Difference of histograms. An alternative method for the estimation of the amount of motion is the accumulation of histogram differences over time [175]. The authors first compute the difference between successive color histograms (one minus the histogram intersection, see Section 2.4.3). Next, they compute the mean of the histogram differences over an entire shot of a film. The result is a coarse measure of the amount of motion in a shot. Since histograms neglect spatial information, motion of objects in the shot may be missed by the feature. Similarly to pixel differences this feature is not robust to illumination changes.

Spatio-temporal slices. A different approach for the extraction of motion content are spatio-temporal slices [157]. For the extraction of spatio-temporal slices a frame sequence is regarded as a three-dimensional volume, where the first dimension is x, the second dimension y, and the third dimension is time t, see Figure 2.8. Spatio-temporal slices are obtained by extracting intersection planes from the volume parallel to the (x, t) plane (horizontal slice) and the (y, t) plane (vertical slice). The horizontal and vertical slices capture the horizontal and vertical motion of pixels over time. The slices show characteristic patters for different camera motions. For a static sequence both, horizontal and vertical slices show horizontal lines. For a sequence with a camera pan, the vertical slice contains horizontal lines while the horizontal slice shows slanted lines, where the sign of the slope represents the direction of the pan and the absolute value of the slope corresponds to the velocity of the pan. The authors of [157] capture the local orientation of motion (slope of the lines) in the slices by a structure tensor and capture the distribution of orientations over time in a tensor histogram. The horizontal and vertical tensor histograms allow the extraction of dominant motion directions. The authors of [157] employ this information for the classification of different camera motions.

MPEG-7 motion activity. The motion activity descriptor compactly represents the spatio-temporal distribution of motion magnitude and direction in a sequence [101]. The descriptor is computed from a dense motion field. A dense motion field contains a vector for each pixel (or each image block) that represents the motion of that pixel (or block) between two successive frames. Motion fields can be obtained for example directly from the motion vectors of macroblocks in a compressed video stream or from

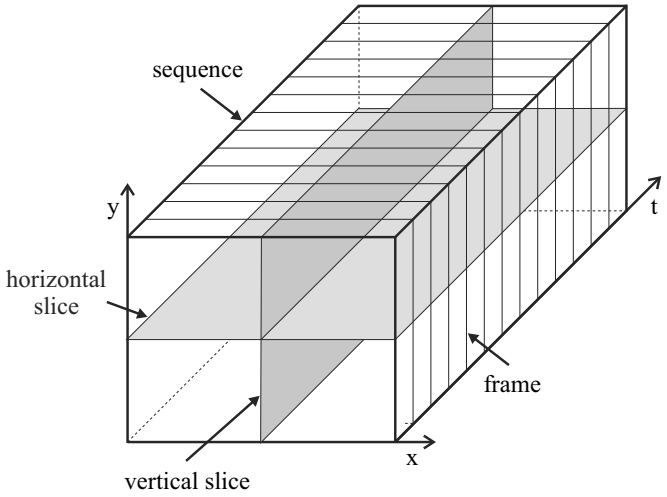


Figure 2.8: The concept of spatio-temporal slices: horizontal and vertical slices represent horizontal and vertical motion over time.

the original (uncompressed) frame sequence by optical flow methods [34, 97, 135]. The motion activity descriptor extracts five different parameters from the motion fields of an entire frame sequence: (i) the *motion intensity* of the sequence which is the standard deviation over all vector magnitudes, (ii) the *dominant motion direction* in the sequence as an angle between 0 and 360 degrees, (iii) the *spatial coherency* of motion which reflects the number and size of moving regions in the sequence, (iv) the *spatial distribution* of motion which contains the mean motion magnitude aggregated over time for different image blocks (uniformly split sub-images), and (v) the *temporal distribution* of motion intensity in terms of an intensity histogram which globally represents the occurrence frequency of different motion intensities. The motion activity descriptor compactly represents different spatial and temporal aspects of the underlying dense motion field and has been employed among others for keyframe extraction [154] and video summarization [55].

Motion trajectory features. Motion information can further be obtained from feature trackers. Feature trackers locate salient points in a frame and try to track them through succeeding frames of a sequence [194]. The result of feature tracking is a sparse set of motion trajectories. In contrast to dense motion fields, the sparse motion fields obtained by a feature tracker capture motion only where it actually appears in the sequence. Different types of features have been proposed for the description of

motion trajectories. A trajectory representation that is invariant to two-dimensional affine transforms is presented in [93]. Instead of representing a trajectory by its spatial coordinates at each time instant, the authors represent each point of the trajectory by the product of its curvature and its velocity magnitude. The resulting representation is invariant to two-dimensional affine transforms but does not reduce the amount of data necessary to store the trajectory. The authors of [122] propose a *directional histogram* that compactly represents the directional information of a trajectory. Each bin in the histogram corresponds to a directional interval. Each point along the trajectory is assigned to that bin whose directional interval encompasses the slope at that point. Finally, the directional histogram is normalized by the total number of points of the trajectory. The directional histogram provides a coarse but compact (fixed-length) representation of a trajectory [122]. In Chapter 9, we employ feature tracking for the extraction of motion information. We employ robust estimates of direction and magnitude as features to represent and compare different trajectories for the clustering of trajectories into motion components, see Section 9.2.2.

2.4 Similarity Measurement

2.4.1 Introduction

In the previous sections, we have reviewed widely used auditory and visual features for the description of media objects. As mentioned in Section 2.1 the result of feature extraction is a d-dimensional vector that represents the underlying media object. The feature vectors of the media objects are usually considered as points in a d-dimensional vector space $V = \mathbb{R}^d$ (vector space model). A crucial step in the retrieval process is the assessment of similarity between different media objects (feature vectors). The basic assumption in the vector space model is that *similar* objects are positioned *near* to each other in the vector space while dissimilar objects are spatially separated.

Distances in the vector space (feature space) can be measured by distance functions (distance measures). For this purpose, we assume that the feature space V is a metric space. The distance δ_{ij} between two media objects represented by feature vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ is measured by a distance function $\delta_{ij} = d(\mathbf{x}_i, \mathbf{x}_j)$ that satisfies the basic metric axioms [186]:

1. *Constancy of self-similarity*: $d(\mathbf{x}_i, \mathbf{x}_i) = d(\mathbf{x}_j, \mathbf{x}_j)$,
2. *Minimality*: $d(\mathbf{x}_i, \mathbf{x}_j) \geq d(\mathbf{x}_i, \mathbf{x}_i)$,
3. *Symmetry*: $d(\mathbf{x}_i, \mathbf{x}_j) = d(\mathbf{x}_j, \mathbf{x}_i)$, and
4. *Triangle inequality*: $d(\mathbf{x}_i, \mathbf{x}_k) + d(\mathbf{x}_k, \mathbf{x}_j) \geq d(\mathbf{x}_i, \mathbf{x}_j)$.

In literature it is often required additionally that $d(\mathbf{x}_i, \mathbf{x}_i) = 0$ (*identity of indiscernibles*). Given this condition and the second metric axiom (minimality) it follows that $d(\mathbf{x}_i, \mathbf{x}_j) \geq 0$ (*non-negativity*). Consequently, a distance function d is a non-negative function $d : X \times X \to \mathbb{R}_o^+$ [190].

While these axioms are a convenient formal basis for the definition of distance functions, psychological experiments have shown that the metric axioms are too restrictive for *perceptual* similarity judgments. All of the above axioms have been rejected or could at least not be verified in psychological experiments. For example, the constancy of self-similarity (axiom 1) has been refuted by [114]. Furthermore, it could be shown that the second and third axiom, minimality and symmetry, are violated in particular experiments [180, 219]. Finally, the triangle inequality does not hold in all experiments and there is no evidence that human similarity judgments follow the triangle inequality [220]. Although it has been shown that the metric axioms are too restrictive for human similarity perception, this does not mean that distance functions are generally inappropriate for similarity judgments. It rather shows that distance functions can in the best case only *approximate* human similarity perception. Additionally in practice distance functions are employed that do not fulfill all metric axioms [190]. We overview important distance (dissimilarity) functions in Section 2.4.2 and present similarity measures in Section 2.4.3.

Given a set of n d-dimensional feature vectors $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, where X is an $n \times d$ matrix, we refer to the elements (components) of a feature vector $\mathbf{x}_i$ from X as x_i^k with $k \in \{1, \ldots, d\}$. Prior to distance computation, it may be useful to *normalize* the feature vectors to reduce bias during comparison. If the components of the feature vectors are in different numerical ranges, the components with higher absolute values may have a stronger influence on the distance computation than components with lower absolute values. This bias can be removed by normalizing the value range of the feature components across all n observations (*min-max normalization*). Min-max

normalization linearly scales the feature components to the range $[0, 1]$. A feature component k is normalized by subtracting the minimum over all n observations and by dividing by the maximum:

$$\widehat{x}_i^k = \frac{x_i^k - \min_{j=1}^n (x_j^k)}{\max_{j=1}^n (x_j^k) - \min_{j=1}^n (x_j^k)}, \text{ for } j = 1, \ldots, n. \tag{2.1}$$

After normalization of all $k = 1, ..., d$ feature components, the values in X are in the range $[0, 1]$. This means that all normalized feature vectors $\widehat{\mathbf{x}}_i$ lie in a d-dimensional unit cube. Optionally, new minimum and maximum values (different from 0 and 1) can be considered during normalization [79].

2.4.2 Distance Measures

The choice of the distance measure is a critical parameter in a retrieval system. The distance measure influences what kind of information from the feature vectors is compared in distance computation and which information is neglected. Neglecting information may induce invariance against transforms like scaling, translation, and rotation.

A large group of distance measures are represented by the *generalized Minkowski metric* (also called L_q metric):

$$d_{L_q}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt[1/q]{\sum_{k=1}^{d} \left| x_i^k - x_j^k \right|^q}, \tag{2.2}$$

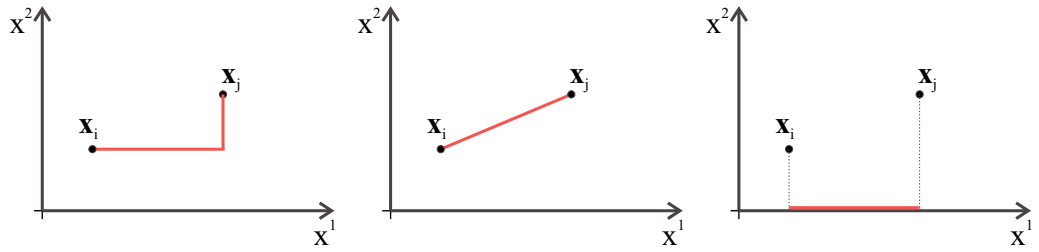
where $q \geq 1$ is the order of the metric. For $q = 1$ the *city block* or *Manhattan* metric is obtained:

$$d_{L_1}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{d} \left| x_i^k - x_j^k \right| \ . \tag{2.3}$$

The city block distance between two points is the sum of absolute distances along each coordinate axis between the two points, i.e. it allows one only to travel parallel to the coordinate axes (see Figure 2.9(a) for an illustration). For $q = 2$ we obtain the *Euclidean distance* (L_2 metric):

$$d_{L_2}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{k=1}^{d} \left| x_i^k - x_j^k \right|^2}, \tag{2.4}$$

which represents the shortest distance between two points in space, see Figure 2.9(b). The Euclidean distance is a widely used distance measure in literature. It seems to be



(a) city block (L_1) (b) Euclidean (L_2) (c) Chebyshev (L_∞)

Figure 2.9: Minkowski distances for $q = 1, 2,$ and ∞ in two dimensions. For the Chebyshev distance, the larger component difference (first dimension in this example) is taken as distance.

the "natural" distance measure in a Cartesian coordinate system, since it intuitively corresponds to what a human observer expects from a distance measure in two or three dimensions. In practice however, it has been shown in psychological experiments, the Euclidean metric suboptimally reflects human distance judgments [16] and that in particular experiments for example the city block distance outperforms the Euclidean distance. Additionally, Aggarwal et al. point out that the natural interpretation of the Euclidean distance is irrelevant in higher dimensions [3].

The Minkowski distances take all feature components into account and yield a low distance value only if all components are similar. In retrieval experiments however, it has been shown that similarity between objects is often characterized by only a few similar components [119]. Furthermore, it has been shown in the study of [119] that when two similar objects are compared to a third similar object their respective subsets of similar feature components are different. According to the study the assumptions made by Minkowski distances do not hold in practice, especially in high dimensions. The generalized Minkowski distance is translation invariant, while it is generally not invariant to scaling and rotation. Only for the Euclidean distance ($q = 2$) invariance to rotation is given [57]. Since, the Minkowski distance computes absolute differences between all components, the components should lie in the same value range in order to avoid that components with high values bias the distance computation. The range of the feature components can be equalized by normalization (see Section 2.4.1).

A further measure derived from the generalized Minkowski metric is the *Chebyshev distance*. For $q \to \infty$ the Minkowski distance L_∞ becomes the Chebyshev distance:

$$d_{L_\infty}(\mathbf{x}_i, \mathbf{x}_j) = \max_{k=1}^{d} \left(\left| x_i^k - x_j^k \right| \right) . \tag{2.5}$$

The Chebyshev distance represents the maximum absolute difference between two components of a pair of vectors (see Figure 2.9(c)). In contrast to all other Minkowski distances, the Chebyshev distance does only take one component of the vectors (with maximum difference) into account. Consequently, it may underestimate similarity (overestimate distance) and is sensitive to outliers [178].

The χ^2 distance is a measure derived from statistics that compares two distributions with each other[126]. If the inputs $\mathbf{x}_i$ and $\mathbf{x}_j$ are two binned distributions (e.g. histograms), the χ^2 distance estimates to which degree two histograms belong to the same distribution:

$$d_{\chi^2}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{d} \frac{(x_i^k - m^k)^2}{m^k}, \tag{2.6}$$

where vector $\mathbf{m} = \frac{\mathbf{x}_i + \mathbf{x}_j}{2}$ is the mean of both distributions and m^k represent the components of the vector. Substituting vector $\mathbf{m}$ in Equation (2.6) yields the χ^2 distance in the following form:

$$d_{\chi^2}(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{2} \sum_{i=1}^{d} \frac{(x_i^k - x_j^k)^2}{x_i^k + x_j^k} \tag{2.7}$$

The χ^2 distance is similar to the squared Euclidean distance. It differs in that the squared differences between the components of $\mathbf{x}_i$ and $\mathbf{x}_j$ are normalized by their sum. This normalization reduces the bias if the components have different value ranges. The χ^2 distance has been successfully applied to histogram features, for example in shot cut detection [192]. An analytically similar measure to the χ^2 distance is the *Canberra distance* [115]. It is defined as:

$$d_{can}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{d} \frac{\left| x_i^k - x_j^k \right|}{\left| x_i^k \right| + \left| x_j^k \right|}. \tag{2.8}$$

The Canberra distance can be regarded as a normalized city block distance [58]. If both input vectors are zero vectors, the Canberra distance must be set to zero to avoid a division by zero. Similarly to the χ^2 distance the normalization reduces bias by

differently scaled components. The Canberra distance has for example been successfully applied to texture comparison in [112] where it outperformed all other evaluated distance measures.

2.4.3 Similarity Measures

For the estimation of similarity, distance functions are mapped to similarities. A similarity measure is a function s that maps a non-negative real number (a distance) into the interval $[0, 1]$, where 0 denotes the minimum of similarity and 1 corresponds to the maximum of similarity. Additionally, the function s should be (i) strictly monotonically decreasing:

$$d(\mathbf{x}_i, \mathbf{x}_j) > d(\mathbf{x}_i, \mathbf{x}_k) \Rightarrow s(d(\mathbf{x}_i, \mathbf{x}_j)) < s(d(\mathbf{x}_i, \mathbf{x}_k)), \tag{2.9}$$

to allow for a consistent mapping and (ii) continuous to avoid jumps [190]. If we assume, that a distance function has a minimum value of 0 and a maximum value of d_{max}, a straight-forward mapping is to invert and linearly rescale the distance function (see Figure 2.10(a)):

$$s(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{d(\mathbf{x}_i, \mathbf{x}_j)}{d_{max}}. \tag{2.10}$$

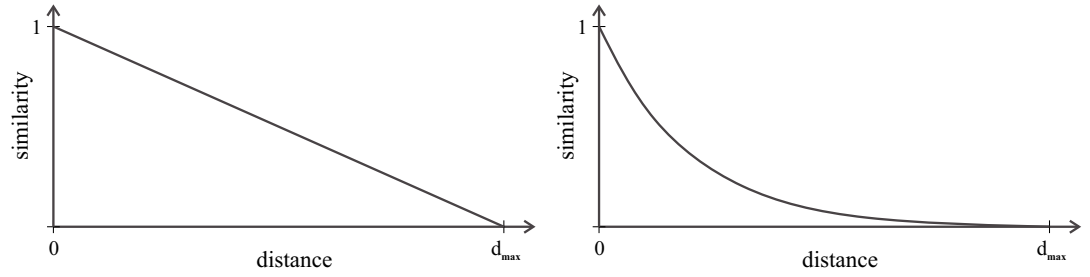
This mapping is problematic if d_{max} is very large (or even infinity) because the sensitivity of the resulting similarity measure for small distances decreases significantly. Psychological research indicates that an exponential relation exists between distance and similarity (see Figure 2.10(b)). According to Shepard [193] the mapping from distance to similarity is approximately:

$$s(\mathbf{x}_i, \mathbf{x}_j) = e^{-d(\mathbf{x}_i, \mathbf{x}_j)}. \tag{2.11}$$

This mapping takes the property into account, that the perceived similarity decreases only to up to a certain limit and then quickly flattens out towards 0. Additionally it increases the sensibility to small distances. Principally, each distance function can be transformed into a similarity measure by an appropriate mapping.

Additionally to(mapped) distance functions, there are measures that directly compute similarity between two points in space. A popular example is the *Cosine similarity* which represents the cosine of the angle between two vectors:

$$s_{cos}(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\| \, \|\mathbf{x}_j\|} = \frac{\sum_{k=1}^{d} x_i^k x_j^k}{\sqrt{\sum_{k=1}^{d} {x_i^k}^2} \sqrt{\sum_{k=1}^{d} {x_j^k}^2}}, \tag{2.12}$$



(a) linear mapping (b) exponential mapping

Figure 2.10: Different functions for mapping distances to similarities.

where $\|\mathbf{x}\|$ is the norm (length) of vector $\mathbf{x}$ and operator " $\cdot$ " represents the inner product. Due to the normalization, the Cosine similarity only takes the direction of the vectors into account but not the length of the vectors. The cosine similarity is 1 for vectors pointing in the same direction, 0 if two vectors are orthogonal and -1 for vectors that point in opposite directions. If the direction where the vectors point to is not important for similarity comparison (i.e. vectors pointing in opposite direction are considered identical), the absolute value of the cosine similarity may be used. Otherwise, the cosine similarity can by mapped into the range $[0, 1]$ by computing $(s_{cos} + 1)/2$.

A similarity measure from statistics is the *Pearson product-moment correlation coefficient*. This correlation coefficient measures the linear dependence between two random variables [33]:

$$s_{cor}(\mathbf{x}_i, \mathbf{x}_j) = \frac{(\mathbf{x}_i - \overline{\mathbf{x}}_i) \cdot (\mathbf{x}_j - \overline{\mathbf{x}}_j)}{\|\mathbf{x}_i - \overline{\mathbf{x}}_i\| \, \|\mathbf{x}_j - \overline{\mathbf{x}}_j\|} = \frac{\sum_{k=1}^{d} (x_i^k - \overline{x}_i^k)(x_j^k - \overline{x}_j^k))}{\sqrt{\sum_{k=1}^{d} (x_i^k - \overline{x}_i^k))^2} \sqrt{\sum_{k=1}^{d} (x_j^k - \overline{x}_j^k))^2}}, \quad (2.13)$$

where $\overline{\mathbf{x}}_i$ is a vector of the same dimension as $\mathbf{x}_i$ that contains at each position $\overline{x}_i^k$ the mean of $\mathbf{x}_i$: $m = \frac{1}{d} \sum_{k=1}^{d} x_i^k$. If two vectors share a perfect linear relationship the correlation becomes 1. For linearly independent vectors the correlation is 0. A negative correlation results in a value between $[0, -1]$. If $\overline{\mathbf{x}}_i$ and $\overline{\mathbf{x}}_j$ are zero vectors ($\mathbf{x}_i$ and $\mathbf{x}_j$ have already zero mean) the correlation coefficient is equal to the Cosine similarity.

A similarity measure for binned data is *histogram intersection* proposed by [207]. Histogram intersection is the normalized intersection of two histograms:

$$s_{hi}(\mathbf{x}_i, \mathbf{x}_j) = \frac{\sum_{k=1}^{d} \min\left(x_i^k, x_j^k\right)}{\sum_{k=1}^{d} x_j^k} \quad (2.14)$$

and represents the common entries of two histograms. If for example $\mathbf{x}_i$ and $\mathbf{x}_j$ are two color histograms, histogram intersection represents the portion of pixels (or the number of pixels if normalization is skipped) in the corresponding images which have the same color. Histogram intersection (with normalization) yields values between 0 and 1 where 0 means that the intersection between both histograms is empty and 1 means that the two histograms are identical. If the sum over both histograms is equal ($\sum_{k=1}^{d} x_i^k = \sum_{k=1}^{d} x_j^k$), it can be shown that histogram intersection is equivalent to the inverted city block distance [207].

Principally, the choice of a distance or similarity measure strongly influences the performance of a retrieval system. The identification of an appropriate measure for a particular task is non-trivial since it is difficult to predict the performance of different measures in advance. In practice, the best-suited distance measure often has to be evaluated empirically.

2.5 Classification

2.5.1 Introduction

The objective of classification is to predict the class membership of a pattern represented by a feature vector. A class ω_i is defined by a class label $i \in \Omega$ where $\Omega = \{1, ..., C\}$ is the set of all class labels and C the number of classes. Each pattern (feature vector) belongs to exactly one class. A classifier can be regarded as a function $c(\mathbf{x})$ of a feature vector $\mathbf{x}$ with:

$$c(\mathbf{x}) = i \Leftrightarrow \mathbf{x} \in \omega_i \tag{2.15}$$

Most classifiers have to be trained before they can be applied to arbitrary test patterns. For this purpose, the available data set is split into training and test sets (see Section 2.6). The training samples are usually chosen randomly, for example by cross-validation [57]. During training, the classifier tries to fit a model to the training data. The quality of the classifier is evaluated by a test set. The test set contains unlabeled feature vectors that are not contained in the training set. A classifier should be able to correctly predict the class labels not only of the test and training vectors, but all arbitrary vectors that belong to one of the selected classes. This is called the generalization ability of a classifier [57].

A large number of different classifiers has been proposed in literature, see [95] for a survey. In the following, we give a brief overview of different classification approaches and then discuss two classifiers which are of special relevance to this thesis in more detail. Generally, we distinguish between generative approaches and discriminative approaches. *Generative* approaches try to empirically estimate the joint probability distribution (density) of the underlying data for each class. For this purpose distribution parameters like mean and covariance matrix of each class are estimated. A new data sample is then assigned to the class which is most likely for the sample based on the estimated densities. Popular generative techniques are the Bayes classifier [57], Gaussian Mixture Models (GMMs) [24], and Hidden Markov Models (HMMs) [171].

Generative classifiers require a large number of training samples for density estimation. With increasing dimension d of the feature vectors, the volume of the corresponding vector space grows exponentially. Consequently, the number of parameters for density estimation grows, as well. The larger the number of parameters, the more data samples must be provided to get reliable estimates. This means that with increasing dimension d, a much (exponentially) larger number of data samples must be provided in order to model the data accurately. Otherwise, the estimates of the parameters become less reliable which results in a degradation of the classifier's performance. This circumstance is called the "curse of dimensionality" [57, 102].

For high-dimensional data, the complexity of density estimation increases rapidly and the estimation often fails or the classifier overfits the data in absence of sufficient training samples. *Discriminative* approaches try to find a function that models the class boundaries directly instead of estimating the joint density of the data for each class. There are probabilistic and non-probabilistic discriminative approaches. Probabilistic discriminative approaches model the conditional probability distribution of a class given the training data. The complexity of modeling the conditional probability distribution is lower than modeling the joint density (especially in high dimensions). A popular example is logistic regression that tries to predict the correct class label from given input data by regression [98]. Additionally to probabilistic approaches, there exist non-probabilistic approaches. The idea of such approaches is to find a function with a preferably low number of parameters (e.g. a linear function) that separates the classes as best as possible. This alleviates the curse of dimensionality since the number of parameters is independent of the size of the training set and better prevents

overfitting (which improves the generalization ability of the classifier). Popular non-probabilistic discriminative classifiers are the Support Vector Machine [29, 228] and the Perceptron [181].

Additionally to generative and discriminative approaches there are classifiers that "learn" directly from prototypes (*instance-based learning*). Such classifiers compare new samples with samples in the training set without building a model from the data. Instance-based learners usually have a low number or even no parameters since they do not make assumptions about the data. As a consequence, their generalization ability is low. The most popular instance-based learning algorithm is nearest neighbor classification [49]. The nearest neighbor classifier simply stores the entire training set and assigns a new sample the class label of the nearest neighbor in the training set.

In the following, we describe two classifiers in more detail which are of special relevance to this thesis. First, we present nearest neighbor classification, since it is related to similarity retrieval where nearest neighbor search is the standard search strategy. We employ nearest neighbor search for the retrieval of motion composition and visual composition in Chapters 9 and 10. Second, we describe Support Vector Machines, which are used for the detection of intertitles in Chapter 4 and gradual transitions in this Chapter 6.

2.5.2 Nearest Neighbor Classification

The nearest neighbor (NN) classifier is a simple instance-based classifier that takes as input a set of n training examples $X = (\mathbf{x}_1, ..., \mathbf{x}_n)$ and a vector of corresponding class labels $y = (y_1, ..., y_n) \in \mathbb{R}^{1 \times n}$ with elements $y_i \in \Omega$ [49]. The NN classifier assigns a new vector $\mathbf{x}$ the class label y_s of the nearest training vector $\mathbf{x}_s$, where:

$$s = \underset{j}{\operatorname{argmin}} \left(\|\mathbf{x} - \mathbf{x}_j\| \right), \ 1 \leq j \leq n. \tag{2.16}$$

Distances in nearest neighbor search can be measured by an arbitrary distance metric $\|.\|$, see for examples Section 2.4. In practice most frequently Euclidean distance is employed which is however not necessarily the best distance measure for a given set of data.

The assignment scheme of nearest neighbor partitions the feature space according to a Voronoi tessellation. The edges in the tessellation correspond to decision boundaries between two classes. Each cell belongs to one class. Figure 2.11 illustrates a Voronoi

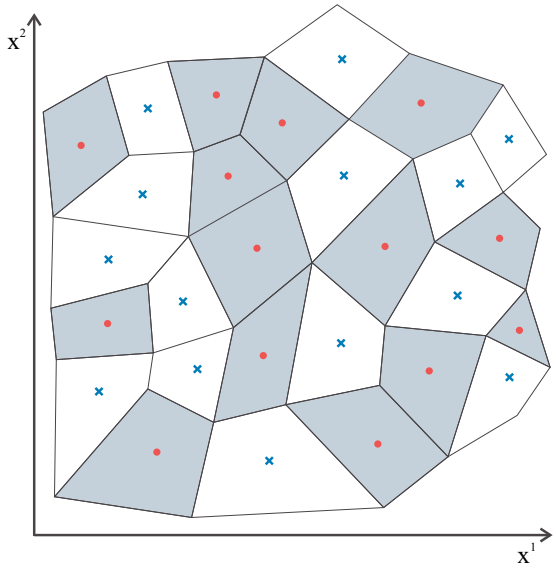


Figure 2.11: Voronoi tessellation in $\mathbb{R}^2$ with features $\mathbf{x}^1$ and $\mathbf{x}^2$ of a binary classification problem. Dots are feature vectors of class ω_1, crosses refer to feature vectors of class ω_2. The gray area is the decision region of class ω_1, the white area represents class ω_2.

tessellation in two-dimensional space for a binary classification problem. The union of all cells that are assigned to the same class, represents the decision region for this class.

The nearest neighbor classifier generates complex decision boundaries for which all training samples are taken into consideration. The decision regions obtained by NN are robust (changing one sample in the training set influences the decision region only locally). However, since all training samples contribute to the decision region and have a corresponding region of influence, outliers may generate patches with a wrong class label.

The K-Nearest Neighbor (K-NN) classifier is an extension of the nearest neighbor classifier which is more robust to outliers. The K-NN classifier takes the K nearest neighbors into account for assigning a new feature vector $\mathbf{x}$ to a class ω_i. From the K neighboring vectors of $\mathbf{x}$, k_j vectors belong to class ω_j, with $\sum_{j=1}^{C} k_j = K$, and C is the number of classes. Vector $\mathbf{x}$ is assigned to class ω_i with the greatest number of representatives (majority vote) in the set of K neighbors:

$$i = \underset{j}{\operatorname{argmax}}\ k_j,\ 1 \leq i \leq C\,. \tag{2.17}$$

If the majority vote is not unique the 1-nearest neighbor can be chosen. For $K = 1$ the K-NN classifier becomes the standard NN approach. Both, K-NN and NN learn the training set by rote. Hence, memory and computation costs grow linearly with the size of the training set ($O(nd)$), where n is the size of the training set and d the dimension of the feature vectors. While training requires practically no computation time, $O(1)$, testing is computationally expensive ($O(nd)$, plus the time needed for majority voting in the case of K-NN).

The nearest neighbor strategy is not only used in classification but the natural search strategy in similarity retrieval. In similarity retrieval we are given a database of n media objects which are represented by n d-dimensional feature vectors $\{\mathbf{x}_1, ..., \mathbf{x}_n\}$ and a feature vector $\mathbf{x}' \in \mathbb{R}^d$ extracted from a user-specified query. Nearest neighbor search is used to estimate the feature vector $\mathbf{x}_i$ from the database that best matches the query vector $\mathbf{x}'$. As a result the corresponding media object is returned to the user. Analogously, the K nearest neighbors can be determined and returned.

2.5.3 Support Vector Machines

The Support Vector Machine (SVM) is a binary linear classifier introduced by Vladimir Vapnik and colleagues [48, 227, 228]. In contrast to K-NN, an SVM tries to find a preferably simple (linear) boundary that separates two classes. Again we are given n training samples $X = (\mathbf{x}_1, ..., \mathbf{x}_n)$ in $\mathbb{R}^d$. The vector of corresponding class labels $y = (y_1, ..., y_n)$ contains values $y_i \in \{-1, +1\}$ corresponding to the two classes ω_1 and ω_2. The objective of SVM training is to find a function $g(\mathbf{x})$ such that:

$$\begin{aligned} &\text{sign}(g(\mathbf{x}_i)) = -1 \text{ if } \mathbf{x}_i \in \omega_1 \text{ and} \\ &\text{sign}(g(\mathbf{x}_i)) = +1 \text{ if } \mathbf{x}_i \in \omega_2. \end{aligned} \tag{2.18}$$

The function $g(\mathbf{x})$ is called *discriminant function*. In the case of SVMs $g(\mathbf{x})$ is linear and represents a hyperplane in $\mathbb{R}^d$: $g(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$, where $\mathbf{w}$ (weight vector) is the d-dimensional normal vector of the hyperplane, b (bias) is the translation of the hyperplane along $\mathbf{w}$. For $b = 0$ the hyperplane goes through the origin. The dot operator " $\cdot$ " denotes the inner product of two vectors.

Two classes ω_1 and ω_2 are *linearly separable* if there exists a weight vector $\mathbf{w}$ and a bias b such that $\text{sign}(\mathbf{w} \cdot \mathbf{x}_i + b) = y_i$ for all samples $\mathbf{x}_i$ in X, i.e. all samples can

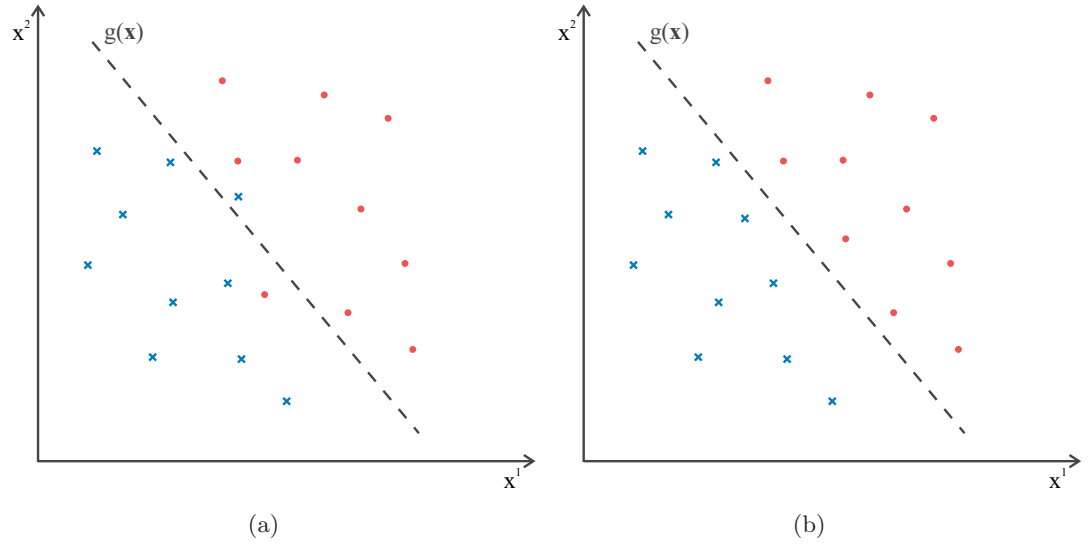


Figure 2.12: Linear separability of classes: (a) the classes are not linearly separable. There is no linear function that is able to separate the two classes without errors (b) the classes are linearly separable.

be correctly classified. Figure 2.12 represents examples for linearly separable and non-separable classes in two-dimensional space. If the training samples of the two classes are *linearly separable*, then the SVM constructs an optimal separating hyperplane $\mathbf{w}\cdot\mathbf{x}+b = 0$ between both classes, that maximizes the distance between the hyperplane and the nearest data points of each class. The data points that determine the hyperplane are the *support vectors*. The distance between the support vectors and the hyperplane is called *margin*. Figure 2.13 depicts the difference between a suboptimal and an optimal separating hyperplane. For an optimal separating hyperplane the margin to both sides is maximized. The larger the margin the higher is the robustness of the classifier.

From Figure 2.13 we observe that the separating hyperplane is defined by a few support vectors only which all have the same distance to the hyperplane. Not all training samples contribute to the hyperplane (as for example in the case of nearest neighbor classification). The support vectors are those samples that are most difficult to separate and consequently the most important samples for the classification task [57]. Due to the low number of support vectors the generalization ability and the robustness of SVMs is generally high [48].

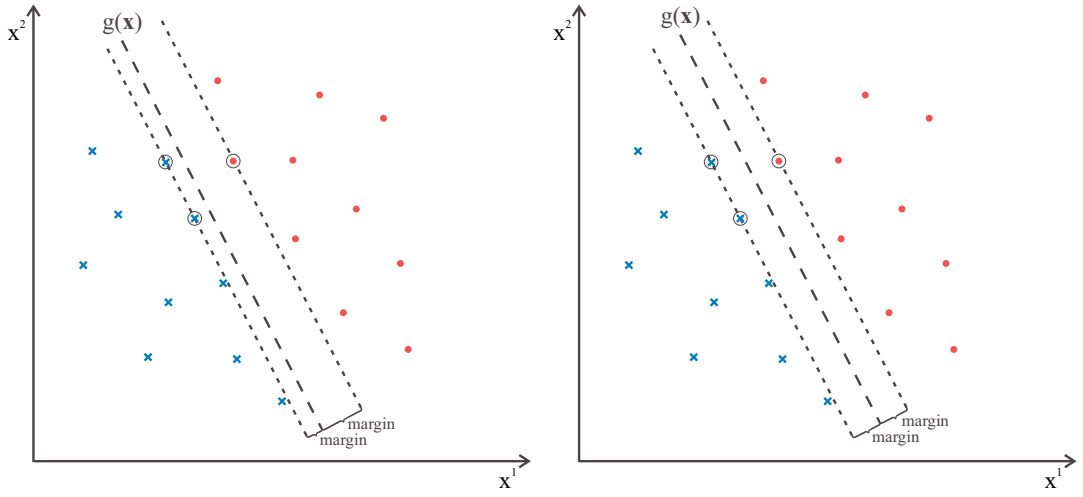


Figure 2.13: Optimal separating hyperplanes: (a) the margin of the hyperplane $g(\mathbf{x})$ is not optimal. (b) shows a hyperplane with maximized margin. The support vectors are encircled.

According to Cortes and Vapnik [48] the optimal hyperplane that maximizes the margin can be computed by estimating the saddle point of the following Lagrange functional:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w} \cdot \mathbf{w} - \sum_{i=1}^{n} \alpha_i \left[y_i \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) - 1 \right], \tag{2.19}$$

where α_i with $1 \leq i \leq n$ are the Lagrange multipliers. The optimal parameters for the hyperplane are obtained by finding the saddle point where $\mathbf{w}$ and b are minimized and $\boldsymbol{\alpha}$ is maximized. The functional can be reformulated into a maximization problem in $\boldsymbol{\alpha}$ (see [65] for details):

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j, \tag{2.20}$$

subject to the conditions:

$$\alpha_i \geq 0, \quad \sum_{i=1}^{n} \alpha_i y_i = 0, \quad 1 \leq i \leq n \ . \tag{2.21}$$

This representation is the dual form of Equation (2.19). It is noteworthy that Equation (2.20) only requires the computation of inner products between feature vectors $\mathbf{x}_i$

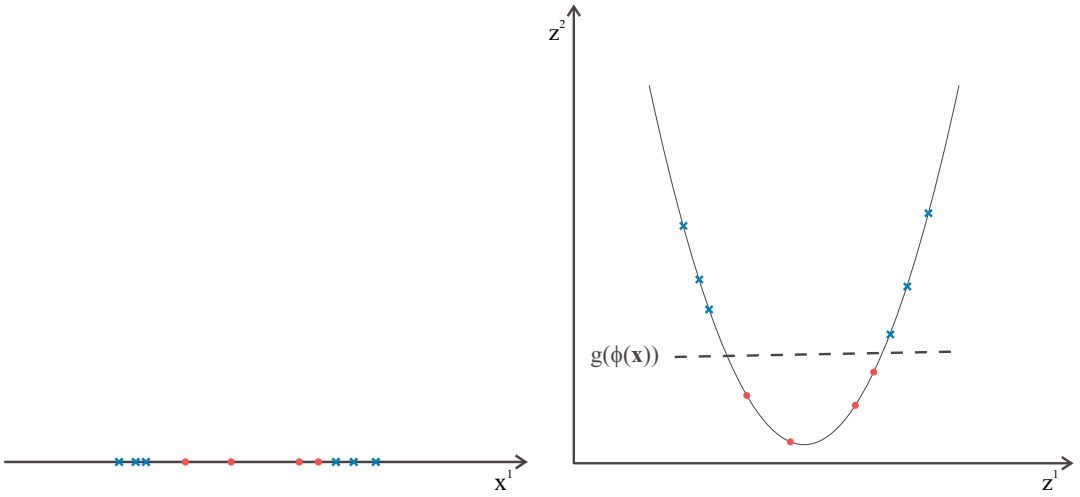
and $\mathbf{x}_j$. This is an important property for the integration of kernels in the following (non-linear discriminant functions).

In practice, two classes are often not linearly separable. For this reason, Cortes and Vapnik introduced *slack variables* which represent penalties for samples that can not be correctly classified by the linear discriminant function [48]. For each sample a slack variable ζ_i is introduced with $\zeta_i = 0$ for correctly classified samples and $\zeta_i > 0$ for misclassified samples. During optimization the sum of all slack variables $\sum_{i=1}^{n} \zeta_i$ is minimized. The separating hyperplane is constructed in such a manner that an optimal tradeoff is found between a maximum margin and a minimum number of misclassified samples.

For some data sets linear separation is generally suboptimal, see for example the data set in Figure 2.14(a). For such data sets non-linear classification is more suitable. However, the complexity of estimating non-linear discriminant functions is higher than that of linear functions. Fortunately, SVMs allow the integration of non-linear discriminant functions in a very efficient way.

The basic idea is the following: instead of estimating a non-linear discrimination function in feature space, the feature vectors are mapped non-linearly from the original feature space $\mathbb{R}^d$ into a higher dimensional space, the target space $\mathbb{R}^{d'}$, by a function $\phi : \mathbb{R}^d \to \mathbb{R}^{d'}$ with $d < d'$. In the higher dimensional target space the feature points move apart from each other which facilitates linear separability. Given an adequate mapping ϕ the data set becomes separable by a linear discrimination function $g(\phi(x))$ in the target space. This linear function in the target space is in turn a non-linear discrimination function in the original feature space. Figure 2.14 illustrates the effect of a non-linear transformation that maps the feature space into a higher-dimensional target space where the samples of the classes become linearly separable.

The transform ϕ and the computations in the high-dimensional target space are complex and can be avoided by the *kernel trick* [8]. Instead of transforming the feature vectors and comparing the feature vectors in the target space, an appropriate non-linear comparison function (the kernel) can be applied in the original space. From Equation (2.20), we observe that the data samples (feature vectors) contribute to the optimization problem only in terms of inner products. The inner product $\mathbf{x}_i \cdot \mathbf{x}_j$ in Equation (2.20) can be replaced by a kernel function K with $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$ that represents an inner product in the target space obtained by the mapping function ϕ.



(a) not linearly separable in $\mathbb{R}^1$ (b) linearly separable in $\mathbb{R}^2$

Figure 2.14: A non-linear mapping from $\mathbb{R}^1$ to $\mathbb{R}^2$: (a) the samples in the original feature space are not linearly separable. In the higher dimensional space (b) the samples become linearly separable.

By the use of kernels the computation of the mapping ϕ and the computations in the target space become implicit and thus can be avoided. This property makes non-linear classification with SVMs efficient.

Any continuous symmetric semi-positive definite function (Mercer's Theorem) is a valid kernel function [143]. This means, that each function that represents an inner product in the target space is a valid kernel [48]. The most popular kernels are [48]:

- Linear kernel: $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j$. The linear kernel does not transform the feature vectors and compares them in the original feature space (linear SVM).
- Polynomial kernel: $K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^d$, where $d > 0$ is the order of the kernel. For $d = 2$ we get a quadratic discriminant function.
- Gaussian radial basis function kernel: $K(\mathbf{x}_i, \mathbf{x}_j) = e^{\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2\right)}$, with $\sigma > 0$.
- Sigmoid kernel: $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(a\mathbf{x}_i \cdot \mathbf{x}_j + b)$. Note that the sigmoid kernel is not semi-positive definite for each combination of values of a and b [227].

Different types of kernels can be used to achieve discriminant functions of different complexity. Especially the global kernels, like the linear kernel and the polynomial ker-

nel usually avoid overfitting and alleviate the curse of dimensionality [48]. Local kernels, such as the radial basis function kernel have shown to be prone to overfitting [21].

2.6 Evaluation of Retrieval Systems

2.6.1 Ground Truth Generation

A fundamental requirement for the objective evaluation of a retrieval system is the availability of a *ground truth*. A ground truth represents the correct (true) assignment of input data to well-defined classes and concepts they belong to in the real world. A ground truth for image classification, for example may assign each image in a database a label that says whether or not it shows an indoor scene or an outdoor scene. Moreover, a ground truth may represent a list of points in time at which a shot cut occurs in a movie for the evaluation of an automatic shot boundary detector. Generally, the goal of a retrieval system is to predict the data in the ground truth as reliable as possible.

There are different ways to obtain ground truth data. One way is to generate the ground truth automatically together with the data, e.g. if synthetic test data is employed. Sometimes ground truth is already available, e.g. from preceding manual investigations and it only needs to be converted into a machine readable format. In most cases however ground truth has to be generated manually by annotating the corresponding media objects. The process of manual ground truth generation (annotation) poses several challenges.

First, ground truths have to be exact and correct. Consequently, annotation requires comprehensive domain knowledge. Principally, annotation should be performed by experts only which are well acquainted with the domain, its characteristics and the classes and concepts of interest.

Second, prior to annotation, an annotation vocabulary or annotation protocol must be defined that provides a detailed and explicit guide to the annotating person. The annotation protocol defines the classes or a taxonomy of classes, their corresponding labels, characteristics, and their relationships. In practice however different classes and concepts are often ambiguous and assessed subjectively by annotators due to cultural influences and different background knowledge. Especially for concepts with a certain degree of semantic richness ambiguities are introduced because such concepts often allow for different possible interpretations. An example are *scenes* in a movie. The

segmentation of a movie into scenes allows for different possible interpretations because there is no unique definition of a "scene" that covers all possible types of scenes that occur in movies in practice. Furthermore, even for much simpler concepts, such as *shot cuts* which have low semantic complexity, ambiguities have been reported [72]. So the major challenge in the creation of an annotation protocol is the specification of well-defined *distinct* classes and concepts in order to avoid ambiguities.

If a certain amount of ambiguity remains in the definitions and consequently also in the ground truth, a possible solution is to integrate an appropriate tolerance in the evaluation. If for example, a boundary between two scenes in a movie cannot be determined exactly, a certain amount of temporal tolerance may be incorporated in the evaluation to compensate for this uncertainty. Another possible solution is to perform parallel annotations by several experts. If their annotations differ, majority voting can be applied to obtain a more reliable ground truth [53].

The performance of a retrieval system heavily depends on the input data and the ground truth. In practice, due to the absence of commonly available annotated data, retrieval systems are often evaluated with proprietary (usually small) data sets and ground truths generated by non-experts. While such systems can be optimized (fit) to the data and the ground truth to yield a high performance, they poorly generalize to a broader and more representative dataset. Consequently, the performance of such systems is biased and not comparable to that of other systems.

The lack of readily available data is an underestimated challenge. Publicly available data with corresponding ground truth is a fundamental requirement for performance evaluation and comparison of retrieval systems. Public data sets and ground truths for selected retrieval tasks have been provided for example by the TRECVID evaluation [198]. However, we observe, there are still numerous research areas that suffer from the absence of publicly available ground truths such as scene segmentation in movies.

2.6.2 Systematic Evaluation

Ground truths enable the systematic evaluation of a retrieval system. We have presented two different retrieval architectures in Section 2.1. A typical query-based retrieval system, see Figure 2.1 and a typical classification task, see Figure 2.2. In the following, we present how systematic evaluation against a ground truth is integrated into both architectures.

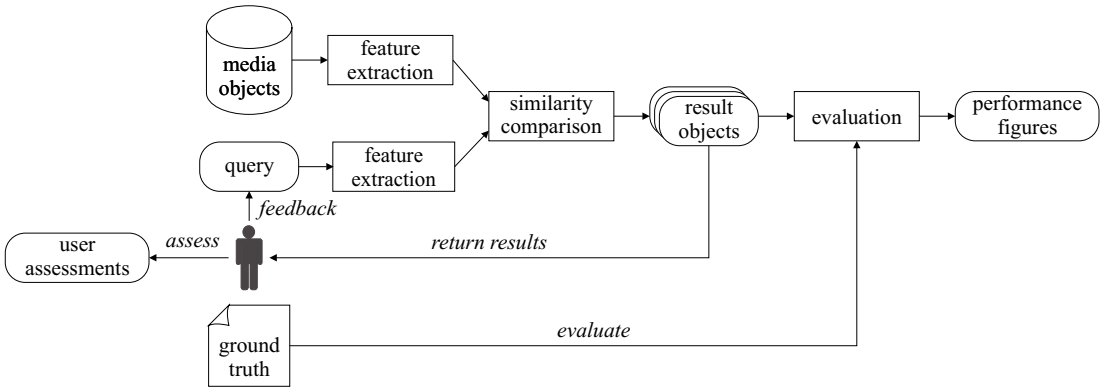


Figure 2.15: The workflow of a typical query-by-example retrieval system including the procedure for quantitative and qualitative evaluation.

A query-based retrieval system (see Figure 2.15) allows for two types of evaluations: qualitative and quantitative evaluation. In *quantitative* evaluation, the objects retrieved from a query are compared to the ground truth. For example, the class label of the result objects may be compared with the class label of the query. From this comparison, performance figures are computed (see below).

For certain tasks no ground truth is available at all. This impedes an objective evaluation. If no ground truth is available a retrieval system may be evaluated *qualitatively* by user assessments in the context of a user study. In this scenario several users subjectively evaluate the relevance of retrieved objects for a given class or concept. An example of a user study is presented in Chapter 10 for the retrieval of visual compositions.

In a classification scenario, evaluation is performed by comparing predicted class labels with the ground truth, see Figure 2.16. First, a classifier is trained with a training set and then the class labels for a disjoint set of test samples are predicted. Next, the predicted labels of the test samples are compared with the ground truth labels and performance figures are computed (see below).

For an objective evaluation the classifier hast to be trained with *different* training sets. Otherwise the classifier may overfit on the training data. Usually, *cross validation* is performed to obtain performance figures that are independent from the training set [57]. In m-fold cross validation, the dataset is randomly split into m different partitions of distinct training and test sets. For each of these m partitions, the corresponding training and test set together contain all samples of the database. The

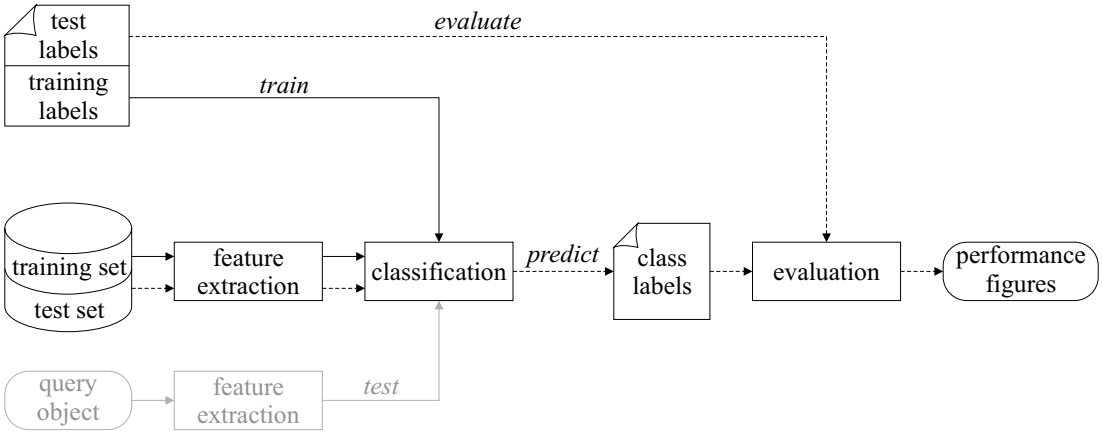


Figure 2.16: The workflow of a typical classification task including the procedure for evaluation. The database and the ground truth are split into a training and test set. The solid arrows are related to the training of the classifier, while the dashed arrows describe testing and evaluation. The query object is grayed out because it is not relevant for the systematic evaluation. It can be considered as being part of the test set.

classifier is trained with all m training sets and is evaluated with the m respective test sets. For evaluation, the mean performance over all m experiments is computed (cross validation error). Note that the cross validation error is not independent of the data since usually parameters are optimized by *iteratively* performing cross validation. As a consequence, the classifier gets progressively optimized (fitted) to the data. For an objective performance evaluation the classifier (trained by cross validation) has to be evaluated with *new* data that has not been used during cross validation.

2.6.3 Performance Measures

A binary classification experiment has four different possible outcomes which are summarized in Table 2.2. We distinguish between *relevant* documents[1] which we want to retrieve and *not relevant* or irrelevant documents which should not be retrieved. If a document is correctly assigned to the class of relevant documents it is a true positive (tp). Correctly rejected (not relevant) documents are true negatives (tn). Additionally, we distinguish between two types of errors that a classifier can make. False positives

[1]We use the term document since it is common in literature when talking about performance measures. A document in the context of this thesis may refer to a particular class of objects (e.g. intertitles) or a concept such as a scene or a shot.

	relevant	not relevant
predicted as relevant	true positive (tp)	false positive (fp)
predicted as not relevant	false negative (fn)	true negative (tn)

Table 2.2: Possible outcomes of a binary classification experiment.

(fp) are irrelevant documents that are falsely detected as relevant and false negatives (fn) are relevant documents that are not detected as relevant.

Different performance measures can be computed from the four key figures in Table 2.2. In the following, tp represents the number of true positives, fp the number of false positives, fn the number of false negatives and tn the number of true negatives obtained in an experiment. *Recall* is the number of retrieved and relevant documents divided by the total number of relevant documents in the database:

$$\text{recall} = \frac{|\{\text{retrieved}\} \cap \{\text{relevant}\}|}{|\{\text{relevant}\}|} = \frac{tp}{tp + fn} \ . \tag{2.22}$$

Recall is often also referred to as the detection rate. A high recall means that most of the relevant documents are actually retrieved. However, it does not consider the number of irrelevant documents that may be retrieved at the same time.

Precision is the number of retrieved and relevant documents divided by the total number of retrieved documents:

$$\text{precision} = \frac{|\{\text{retrieved}\} \cap \{\text{relevant}\}|}{|\{\text{retrieved}\}|} = \frac{tp}{tp + fp} \ . \tag{2.23}$$

High precision means that most of the retrieved documents are actually relevant. Precision does consider the number of relevant documents that are not retrieved. Both, recall and precision are in the range from 0 to 1.

Additionally, two error rates can be computed: the *false positive rate* (also called fallout), is the number of false positives divided by the total number of documents that are not relevant:

$$\text{false positive rate} = \frac{|\{\text{retrieved}\} \cap \{\text{not relevant}\}|}{|\{\text{not retrieved}\}|} = \frac{fp}{fp + tn} \tag{2.24}$$

and the *false negative rate* is the number of false negatives divided by the total number of relevant documents:

$$\text{false negative rate} = \frac{|\{\text{not retrieved}\} \cap \{\text{relevant}\}|}{|\{\text{relevant}\}|} = \frac{fn}{tp + fn} \ . \tag{2.25}$$

Recall and precision are two measures that are related inversely proportional to each other. Increasing the recall usually decreases the precision and vice versa. Each of the two measures can be optimized at the expense of the other. Consequently, the performance of a retrieval system is characterized by *both* performance figures. Alternatively, a combined performance figure can be computed from recall and precision, the f-measure. The generalized f-measure [226] is defined as:

$$f_\beta = (1+\beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}} = \frac{(1+\beta^2) \cdot tp}{(1+\beta^2) \cdot tp + \beta^2 \cdot fn + fp} \ . \tag{2.26}$$

The f-measure represents a harmonic mean of recall and precision [177] The parameter β is used to balance the influence of recall and precision. An f-measure with $\beta > 1$ puts more weight on recall while $\beta < 1$ weights precision stronger. For $\beta = 1$ we obtain the f_1-measure:

$$f_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{2 \cdot tp}{2 \cdot tp + fp + fn} \ , \tag{2.27}$$

which weights recall and precision equally.

The performance of a retrieval system is usually measured for different system configurations (e.g. for different values of a system parameter) resulting in a series of recall-precision pairs. The tradeoff between recall and precision values is best illustrated in a *recall-precision graph*. Since recall and precision are inversely proportional, the graph usually has a typical shape which is represented in Figure 2.17. The recall-precision graph shows the recall on the abscissa for different precisions on the ordinate. The general goal is to maximize recall and precision at the same time (raise both towards 1). The maximum f_1 score is obtained by the pair for which recall and precision are approximately equal (see the encircled point in Figure 2.17). A perfect retrieval system would achieve an f_1 score of 1 (which corresponds to the top-right corner of the graph).

In practice, there are situations where no (or incomplete) ground truth is available. In this case it is not possible to compute recall because the total number of relevant documents, $|\{\text{relevant}\}|$ in Equation (2.22), cannot be computed. In such situations only precision can be computed for a given result set. An adequate performance measure is *precision at n* (short: prec@n) which is the portion of relevant documents in a result set of size n. Information about the relevance of result objects can be obtained by user assessments.

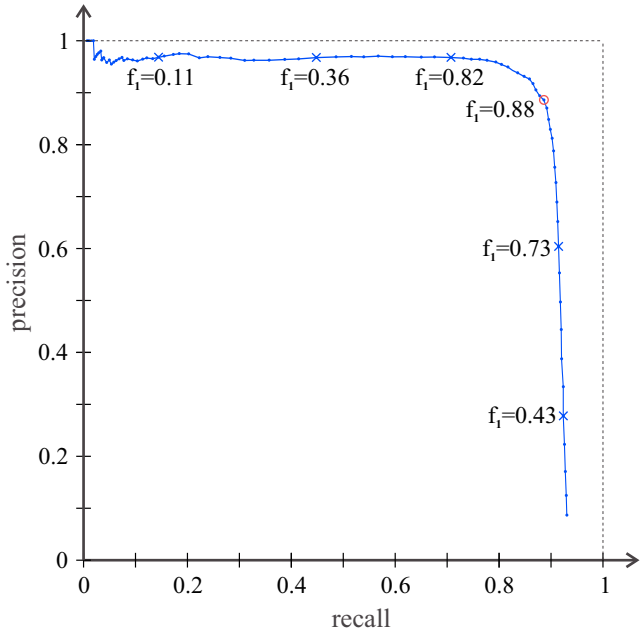


Figure 2.17: A typical recall-precision graph that illustrates the tradeoff between recall and precision. Along selected points of the curve (marked with an "x") the corresponding f_1 values are provided. The point with the highest f_1 value ($f_1 = 0.88$) is encircled.

Prec@n enables the evaluation of the *ranking* obtained by a retrieval system. From a good ranking we expect, that the most relevant documents have the highest rank. This means, that for a good ranking the prec@n values start with 1 for $n = 1$ and then slowly decrease monotonically with increasing n. The slower prec@n decreases the better is the performance and the ranking of the retrieval system.

Chapter 3

Archive Film Material

The archive film material investigated in this thesis has not been subject to automatic analysis and retrieval so far. The film material is challenging for automatic analysis due to its sophisticated stylistic properties and due to its low material quality. In this chapter, we first review important stylistic aspects of the films in Section 3.1. Next, we analyze the state of the film material and overview the contained artifacts and their effects on automatic analysis in Section 3.2. Finally, in Section 3.3 we investigate automatic film restoration in the context of the archive film material.

3.1 Background

The films under investigation are characterized by sophisticated stylistic attributes that were highly innovative for the early years of filmmaking. The films stem from the soviet filmmaker Dziga Vertov (born as David Kaufman on January 15, 1896 in Bialystok, Poland, died February 12, 1954 in Moscow, Russia) [215] who is famous for his highly formalized style of filmmaking. The first films edited by Vertov are newsreel series for the soviet regime ("Kino-Nedelya", 1917 and "Kinopravda", 1922). The newsreel series show political, social and economic events of the time and have a purely documentary character, which means that the series contain hardly any staged scenes. Different topics are usually separated by intertitles which give contextual information (necessary due to the absence of sound). In later series the style of montage becomes increasingly experimental and Vertov progressively neglects the narrative structure.

After the newsreel series Vertov increasingly focuses on the production of feature-film length documentaries. Thereby, Vertov strictly avoids narrative elements and links shots solely by semantic relationships. Additionally, he experiments with different stylistic devices and artificial effects. The result are artistic and experimental documentaries that were revolutionary for that time of filmmaking. The most important stylistic devices (in the context of this thesis) are reviewed in the following. A more comprehensive presentation of Vertov's work and his way of filmmaking is provided in [215].

The documentaries by Dziga Vertov are characterized by sophisticated and repeatedly used *visual compositions*. Visual compositions comprise the spatial arrangement of objects and camera perspective. Typical composition types are diagonal composition, symmetric composition, and compositions achieved by unusual camera perspectives, e.g. showing a train passing by from below by a camera mounted between the rails. Details and examples of typical visual compositions in the films are provided in Chapter 10.

Vertov experimented with innovative techniques to achieve artificial compositions and special effects. The filmmaker frequently uses *multi-image* compositions (e.g. split-screen) and *multiple exposure* effects to merge several images together and to establish a semantic relationship between them. Examples of multi-image and multiple exposure compositions are shown in Figures 3.1(a) and 3.1(b). The multi-image frame in Figure 3.1(a) shows four different images joined together: a piano player and three shots of ballet dancers. The multiple exposure frame in Figure 3.1(b) shows the cameraman with his camera in a glass of beer. Vertov further employed the *stop-motion* technique to animate real objects as well as hand-drawn cartoons and intertitles. Figure 3.1(c) shows an example of an animated cartoon.

Additionally to visual composition, Vertov experimented extensively with motion in his films. Both, camera and object motion and the interaction of both are used to create complex *motion compositions*. Typical examples in Vertov's films are the motion of machines and parts of machines (pistons, cogwheels, etc.) and typical activities of workers (e.g. hammering, sawing, drilling). The captured motions are often repetitive (cyclic, up/down, left/right) and are often emphasized by simultaneous rhythmical (partly contrapuntal) camera motions. Additionally, Vertov shows chasing scenes from

(a) multiple images (b) multiple exposure (c) animated cartoon

Figure 3.1: Examples for different artificial effects in Vertov's films.

different perspectives and shaky hand-camera shots to increase motion intensity. See Chapter 9 for an investigation of the retrieval of motion compositions.

A further stylistic device of Vertov is the manipulation of the *temporal axis*. Vertov shows scenes in forward and reverse order, repeats scenes, and uses fast motion and slow motion. Additionally, Vertov employs the *freeze frame* effect where single frames are unexpectedly frozen for a particular amount of time.

Vertov puts special attention on the transitions between successive shots. The filmmaker employs a wide range of different gradual transitions (dissolves, wipes, etc.) to create smooth transitions between shots. Examples of different types of gradual transitions are shown in Chapter 6. Additionally, *match cuts* are employed to smooth transitions between shots. Vertov employs *form cuts* where similarly shaped objects appear in two successive shots (see Figure 3.2 for an example), as well as *matching motion*, where the motion direction between two shots is kept consistent (see Section 9.4) [20]. On the contrary, Vertov sometimes deliberately violates continuity rules, for example by the integration of *jump cuts* which yield a discontinuity in motion which appears unexpected for the viewer.

Vertov strictly opposed any narration in his films. He used visual (and later also auditive) motifs that repeatedly appear in a film to introduce structure. Usually, each motif has an associated semantic meaning. Typical motifs are for example power poles and reservoir dams of hydroelectric power plants (as a symbol for electricity and thereby a symbol for economic and social development and prosperity), and factory chimneys (as a symbol for industrial progress). Figure 3.3 shows four examples of the typical power pole motif. The first three examples in Figures 3.3(a)-3.3(c) are additionally framed by a superimposed iris mask. The fourth example in Figure 3.3(d) is a multiple

(a) shot 1 (b) shot 2

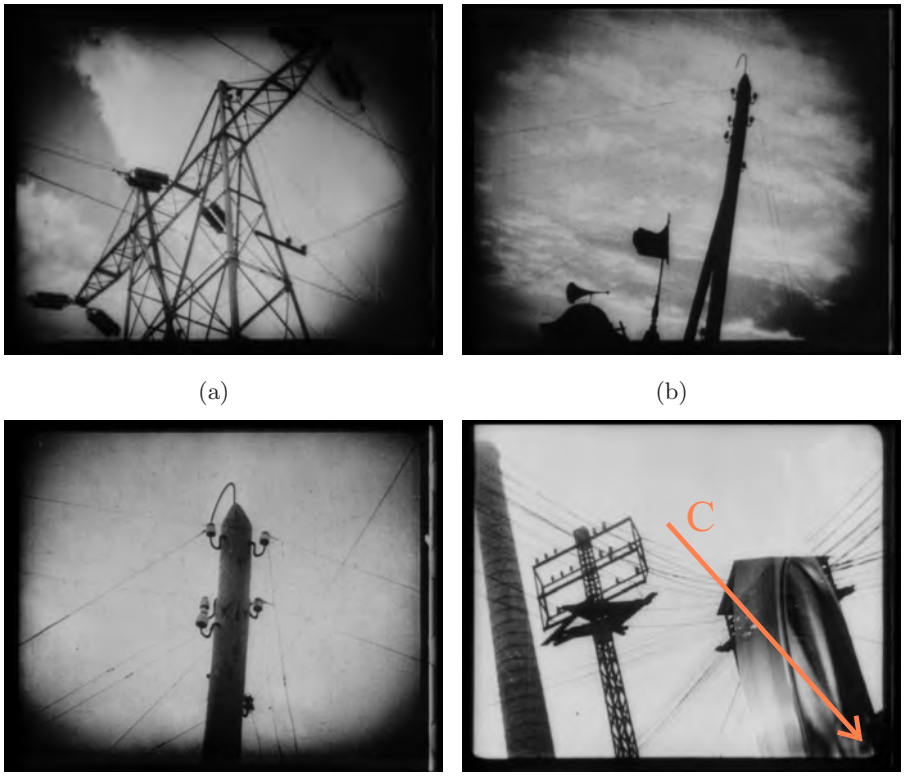
Figure 3.2: Keyframes from two successive shots which are connected by a form cut. Note that the two shots originate from different scenes and show different objects and people.

exposure shot. A flywheel is superimposed inside the tower at the right side of the frame (the arrow labeled "C" points at the center of the wheel). In all four examples we observe that the similarity between the instances of the motif exist mostly on a semantic level.

With the establishment of sound film, Vertov incorporated also auditory motifs. Typical auditory motifs are church bells and work sirens. Vertov experimented with the co-occurrence and correlation of visual and auditory motifs to convey sophisticated messages to the viewer, see Chapter 8 for an example.

The montage applied by Vertov has highly experimental character. Vertov employs *accelerated montage* where the shot frequency is increased successively. In such sequences the shot length is decreased down to a few frames or even only a single frame. Subsequent extremely short cuts visually merge and create a special type of flicker. Such sequences are especially challenging for shot cut detection (see Chapter 5). Vertov further inserts completely black frames into sequences at regular intervals to create rhythmic patterns. Chapter 4 shows an example for such a sequence.

The montage style of Dziga Vertov has a very systematic and formalistic character. The filmmaker arranges shots of different motifs systematically over time. The result are parallel montages where the visual motifs alternately appear in repetitive patterns. For one such sequence a plot sketched by Dziga Vertov (see Figure 3.4) has been preserved that shows how the filmmaker conceptualized the sequence. Time is represented horizontally in the plot. Each row represents a motif and each column represents a



(a) (b)

(c) (d)

Figure 3.3: Different instances of the power pole motif.

shot. The entries in the cells represent the duration of each shot (in number of frames). The right-most column contains the sum of frames for each motif in the entire sequence. The plot indicates that both, the temporal composition of the motifs and their corresponding shot lengths have been systematically determined [215]. Keyframes of the first 15 shots of the sequence are shown in Figure 3.5. The sequence shows the hissing of a flag and alternately shows the flag and different faces that observe the hissing. The sequence is consistent with the plot in Figure 3.4, except for a few shots at the beginning that are missing (maybe lost due to film tears).

The distinct composition and montage of Vertov's films make them a challenging material for automatic film analysis. The large variety of employed stylistic devices and the strong formalistic structure raise novel requirements for automatic film analysis and retrieval. However, the state of the material is challenging for automatic analysis, as well. Due to their old age the films contain numerous artifacts that interfere with au-

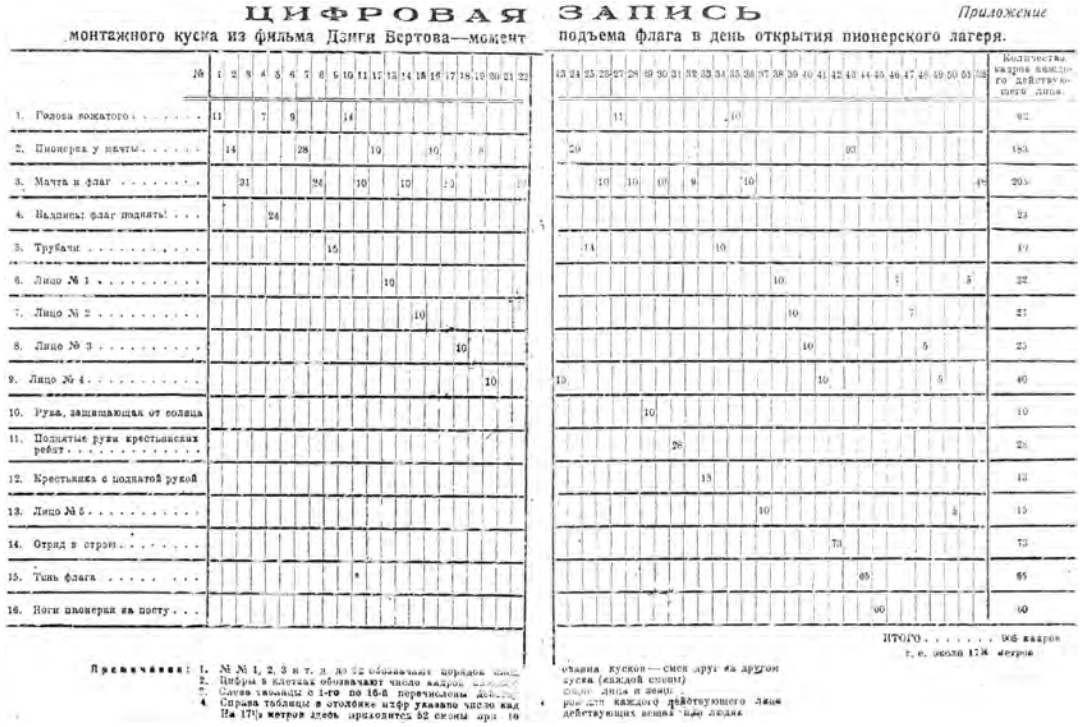


Figure 3.4: The visual representation of the montage of a sequence [17].

tomatic analysis methods. These artifacts impede the automatic analysis and retrieval of the films. We discuss the different artifacts present in the historic material and their impact on automatic analysis in the following.

3.2 State of the Material

The archive films investigated in this thesis have been produced in the Soviet Union in the 1920s and 1930s by Dziga Vertov. Most of the films are silent, only the late films contain a sound track. The original material is 35mm black-and-white film made of cellulose triacetate (see Figure 3.6). The films are played at non-standard frame rates of 18 to 21 frames per second for silent film and 25 frames per second for sound film. In sound films, the soundtrack has been optically stored on the filmstrip (sound-on-film technique [80]) alongside the visual content of the frame, see Figure 3.7.

An overview of the available archive films is given in Table 3.1. For two of the films different versions exist. For "Enthusiasm" there is a slightly varying restored version of the original version [215]. For "Man with a Movie Camera" there exist a version from

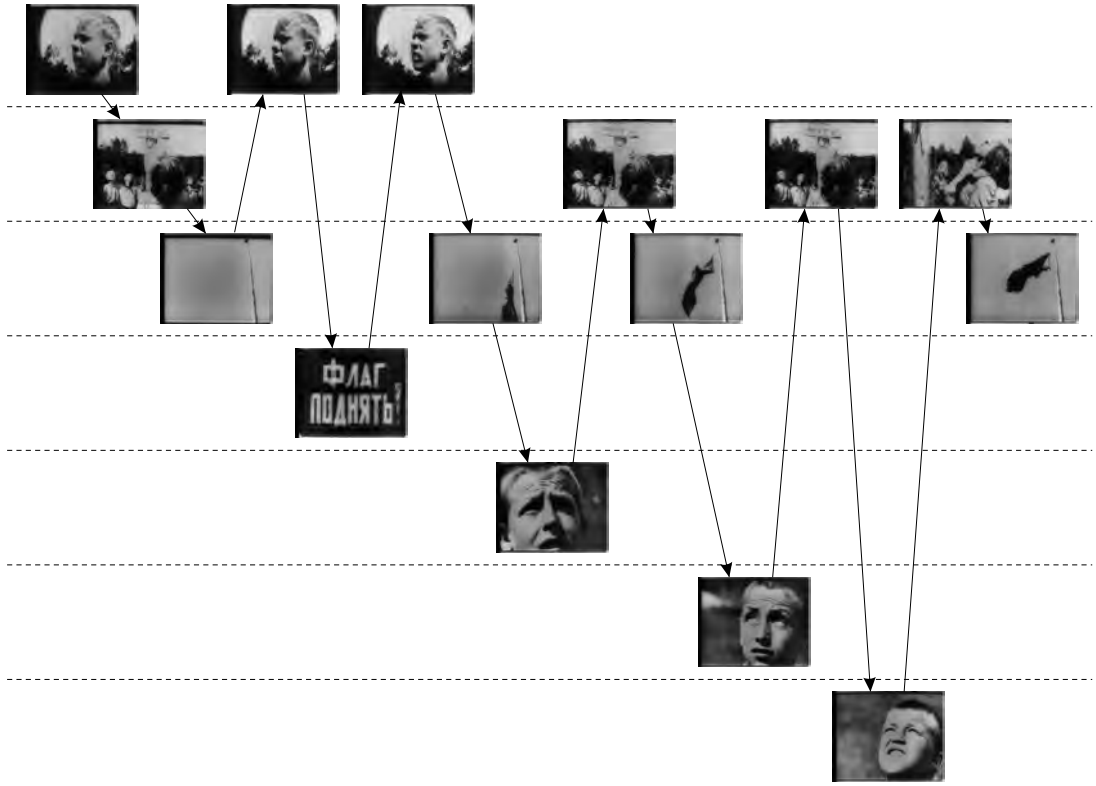


Figure 3.5: Keyframes of the first 15 shots of the sequence that shows the hissing of a flag. Each row shows one of the motives that are specified in the plot in Figure 3.4.

the Vertov Collection of the Austrian Film Museum in Vienna (V) and a version from Amsterdam (A), with best thanks to the EYE Film Institute Netherlands (Mark-Paul Meyer).

Prior to automatic analysis the analog filmstrips have been digitized. For this purpose we scan the films frame-by-frame in PAL quality (720x576 pixels) and with 256 gray values (8 bit). The result of digitization is an image sequence that represents each frame of a film. The frame-by-frame digitization avoids the introduction of interpolated frames which usually occur during digitization at standard frame rates (e.g. 25 fps for PAL) when the projected film is for example captured by a digital camera. Since interpolated frames represent information that does not exist in the original material they would interfere with automatic analysis. It is crucial to note that Vertov employed the exact number of frames for a given shot as a stylistic device (as can be observed from the plot in Figure 3.4 in Section 3.1). Thus interpolated frames would tamper

Figure 3.6: A 35mm silent film strip [17].

Film Title	Fps	Duration	#Frames	#Shots	Sound?	Abbrev.
Kinoglaz	18	01:17:54	84132	1304	no	KGLZ
Kinopravda 21	18	00:32:27	35060	413	no	KP21
Stride Soviet!	18	01:12:28	78272	1110	no	SRSV
A Sixth of the World	18	01:04:03	69182	1017	no	6thW
The Eleventh Year	18	00:58:26	63123	660	no	11th
Man with a Movie Camera (V)	18	01:28:40	95768	1782	no	MMCV
Man with a Movie Camera (A)	18	01:26:47	93743	1781	no	MMCA
Enthusiasm (Original)	25	01:04:44	97116	604	yes	EsmO
Enthusiasm (Restored)	25	01:04:45	97134	612	yes	EsmR
Three Songs of Lenin	25	00:59:20	89023	817	yes	3SoL
Schatten der Maschine[1]	18	00:23:43	25622	420	no	SdM

Table 3.1: Films of Dziga Vertov analyzed in this thesis in chronological order.
[1] The film "Schatten der Maschine" is a compilation film by Victor Blum and reuses content produced by Dziga Vertov

with the intended statements of the films [215]. Additionally, we skip compression to avoid the introduction of additional artifacts in the digitized stream. If an audio track is available in a film, the track is scanned as well and converted into an uncompressed PCM coded file.

The provided filmstrips are multiple-generation copies that were never intended to be used for other purposes than backups. Due to this fact, these copies were not handled with much care in the film archives. Today, the original filmstrips do not exist anymore, hence the available backup copies are the only existing source material left. The state of the material has degraded significantly, during storage, copying, and playback over the last decades.

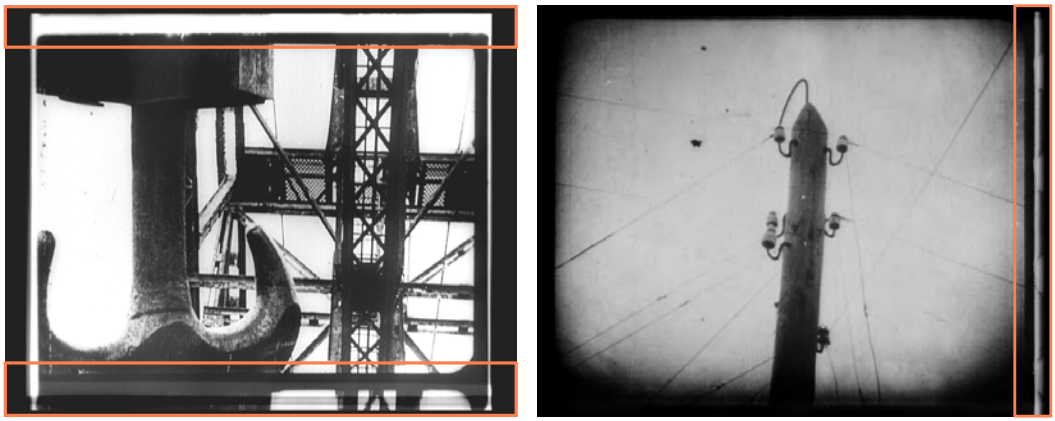
(a) (b)

Figure 3.7: Two examples of sound-on-film in historic material from the film "Enthusiasm". The waveform of the sound is optically stored at the left side of the frame [17].

The filmstrips are made of organic material (cellulose triacetate). Storage over the last decades resulted in shrinking of the filmstrips due to chemical processes in the base support. Thereby, the filmstrips physically contracted horizontally and vertically. Shrinking results in frame displacements and non-linear geometric distortions. Due to vertical shrinking the framelines (the area between two successive frames on the filmstrip) become visible and sometimes also the content of the next frame, see Figure 3.8(a). The shrinking in horizontal direction is usually less disturbing. However, if the horizontal shrinking exceeds a certain limit the perforation of the filmstrip becomes visible in the frame, as in Figure 3.8(b). Additionally, the filmstrips were often stored under suboptimal conditions. As a result, mold and humidity harmed the filmstrips during the long time of storage. See Figure 3.9 for distortions originating from mold and humidity.

The archive films are multiple-generation copies. When copying is performed under suboptimal conditions dirt and dust is copied into the films. With each generation of copy dirt, dust and previously existing artifacts (e.g. scratches, blurred images) accumulate which results in a broad spectrum of distortions, see Figure 3.10. Additionally, copying leads to a degradation of contrast with each generation of copy. The result are low-contrast images as shown for example in Figure 3.9(b).

When the filmstrips are not exactly aligned to each other during copying the frames of the original filmstrip are incorrectly mapped to the frames of the new filmstrip. The result are frame displacements and visible framelines. For examples see Figures 3.9



(a) vertical shrinking (b) horizontal shrinking

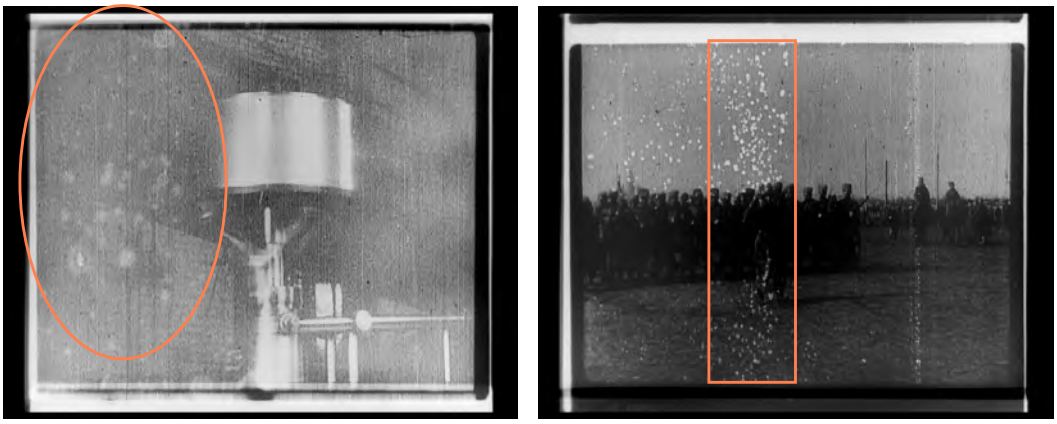
Figure 3.8: The effect of the shrinking of filmstrips. (a) due to horizontal contraction of the filmstrip framelines become visible at the top and the bottom of the frame, as well as image content of the next frame (at the bottom). (b) horizontal shrinking causes the perforation of the filmstrip to become visible (at the right side of the frame)

and 3.10 where all frames suffer from these artifacts. The frame displacements from copying together with the displacements caused by film shrinking (see Figure 3.8) introduce a significant *shaking* in the films.

Further artifacts have been introduced during playback of the films in old projectors. Dirt present in the mechanics of the projectors (in the film transport) introduce vertical scratches that cover many subsequent frames. Vertical scratches are shown for example in Figure 3.10(e).

Playback and presentation of the films bear the risk that a filmstrip tears. Each time, a filmstrip tears a few frames of the strip are destroyed. When the film is then glued together the absence of the destroyed frames produces abrupt jumps (unintended jump cuts) in the movies. Additionally, the splice becomes a visible artifact at the position where the filmstrip is glued together. An example of a film tear is shown in Figure 3.11.

Additionally to artifacts from storage, copying and playback the films suffer from limitations of the recording technique of the early 1920s and 1930s. In the early years of filmmaking, the film transport was controlled manually. For this purpose, the cameraman moved a crank at one side of the camera to move the filmstrip and to control the shutter of the camera, see Figure 3.12. The manual film transport yields variations in



(a) (b)

Figure 3.9: Artifacts originating from liquids and mold.

the frame rate across the filmstrip. As a consequence the exposure along the filmstrip varies which in turn generates frames of different brightness. The resulting effect during playback of the films is strong and fast alternating flicker as shown in Figure 3.13.

The artifacts in the historic films impede the automatic analysis. We distinguish between three different classes of artifacts in the context of automatic analysis: global artifacts, local artifacts and temporal artifacts. *Global* artifacts influence the entire area of a frame and comprise shaking, flicker and low-contrast. Shaking is most disturbing in motion analysis where motion vectors between pixels in the image have to be computed reliably. Due to shaking the resulting motion estimates are noisy and often the tracking of motion is not possible over longer time spans. Together with the complex compositions of camera and object motion present in the films the analysis of motion is challenging for the material. We investigate motion analysis and the retrieval of motion compositions in Chapter 9.

Flicker is problematic since it influences the overall brightness of frames and impedes similarity comparisons between frames. Similarity comparisons between frames is an essential part in most visual retrieval tasks, such as shot cut detection and scene segmentation. A common practice is to compare frames based on color and intensity histograms (see Sections 2.3.3 and 2.3.4). However, such comparisons are not robust in the presence of flicker because flicker distorts the histograms globally. Generally,

(a) wrong exposure (b) dirt (c) scratch

(d) a fingerprint (e) dirt, vertical scratches (f) blurring

Figure 3.10: Artifacts introduced from (repeated) copying (dirt, scratches, and distortions in brightness, blurring).

image descriptors that rely on intensity information (brightness) are not suitable for the representation of the archive film material due to the heavy flicker.

The same as for flicker applies to low-contrast images. Again image descriptors that rely on brightness are not suitable because they do not capture distinctive information from low-contrast frames. Additionally, the extraction of local image descriptors (see Section 2.3.7) is difficult due to a lack of distinct feature points in low-contrast images. Similarly, the extraction of local image descriptors is problematic in blurred images which lack in distinct structures necessary for the identification of feature points.

Additionally to global artifacts, there are *local* artifacts that affect only a part of a frame's area. Local artifacts comprise visible framelines and frame borders (e.g. perforation), scratches, dirt, dust, and artifacts from liquids spilled over the filmstrip and mold. Local artifacts represent misleading information that disturbs automatic analysis. Visible framelines and frame borders can easily be removed by cropping the frame borders for an entire film. However, cropping with a constant offset also removes some image information since the position of the frame borders varies over time.

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

Figure 3.11: Three successive frames of a sequence. The film has teared after frame 1 and several frames are missing between frame 1 and frame 2 which results in a discontinuity in motion. In frame 2 artifacts from gluing the film together are visible in the upper part.

(a) (b) (c)

Figure 3.12: Recording with a historic movie camera. The cameraman rotates the crank manually to transport the filmstrip.

Artifacts like dirt and scratches interfere with analysis techniques that operate on small scales (small analysis windows), such as block-based image features with small block-size and local descriptors of feature points. As a consequence, the description of fine structures in the frames is prone to errors. Additionally, the local artifacts generate abrupt visual changes that interfere with temporal movie analysis, required for shot and scene segmentation.

The third class of artifacts are *temporal* artifacts which comprise distortions of the temporal axis of a filmstrip. Temporal artifacts are for example jump cuts (introduced by film tears) which introduce motion discontinuities. Such discontinuities interfere with motion analysis. Actually, jump cuts have been employed by Vertov also on purpose as a stylistic device. Today it is often not clear if a jump cuts has been intended by the filmmaker or has been introduced by a film tear.

(a) (b) (c)

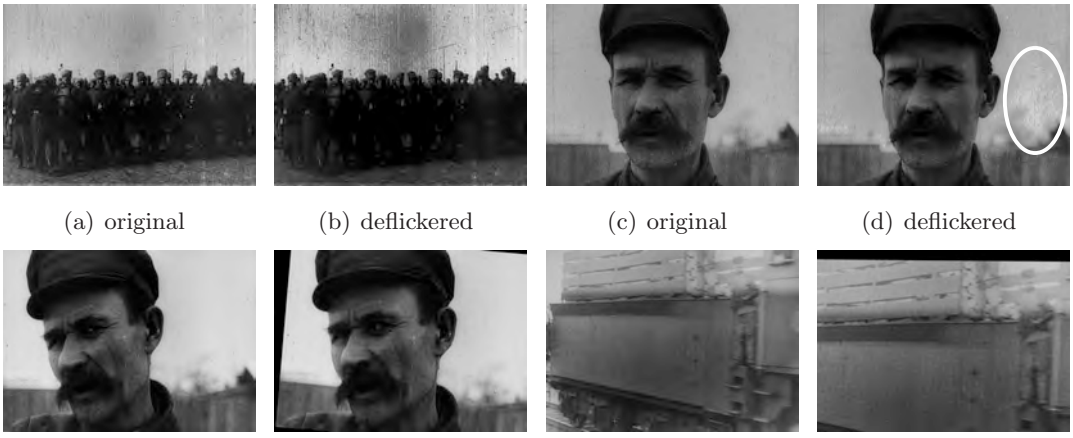
Figure 3.13: Three successive frames from a shot in "Kinopravda 21". Heavy flicker is introduced due to the manual and uneven film transport.

The historic film material and its artifacts challenge automatic analysis techniques and often make existing techniques inapplicable. A first step towards the analysis of archive film material is an automatic restoration that removes the most disturbing global and local artifacts.

3.3 Restoration

A straight-forward approach to improve the quality of historic film material is to apply professional software for its restoration. However, professional software for film restoration is expensive and usually requires human interaction or supervision [188]. This makes high-quality restoration of large amounts of films costly and often not feasible.

A cost-effective alternative to professional film restoration are algorithms that filter specific artifacts fully automatically. We exemplarily explored algorithms for deflicker, noise reduction, and image stabilization to remove flicker, dirt, and shaking. One might expect that such preprocessing improves the following content-based analysis and retrieval. In fact, most methods reduce the corresponding artifacts. However, they introduce new artifacts which are often more disturbing than the original ones. For example, the employed *deflicker* method based on histogram alignment [167] significantly dampens the brightness variations across the frames but fails when the brightness variations exceed a certain level resulting in contrast distortions as depicted in Figures 3.14(a)-3.14(d).



(a) original (b) deflickered (c) original (d) deflickered

(e) original (f) stabilized (g) original (h) stabilized

Figure 3.14: Artifacts that originate from automated restoration. 3.14(a) and 3.14(c) show keyframes of two sequences where the deflicker filter does not work correctly due to large brightness variations. In 3.14(b) noise is emphasized (mostly in the sky) and in 3.14(d) noise is introduced in the background. 3.14(e) and 3.14(g) show keyframes of sequences where stabilization fails. 3.14(e) shows a man who turns his head. The stabilizer fails to compensate for the object motion resulting in an unwanted rotation of the frame in 3.14(f). 3.14(g) shows a train passing by. Since there is hardly any static background, the stabilizer fails to align the images and falsely translates and scales up the frame in 3.14(h).

We reduce *noise* by a temporal median filter. This removes most scratches and dirt but cancels out image details which are necessary for subsequent analyses, e.g. detection and tracking of feature points for motion analysis, see Section 9.2.

Stabilization aims at removing shaking from a sequence which is caused by repeated copying and film shrinking. The challenge is to remove shaking independently from the *intended* camera and object motions. Stabilization methods work well for scenes with small moving objects or smooth camera operations [214]. In scenes with large moving objects or fast and non-uniform camera motion, stabilization methods often confuse unintended shaking with the intended motion. This behavior leads to unexpected results such as rotation and unwanted warping of the frames (see Figures 3.14(e)-3.14(h)).

We observe, that fully automated algorithms introduce new artifacts and remove detail information that is necessary for automated processing. From these observations, we conclude that such an automated preprocessing is not advisable with the investigated archive film material and that human interaction would still be required. In the course

of our investigations, we observe that most retrieval tasks are influenced by a specific type of artifacts only or a subset of types. We conclude to skip automated preprocessing and instead aim at the development of retrieval methods that are robust to the actually relevant artifacts.

Chapter 4

Detection of Black Frames and Intertitles

As a first step in automatic analysis, we aim at the detection of two lower-level concepts that are especially important in archive film, namely *intertitles* and *black frames*. Both these concepts have specific structural and semantic properties relevant for understanding the films. First, we discuss properties of black frames and intertitles for archive and contemporary films in Section 4.1. Next, we present a method for the detection of black frames that is robust to the artifacts in archive film material in Section 4.2. In Section 4.3 we investigate the detection of intertitles in archive film material. We propose appropriate features for intertitle detection and show that this task can be solved reliably. The investigations in this chapter show that recognition tasks that appear to be trivial may become more complex due to the sophisticated style and the artifacts of the archive films.

4.1 Introduction

At first glance the automatic identification of black frames and intertitles is trivial. However, in presence of the numerous artifacts in the investigated archive films (see Section 3.2) both tasks become challenging for automatic analysis. We perform two preliminary investigations for the retrieval of black frames and intertitles that illustrate the influence of the archive film material on retrieval tasks.

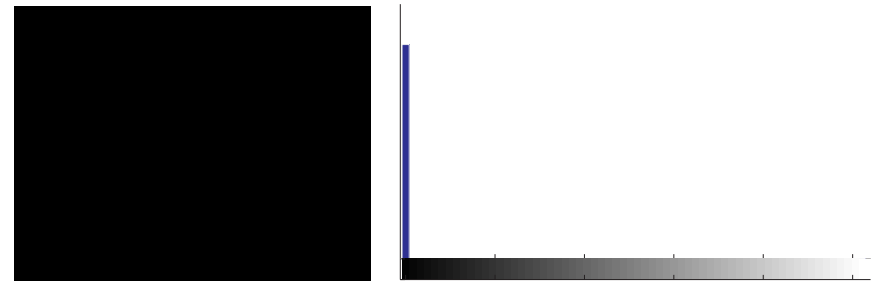
Figure 4.1: Black frames cut in-between a series of frames showing rail tracks. The black frames visually evoke the otherwise auditory impression of passing over expansion joints.

Intertitles are a classic element of early filmmaking, a time when no soundtrack was available. Intertitles contain additional descriptions not depicted on the screen: They introduce characters, locations and provide temporal context. Additionally, intertitles provide temporal structure by separating different topics in archive documentaries. Today, intertitles are seldom used, however they did not vanish completely. They are mostly applied as an artistic means, for example in "Pulp Fiction" by Quentin Tarantino and Jim Jarmusch's "Ghost Dog - The Way of the Samurai" where intertitles introduce new scenes. Intertitles are also used in television drama series like the "Law & Order" franchise. Modern intertitles usually contain a small number of colors and, thus, are easy to detect. The detection of intertitles forms the basis for further investigations such as optical character recognition and keyword extraction.

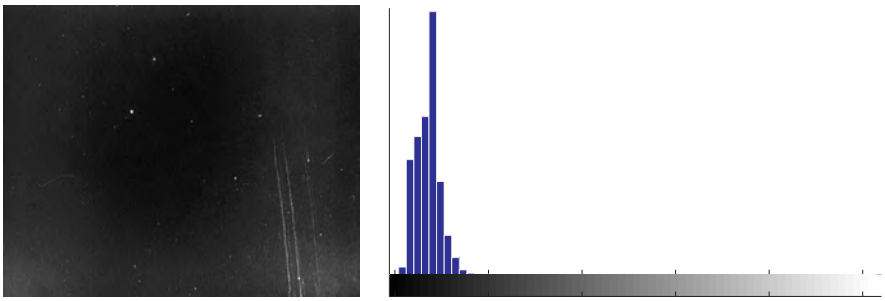
Black frames, as the name implies, are entirely black frames usually found at the beginning of fade-ins and at the end of fade-outs. Additionally to these technically motivated uses, they are applied as an artistic means. A widely known sequence where black frames are used as an artistic means shows a train ride where the director alternately shows shots of the rail track and black frames (see Figure 4.1). This should evoke the impression of the train passing over expansion joints (responsible for the "clickety-clack").

4.2 Detection of Black Frames

The detection of black frames seems to be a trivial task. Since they are monochrome and black, their mean gray value should be near zero while the variance of the gray values should be minimal. However, this is only true for high quality material as depicted in Figure 4.2(a). In archive film material containing flicker and degraded contrast, black frames often do not contain any black pixels at all (see Figures 4.2(c)-4.2(d)).



(a) blackframe from modern film (b) histogram of blackframe from modern film



(c) blackframe from archive film (d) histogram of blackframe from archive film

Figure 4.2: Black frames from archive and modern film material and their respective intensity histograms. Black frames from archive film material often do not contain any black pixels at all.

Thus, we devise a method based on higher-order statistics of the gray value distribution. We extract three features for black frame detection:

- the centroid of the intensity histogram,
- the variance of the frame's gray values, and
- the rolloff point of the gray value distribution (the 95% percentile of the distribution).

These higher order moments of the gray value distribution are indicators of the "blackness" and the monochromaticity of the frame. The rolloff point of the gray value distribution is itself a gray value which is low for black frames because most of the pixels are dark. In a non-black frame the rolloff point is higher because they contain more bright pixels. Similarly, the centroid of the intensity histogram is a gray value derived from the distribution of gray values in the frame. It is lower for black frames

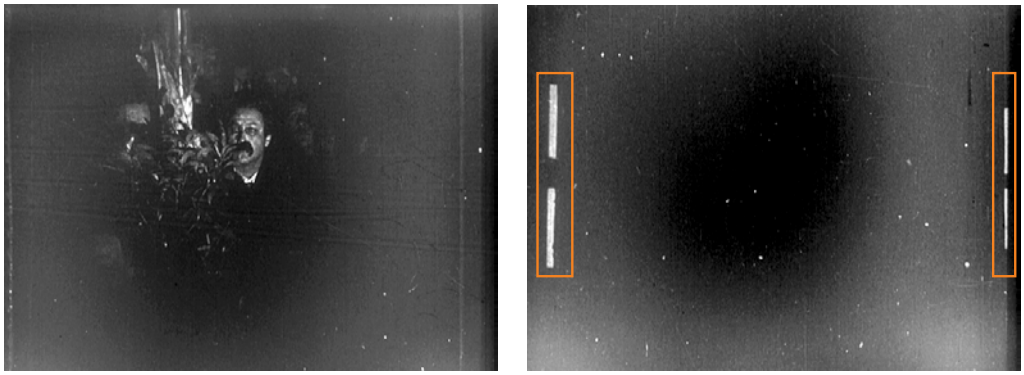
and higher for non-black frames. The variance of the frame's gray values reflects the number of distinct gray values in the frame. In summary, black frames have a low variance of the gray value distribution, because they have mostly the same gray value and they have a low value for the centroid as well as for the rolloff point because most of the pixels are dark.

We detect black frames by thresholding the three extracted feature values. For this purpose, we experimentally determine values for the thresholds. Since it is a requirement by the film scientists that the likelihood of missing a black frame is low, we perform threshold determination in a way that prefers recall over precision. A frame is classified as a black frame if the values for all three features stay below the respective thresholds. The threshold for the centroid is 50, for variance 0.0007 and for rolloff 45.

This approach allows for a reliable identification of black frames. The method achieves 93% recall and 55% precision in a set of 35060 frames from the film "Kino-pravda 21". We observe that precision is low compared to the recall because there is a large number of false positives. Figure 4.3 depicts two false positives. Closer inspection of the two examples reveals that Figure 4.3(a) shows a very dark frame that is falsely detected by the proposed method. Figure 4.3(b) depicts a frame from an animated sequence. The animated sequence starts with a black frame. Then white rectangles are introduced on the left and right side of the image. The rectangles grow from the center in vertical direction to become vertical bars. All the frames from the beginning of this animated sequence are recognized as black frames because they contain only a very small number of bright pixels.

4.3 Detection of Intertitles

Intertitles commonly show monochromatic text on monochromatic background. As a consequence, we expect them to have a specific gray value distribution which is bimodal. For contemporary material this expectation is usually met as can be observed in Figures 4.4(a) and 4.4(b). In archive film material however the gray value distribution of intertitles is usually not bimodal. An example is given in Figures 4.4(c) and 4.4(d)) where we observe no distinct peaks for foreground and background in the histogram. The foreground (the text) and the background gray value distributions even overlap.



(a) a very dark frame showing a person (b) a frame from an animated sequence

Figure 4.3: False positives returned by the black frame detection algorithm. (a) shows a dark frame depicting a person that is falsely identified as a black frame. (b) depicts a frame from an animated sequence in which the highlighted bright bars starting from the center grow in vertical direction. Frames from this sequence are identified as black frames until the bars have grown to a specific size. Note that in both (a) and (b) the frames are significantly brightened to make them recognizable.

We propose a robust method for the detection of intertitles based on edge- and intensity histograms. Feature extraction is performed in two passes, first we extract features for single frames and second we aggregate features over entire shots. Two features are extracted for each frame:

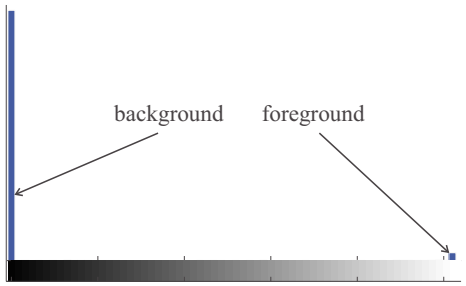
- a global MPEG-7 edge histogram and
- local intensity histograms with 128 bins where the image is uniformly split into 9 image blocks.

In the second pass four features are extracted for each shot: First, we compute the mean edge histogram over all frames of the shot. This feature reflects the average number of edges in a shot which is usually high in intertitles due to the displayed text.

Second, we compute the variance of all edge histograms across a shot. The resulting *variance histogram* reflects the temporal variance of the single histogram bins throughout the shot. The sum of this variance histogram represents the second feature. For intertitles which are mostly static, this sum of variances should be low.

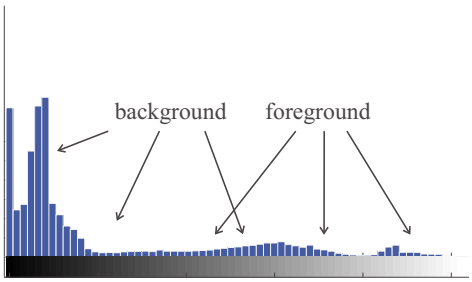
Next, we compute the variance of the block-based intensity histograms (extracted in the first pass). The resulting variance histograms (one for each block) are concatenated

(a) an intertitle from the contemporary film "Pulp Fiction".



(b) histogram of the modern intertitle.

(c) an intertitle from archive film.



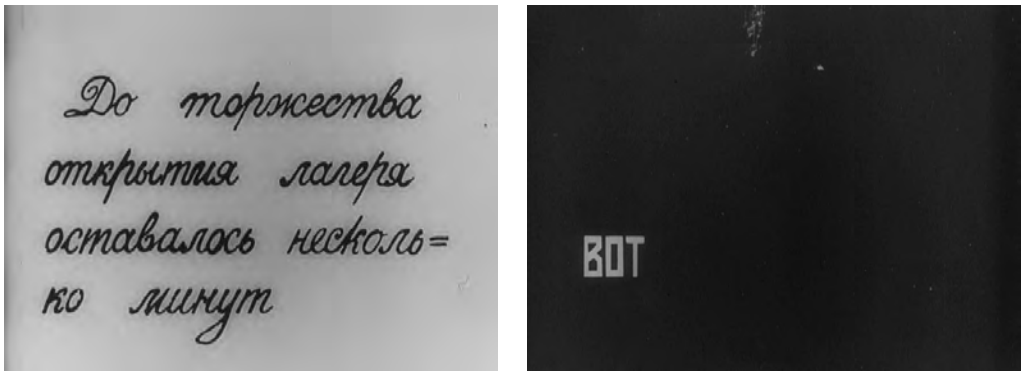
(d) histogram of the intertitle from the archive film.

Figure 4.4: Intertitles from archive and modern film material and their corresponding intensity histograms. In contrast to archive film material, the gray value distribution of intertitles in modern films usually has two distinct peaks which can be detected easily. For historic material this does not apply.

and summed. The resulting scalar value is the third feature and represents the intensity variation over time which should be low for intertitles.

Finally, we extract the fourth feature that measures the "bimodality" of the intensity histograms. For this purpose, we compute a mean intensity histogram over a shot. We approximate the mean histogram with a spline to smooth it and to remove minor peaks. The maximum distance between any two peaks in the spline represents the fourth feature. Ideally, intensity histograms of intertitles are bimodal and look like in Figure 4.4(b)). The feature measures how close the current shot's intensity histogram resembles an ideal one.

The four features are concatenated into an eight-dimensional feature vector and form the input of a support vector machine with a polynomial kernel of second order. The proposed method is able to achieve a recall of 95% and a precision of 81% with a



(a) (b)

Figure 4.5: Two intertitles from the film "Kinoglaz". The proposed method is able to identify both correctly as intertitles despite their visual differences and the artifacts.

test set that contains 1716 arbitrary (non-intertitle) shots and 167 intertitles from the films "Kinopravda 21" and "Kinoglaz". Figure 4.5 depicts two intertitles the method successfully detects. We observe that the method is suitable for the detection of diverse intertitles even if artifacts are present.

4.4 Summary

We have presented two preliminary investigations for the detection of two basic concepts from archive film: black frames and intertitles. Both concepts have specific relevance in archive film and both advocate the need for special methods targeting archive film. In modern film the detection of these concepts is much simpler, even trivial while in archive film effort has to be invested in order to detect them sufficiently well.

For the detection of black frames we devise a method based on higher order statistics of the gray value distribution. We employ content-based features that capture the darkness and monochromaticity and use thresholds for the final decision. With this method we are able to reliably identify black frames even if the film strip's quality is degraded. We identify intertitles using features that capture edge and intensity information. We perform feature extraction in two passes. In the first pass, we extract features for all frames. In the second pass, we aggregate the frame-based features over

entire shots. The features target specific properties that distinguish intertitles from other types of shots. We achieve satisfactory detection results with an SVM classifier.

In this chapter we have implicitly worked with shot boundary information for the detection of intertitles. In the concrete case shot boundaries were derived from manually created ground truth. This manual identification of shot boundaries is time consuming and error-prone and motivates automatic shot boundary detection techniques. In the next chapter we present such an automatic shot boundary detection technique.

Chapter 5

Detection of Shot Cuts

Shots are the basic units of film and represent a fundamental lower-level concept for the construction of films. The detection of shot boundaries is the basis for most higher-level investigations and analyses. This chapter focuses on the detection of abrupt shot cuts which are the most common type of shot boundary while Chapter 6 addresses the detection of gradual transitions. In this thesis, we incorporate shot boundary information in the segmentation of a film into scenes in Chapter 7 and in the extraction of synchronous montage sequences in Chapter 8. Furthermore, the retrieval of motion composition and visual composition in Chapters 9 and 10 is performed on previously extracted shots. Additionally to these applications, shot boundaries provide information on the length of shots and allow the analysis of montage patterns (see Section 3.1) which are typical for the film material under consideration [246]. In this chapter, we present a method for the detection of shot cuts that is designed with historic archive film in mind. We identify shortcomings of existing shot boundary detection techniques in Section 5.1. In Section 5.2 we review existing techniques that are either developed for, or applicable to, archive film. We propose a robust shot boundary detection technique in Section 5.3. Section 5.4 provides an extensive evaluation of our method and performance comparisons with state-of-the art techniques.

5.1 Introduction

A shot in an edited film is defined as "the length of film from one splice or optical transition to the next" [20]. Shots can be bounded by either abrupt shot cuts (which

originate from joining two film strips by a splice) or by a gradual transition (by using multiple exposure techniques and masks). The segmentation of a film into its shots is performed by detecting the shot boundaries between the shots. In the following, we identify shortcomings of existing shot cut detection approaches in the context of the investigated archive film material.

First, the film material is black and white. Most state-of-the-art shot boundary detection algorithms incorporate color information, e.g. in TRECVID 2006 15 out of 19 shot boundary detection algorithms relied on color and only four used intensity information [198]. In TRECVID 2007 only three out of eleven algorithms were based on grayscale information. In the context of historic archive film (which usually is black and white) it is not safe to assume that color-based shot boundary detectors are applicable.

Second, most state-of-the-art methods have been developed and evaluated in the context of high-quality video material, such as TRECVID data. Archive film material contains numerous artifacts that interfere with shot cut detection. Dirt, dust, liquids (spilled over the filmstrips), and scratches introduce noise that generates abrupt visual changes. These unintended changes interfere with established shot cut detection algorithms that are based on pixel differences, edges, and corners (feature points). Additionally, frame displacements disturb techniques that rely on motion information. Flicker globally influences the distribution of intensity values, which limits the power of histogram-based approaches. Kopf et al. discuss several other issues in the context of the analysis of old films [113].

Third, the film material under investigation contains sophisticated montage patterns such as accelerated montage, see also Section 3.1. Accelerated montage sequences contain sequences of very short shots (down to one frame in length). This introduces problems for shot cut detection techniques which rely on larger processing windows (e.g. [46] employ processing window of 21 to 71 frames). Shot cut detectors for archive film material must rely on small processing windows to enable the detection of short shots.

From the stylistic properties and the physical state of archive film we draw the conclusion that there is a demand for the adaptation of shot cut detectors for archive film material. In this chapter, we develop a shot cut detector that takes the above mentioned shortcomings into account. The detector is an extension of the approach introduced in [46] and [47]. We integrate content-based image features that rely on

intensity information only and that are robust to most of the global and local artifacts in the archive films (see Chapter 3). Additionally, we reduce the size of the processing window to cope with the complex temporal structure of the material. Finally, we introduce a more effective fusion scheme for the features in the framework. Experiments on archive films show that the technique outperforms readily available and established shot cut detection algorithms.

5.2 Related Work

Shot cut detection is a well-investigated topic in video analysis. Extensive work has been conducted so far [28, 241]. Most state-of-the-art shot cut detectors rely on color information which may be observed from the submissions to TRECVID's 2006 and 2007 shot boundary detection tasks. Note that in 2008 shot cut detection was seen to be solved and was excluded from the TRECVID evaluation. However, we observe that only few techniques support low-quality black-and-white material. Generally, even fewer techniques target the special case of archive film. Archive film has unique properties that challenge established analysis algorithms [113]. Urhan et al. propose novel techniques for shot cut detection geared towards old film [224, 225]. Their approaches exploit phase correlation and kernel-based comparison to detect abrupt shot cuts in visually degraded and distorted films. Another work on shot segmentation in archive film material stems from Kopf et al. [113] where the authors perform summarization of historical archive films based on previously segmented shots.

A promising method for shot cut detection (and also a top performer in the TRECVID benchmark) has been proposed by Cooper and Foote [46]. The method has originally been proposed in [67] to segment musical signals and has later been extended to shot cut detection in color video [46]. The authors first compute the similarity of adjacent frames and construct a similarity matrix. Next, they analyze the similarity matrix with a specific filter in order to detect shot cuts. Color and intensity histogram features are the basis for the construction of the similarity matrix. However, these features are not applicable to archive film material where flicker and intensity variations are omnipresent. Another feature proposed in [46] are the global low-order discrete Cosine transform (DCT) coefficients of the three color channels. Again this feature relies on color and cannot be applied directly to black-and-white film material. The

feature can be adapted to black-and-white material by computing the global low-order DCT coefficients of the intensity channel which represent the global intensity distribution of a frame. Experiments show that this adapted feature does not contain enough discriminatory information for shot cut detection. We observe, that archive film material demands for features that are *robust* and at the same time represent significant *discriminatory* information.

5.3 Robust Shot Cut Detection

We extend the method by Cooper and Foote [46] for shot cut detection in several ways. First, we integrate robust and discriminatory *features* that are solely based on intensity information. Next, we apply the proposed self-similarity analysis for each feature *separately*. Finally, we *fuse* the results for each feature and apply peak detection to identify potential shot cuts.

5.3.1 Feature Extraction

We propose two features for shot cut detection that are robust against the artifacts present in archive film material. The low-frequency content of the frames is captured by a block-based discrete Cosine transform feature (bbDCT), while the high-frequency content is represented by an edge descriptor.

For the first feature (bbDCT), we uniformly split each frame into B image blocks. We transform each block into frequency domain by a DCT and extract the first N low-frequency coefficients. The coefficients of all blocks yield a $B * N$-dimensional feature vector. The parameter N should be chosen in a way that high-frequency distortions are removed. Parameter B determines the block size. It balances the influence of frame displacements and motion in a block and the amount of preserved spatial information. Large blocks lead to high robustness against frame displacements, but to a loss of spatial information (and thereby a loss of expressiveness of the feature) while small blocks lead to the opposite. The bbDCT represents the coarse intensity distribution among the blocks of the frames. It is robust against local high-frequency artifacts, such as dirt and scratches. Furthermore, it compensates for frame displacements and flicker to a high degree.

The second feature captures the orientations of the edges in the frames. Edges represent highly discriminatory information for shot cut detection. They represent semantically meaningful information, such as contours and object boundaries that usually change considerably across shot cuts [123]. We employ an edge histogram, similar to the MPEG-7 edge histogram. The edge histogram (EH) is computed for the same B blocks as the bbDCT. The histogram of each block contains five bins, for horizontal, vertical, 45 degree, 135 degree, and non-directional edges. The edge histogram for the entire frame contains $B * 5$ bins. The EH represents the distribution of orientations of the edges across the blocks of the frame. It is highly robust to frame displacements since it captures global information within in each block. Additionally, the feature is invariant to flicker. The EH captures high-frequency information which makes it prone to artifacts like scratches and dirt. The influence of these artifacts is usually low compared to the influence of the dominant and meaningful edges. However, global artifacts, such as scratches across the entire frame are reflected in the feature (see Figure 3.10(c)).

Both features, bbDCT and EH are well-suited for combination, since they capture orthogonal and thereby complementary information. The bbDCT feature represents low-frequency information, while the EH summarizes high-frequency content.

5.3.2 Similarity Comparison

In a next step we compare frames by computing the similarity between their features. This results in a similarity matrix from which we derive a function that is used as an indicator for shot cuts. We compute the similarity between feature vectors of frames similarly to Cooper and Foote [46]. First, we extract both features for each frame, resulting in a one-dimensional feature vector. Next, the pairwise similarity of all feature vectors is computed by the Cosine similarity (see Section 2.4.3).

Computing the pair-wise similarity of adjacent frames results in a (symmetric) similarity matrix with maximum values at the diagonal. An entry at position (i, j) in the matrix corresponds to the similarity of two feature vectors of two frames i and j (see Figure 5.1(a) for an illustration of similarity matrix construction). The similarity matrix represents all possible similarity comparisons between all frames under consideration.

Time progresses along the rows and the columns of the matrix, as well as along the main diagonal. Similar frames yield high values (high similarity) while dissimilar

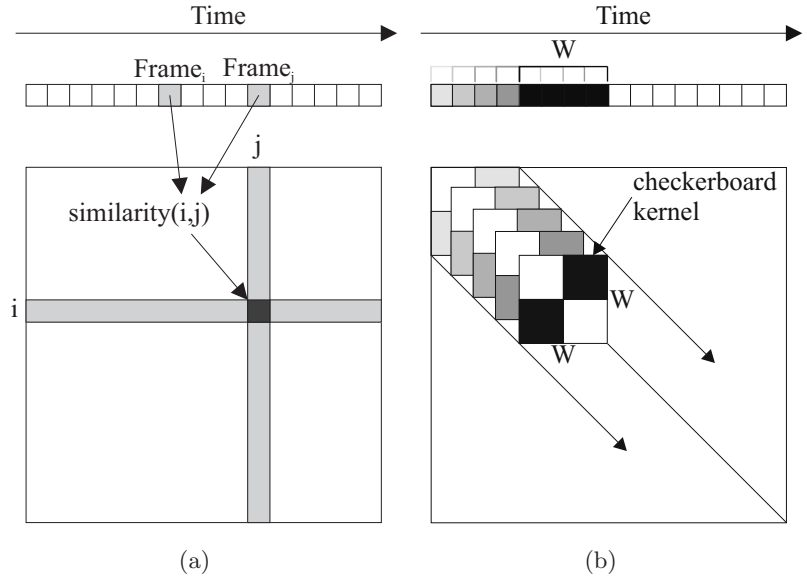


Figure 5.1: Similarity comparison: (a) the schema for constructing the similarity matrix; (b) the checkerboard function is moved along the diagonal of the matrix. The size W of the checkerboard function defines the number of frames under consideration.

frames yield low values (low similarity) in the matrix. The higher the similarity the brighter the gray values in the matrix. Figure 5.2 shows an example of a similarity matrix computed for 1000 frames. White pixels indicate nearly identical frames and black pixels represent dissimilar frames. The main diagonal (entries (i, i)) represents the self-similarity of the frames, which is always 1 (each frame is maximally similar to itself). Sequences of similar frames (e.g. frames of a shot) produce bright squares along the diagonal. This results in a checkerboard pattern along the diagonal, as shown in Figure 5.2.

Shot cuts can be found by detecting positions in the matrix where white squares adjoin each other at the diagonal. This means, that the task of shot cut detection is equivalent to detecting checkerboard patterns along the diagonal of the matrix. We move a square function (of size W) that looks like a checkerboard itself along the diagonal of the similarity matrix, in order to detect potential shot cuts. The checkerboard filter is smoothed by a Gaussian filter to avoid artifacts at the filter borders. The resulting checkerboard filter is shown in Figure 5.3.

The process of filtering is illustrated in Figure 5.1(b). At each position along the diagonal the checkerboard function is multiplied with the covered region of the matrix

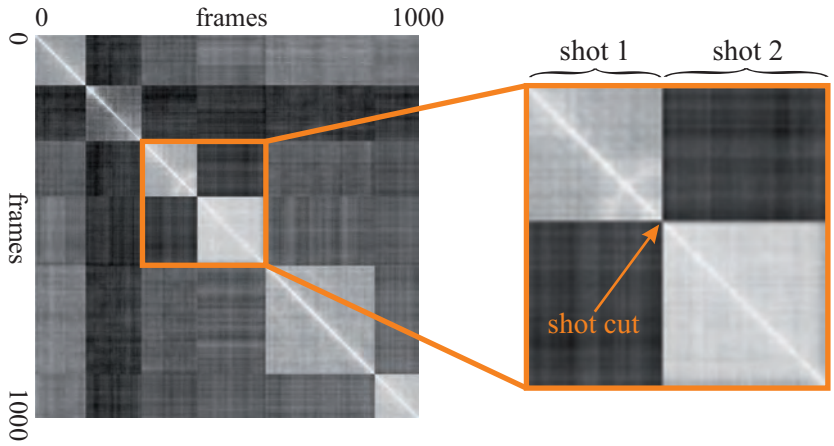


Figure 5.2: The left side shows an excerpt of the similarity matrix of 1000 frames of the film "The Eleventh Year". The bright squares along the diagonal indicate shots. The right side depicts the magnified checkerboard pattern produced by two adjacent shots.

and the result is summed up. This yields a high correlation value at positions where the matrix and the function are similar (checkerboard-like) and a low value otherwise. We obtain one correlation value for each position of the checkerboard function. The final result is a correlation function C for all frames. This function is used as an indicator for shot cuts. The correlation function C can be considered a *novelty curve* where peaks indicate a high novelty (an arbitrary event) in the underlying time series).

Computation of the entire similarity matrix is much too expensive and would require excessive amounts of memory. In practice, for shot cut detection it is sufficient to compute only similarities near the diagonal of the similarity matrix. The size W of the checkerboard function, defines the size of the processing window.

5.3.3 Shot Cut Detection

The correlation of the checkerboard kernel is a one-dimensional function C over all frames. The function shows peaks at potential shot cuts and has values near zero in homogeneous areas. A point in the correlation function is considered a shot cut if it is a local maximum and the difference to the preceding value exceeds a threshold t_c. The peak detection results in a list of detected shot cuts.

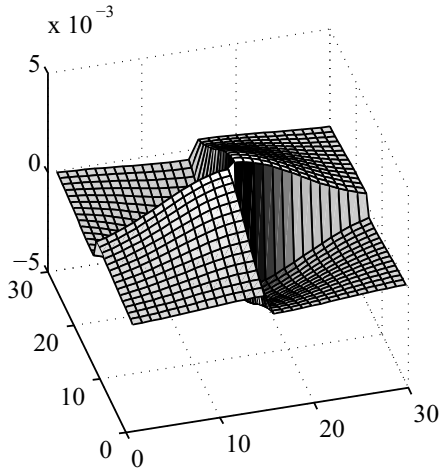


Figure 5.3: The Gaussian filtered checkerboard filter.

5.3.4 Feature Combination

We propose a novel scheme for the combination of several features for shot cut detection in this framework. There are several ways to fuse the information contained in the features. One possibility is to concatenate all features into one (high-dimensional) vector and then perform similarity comparison and shot cut detection based on this vector (*early fusion*). However, the influence of each component of the feature vector during similarity comparison is low because the feature vector has a high dimension. Consequently information captured by the features is lost at an early stage of processing. We expect suboptimal results for this approach.

Another possibility is to perform similarity comparison in parallel and independently for all features and to merge the resulting kernel correlation functions afterwards. The advantage of this approach is that more information is preserved until the end of the process. We propose a fusion scheme where one similarity matrix is computed for each feature. From each similarity matrix a separate correlation function is derived. Finally, the correlation functions are linearly combined to obtain a final correlation function for shot cut detection.

In our case, we compute two kernel correlations C_{bbDCT} and C_{EH} for the frequency and edge features and fuse them by a linear combination:

$$C_{merged} = w_{bbDCT} * C_{bbDCT} + w_{EH} * C_{EH}, \tag{5.1}$$

where w_{bbDCT} and w_{EH} are weighting factors. The merged correlation function C_{merged} is employed for shot cut detection as described in Section 5.3.3.

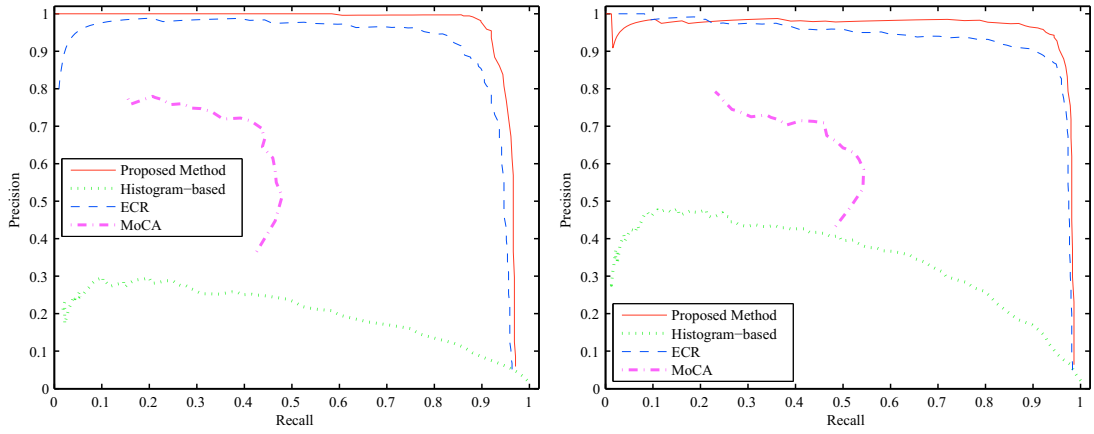
5.4 Experimental Results

We compare the proposed method with established, readily available techniques, namely an edge-based algorithm (Edge Change Ratio - ECR) proposed by Zabih et al. [245], the MoCA shot cut detector (which is based on intensity histograms) [125] and a block-based histogram technique from [28].

The parameters for the proposed technique are chosen as follows: The number of image blocks B determines the robustness to frame displacements and motion. A value of $B = 9$ has shown to be a good tradeoff. A number of $N = 36$ low-frequency DCT coefficients is suitable for the material employed in the experiments. The size of the kernel W has to be proportional to the length of the shortest shots in the films. A small kernel is necessary in order to detect shots of only a few frames which frequently appear in the investigated archive film material. We reduce the kernel size to $W = 6$ frames which is significantly lower than the kernel size of 21 to 71 frames employed by Cooper and Foote in [46]. The weights w_{bbDCT} and w_{EH} of the linear combination are chosen to be 0.5 because experiments showed that variations of the weights did not result in improved performance.

In a first step we evaluate the shot cut detection techniques mentioned above for two archive documentaries: “Kinopravda 21” and “The Eleventh Year”. “Kinopravda 21” has 35060 frames ($\approx$ 32min at 18fps) and contains 411 shot cuts, “The Eleventh Year” is 63123 frames long ($\approx$ 58min at 18fps) and contains 646 shot cuts. We apply the proposed method to both films and compare the results with that of the above mentioned techniques. Performance is measured in terms of recall and precision. We build recall-precision graphs by varying the threshold t_c (see Section 5.3.3). Figures 5.4(a) and 5.4(b) show recall versus precision of the employed techniques for both films.

We observe that the intensity histogram-based techniques are not appropriate for shot cut detection in archive film. The main reason for this is that the histograms are not robust against flicker. ECR yields significantly higher recall and precision than the histogram-based techniques. This proves the assumption that edge information is more robust to the artifacts in archive film. However, the proposed technique outperforms



(a) Kinopravda 21

(b) The Eleventh Year

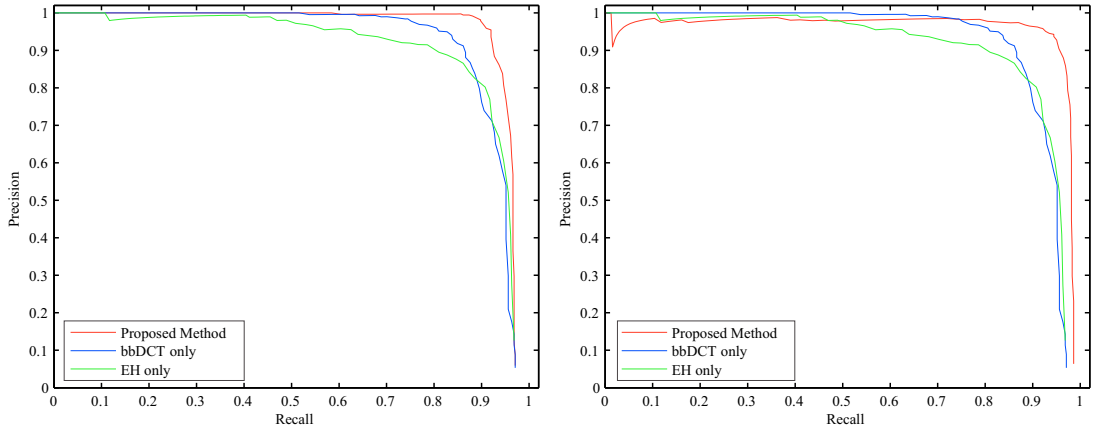
Figure 5.4: Recall-precision graphs for both films. The solid line is the proposed method, the dashed line is ECR, the dotted line is the histogram-based approach and MoCA is the dash-dot line.

	Hist.-based	MoCA	ECR	EH	bbDCT	early fusion	linear comb.
Kinopravda 21	0.32	0.54	0.88	0.86	0.89	0.89	0.94
The Eleventh Year	0.46	0.57	0.91	0.86	0.89	0.90	0.94

Table 5.1: The maximum f_1 scores obtained from the recall-precision pairs for all investigated methods

ECR in recall as well as in precision for both films. For the film "Kinopravda 21" both measures are significantly increased compared to ECR. In the film "The Eleventh Year", the proposed method mainly increases precision. These results are promising in the context of the highly degraded material. We summarize the performance of the discussed methods in Table 5.1. We compute the f_1 scores for all recall and precision pairs obtained in the experiments and list the maximum f_1-scores, in order to provide a measure of the achievable performance.

We further analyze the performance of different feature selections and fusion strategies in the context of the proposed method. As mentioned in Section 5.3.1 the bbDCT and the EH describe complementary information and thus we assume them to be good candidates for combination. We verify this assumption by comparing the performance of the shot cut detector based on individual features with the performance of the fea-



(a) Kinopravda 21

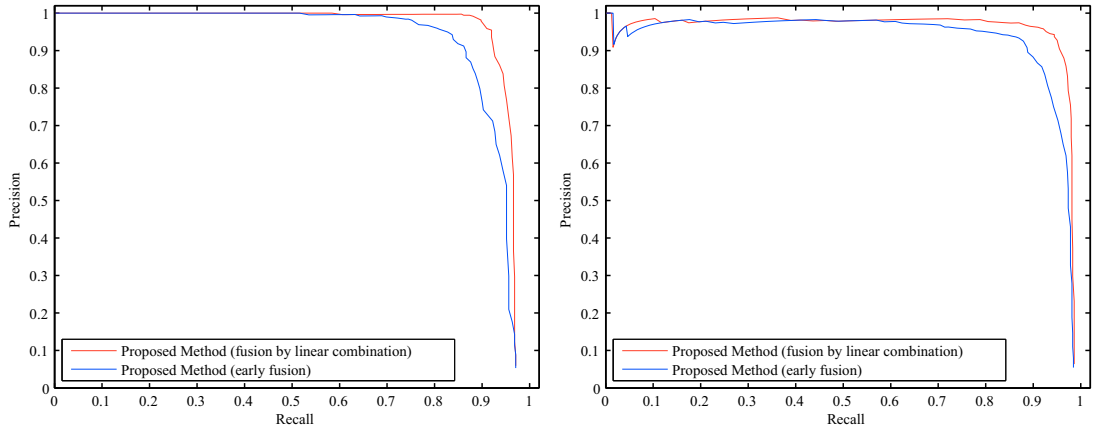
(b) The Eleventh Year

Figure 5.5: Recall-precision graphs for both films. The combination of the features significantly increases performance compared to the single features.

ture combination (using linear combination). The resulting recall-precision graphs (see Figure 5.5) show that the individual features yield only suboptimal results. The fusion by linear combination raises the performance figures significantly, which proves its beneficial effect. Table 5.1 lists the respective performance figures in columns *EH*, *bbDCT*, and *linear comb*.

We further evaluate the performance of the two fusion strategies presented in Section 5.3.4. The combination of the features prior to similarity comparison (early fusion) yields only suboptimal results, similarly to the performance of the single features. The second strategy (linear combination of individual kernel correlations) significantly increases the performance as shown in Figure 5.6. These results show that the fusion scheme with linear combination better exploits the information captured by the features.

In the following, we perform retrieval experiments with a larger number of archive films. In this evaluation we focus on the selection of the detector's parameters: the window size W of the checkerboard kernel and the threshold t_c for peak detection. First, we evaluate recall and precision for different values of W. Table 5.2 shows the optimal achievable f_1 score for the investigated kernel sizes. We observe that there is no common best option for W. The size of the checkerboard filter W restricts the minimum distance between two distinguishable shot cuts and thereby the minimum



(a) Kinopravda 21

(b) The Eleventh Year

Figure 5.6: Recall-precision graphs for both films. We observe that compared to the early fusion, the linear combination of the kernel correlations increases the performance

duration of a recognizable shot. As a consequence, films with many short shots, such as "Man with a Movie Camera" require a smaller processing window ($W = 4$ in our experiments) than films with longer shots.

Next, we perform experiments for different values of parameter t_c. For this purpose we compute recall and precision for 100 values of t_c and select the parameter value that maximizes the f_1 score. The optimal parameter values of t_c for different films are summarized in Table 5.3. We observe that the optimal values of t_c lie in the range [0.15 0.2].

From Table 5.2 we observe a relation between the parameters W and t_c. The larger the values of W the smaller the values of t_c. We expect this relation because the kernel correlation curve produced by the checkerboard kernel becomes smoother with increasing kernel size. With increasing smoothness the height of the peaks that indicate the shot cuts decreases. A regression analysis reveals a quadratic relationship between the two parameters. This means that, given the checkerboard kernel size W we are able to estimate a suitable parameter value of t_c. In this way we can efficiently estimate the parameter value in the absence of ground truth. For the estimation of kernel size W knowledge about the minimum shot length is beneficial.

Film	$W = 2$	$W = 4$	$W = 6$	$W = 8$	$W = 10$
Kinoglaz	0.9396	**0.9490**	0.9480	0.9455	0.9412
Schatten der Maschine	0.9053	0.9157	0.9235	0.9255	**0.9259**
The Eleventh Year	0.9312	0.9365	**0.9435**	0.9426	0.9428
Enthusiasm (Original)	0.6455	0.8175	0.8876	0.9061	**0.9123**
Enthusiasm (Restored)	0.7022	0.8516	0.9054	0.9120	**0.9125**
Man with a Movie Camera (V)	0.8907	**0.9024**	0.8871	0.8567	0.8305
Man with a Movie Camera (A)	0.8861	**0.8974**	0.8783	0.8522	0.8281
Three Songs of Lenin	0.8800	0.9299	**0.9513**	0.9511	0.9483

Table 5.2: f_1 scores for different films and kernel sizes W. The highest f_1 scores for each film are typeset bold. We observe that the optimal kernel size depends on the film.

Film	$W = 2$	$W = 4$	$W = 6$	$W = 8$	$W = 10$
Kinoglaz	0.2415	**0.1993**	0.1795	0.1634	0.1542
Schatten der Maschine	0.2539	0.2155	0.1881	0.1791	**0.1635**
The Eleventh Year	0.2172	0.1822	**0.1579**	0.1481	0.1389
Enthusiasm (Original)	0.3817	0.2969	0.2431	0.2192	**0.1979**
Enthusiasm (Restored)	0.3990	0.3070	0.2434	0.2129	**0.1994**
Man with a Movie Camera (V)	0.1931	**0.1526**	0.1161	0.0968	0.0871
Man with a Movie Camera (A)	0.1953	**0.1567**	0.1249	0.1008	0.0930
Three Songs of Lenin	0.2620	0.2142	**0.1830**	0.1724	0.1639

Table 5.3: The optimal thresholds t_c (with respect to f_1 scores) for the different kernel sizes W. Thresholds that produce optimal retrieval performance in combination with W are typeset bold as in Table 5.2. Regression analysis reveals a quadratic relationship between the threshold and the kernel size.

5.5 Summary

In this chapter we have presented a robust shot cut detector for archive film material. The method extends a state-of-the-art method for shot cut detection by more robust and suitable features for archive film and a more effective feature fusion scheme. We employ block-based DCT coefficients to capture low-frequency content in the film's frames and an edge histogram to represent the high-frequency content. We perform a self-similarity analysis of subsequent frames and compute a correlation function (novelty curve) for each feature separately. The linear combination of both novelty curves yields the final novelty function for shot cut detection. The peaks in the novelty function correspond well to the shot cuts in the films.

We perform experiments with a large number of archive films. From the experiments we observe that both content-based features are robust against the artifacts present in the films. We further learn that a linear combination of the novelty curves outperforms the original approach where the features are fused by concatenation prior to the similarity computation. The proposed method achieves satisfactory results with f_1 scores beyond 0.90 for the detection of shot cuts in archive film.

The proposed method outperforms other established shot cut detectors and is able to cope with the complex spatio-temporal structure and the manifold artifacts of archive films [247]. The next step is the identification of other types of shot boundaries, the so called gradual transitions, which is the topic of the next chapter.

Chapter 6

Detection of Gradual Transitions

In this chapter, we present a method for the detection of gradual transitions in archive film material that builds upon the shot cut detector presented in Chapter 5. We present the different types of gradual transitions in Section 6.1 and discuss the differences between gradual transitions in archive and contemporary films. For gradual transition detection two basic types of approaches exist, namely specialized approaches and unified approaches. We review both types of approaches in Section 6.2 and analyze their strengths and weaknesses. In Section 6.3 we develop a unified approach for the detection of gradual transitions in archive film material based on the shot cut detector from Chapter 5 and the method presented in [47]. The major contribution of this chapter is a first *systematic evaluation* of gradual transition detection in archive film material in Sections 6.4 and 6.5. In the systematic evaluation we investigate the individual processing steps of gradual transition detection, different low-level features and feature combinations, similarity measures, fusion strategies, and different system parameters and evaluate their effect on the detector performance. Additionally, we perform experiments and evaluations with contemporary material in order to compare the behavior of the method and its components for contemporary and archive film material.

6.1 Introduction

There are different ways to connect two consecutive shots in a film. Additionally to abrupt shot cuts, which are the most common type of shot boundaries, a filmmaker has the opportunity to connect two successive shot by a gradual transition. In contrast

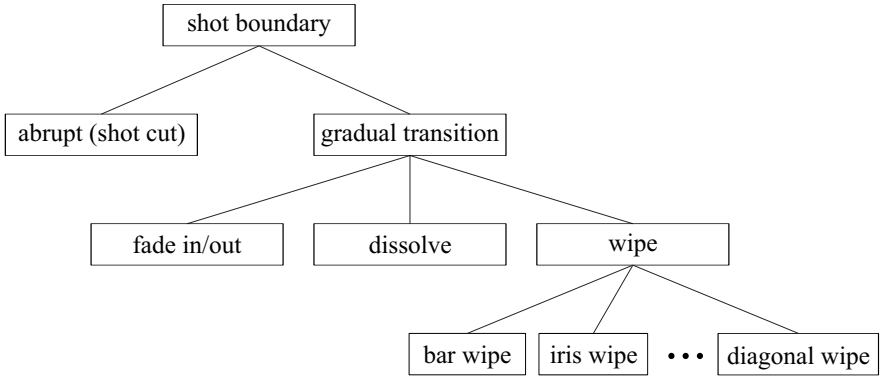
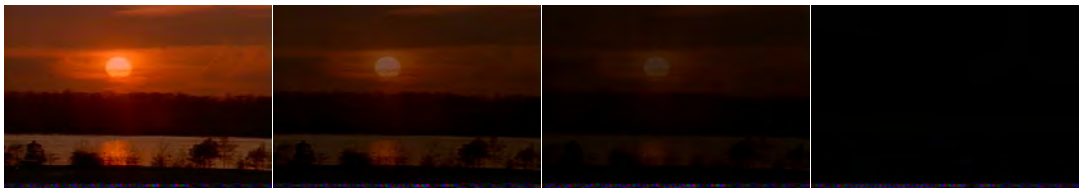


Figure 6.1: A taxonomy of shot boundaries. Boundaries between shots are either abrupt transitions (shot cuts) or gradual transitions. We distinguish between three structurally different types of gradual transitions: fades (either fade-in or fade-out), dissolves and wipes. For wipes a large number of subtypes exist, such as bar wipes, iris wipes, etc.

of shot cuts, gradual transitions represent a *continuous* change between two successive shots. The concept of gradual transitions is an important stylistic means for the filmmaker to smoothly change from one shot to the next. A wide range of different gradual transition types exist, such as fades (fade-in and fade-out), dissolves, and wipes. A taxonomy of shot boundaries is shown in Figure 6.1.

A *fade-out* is a transition where "the image on the screen fades to black" [20]. A *fade-in* refers to the opposite: a shot "gradually fades in from black" [20]. Fade-outs and fade-ins are often combined to lead the viewer from one shot to the next. A *dissolve* is a gradual transition where one shot directly fades into another shot. For a short duration both shots are visible simultaneously. According to Beaver the typical length of a dissolve is two seconds [20]. Figure 6.2 shows an example of a fade-out and a dissolve in contemporary material. The third class of gradual transitions are *wipes*. Wipes denote special transforms between two shots. A typical wipe is for example the *horizontal wipe*, where one shot pushes away the previous shot from left to right or from right to left. Wipes were more common in the earlier era of filmmaking [20]. In the silent films of Dziga Vertov for example wipes are frequently employed to switch from one motif to the next. A typical wipe in the films is the *iris wipe*. In an *iris-in wipe* "an existing image moves into a circle which rapidly decreases in size until it disappears" [20]. Synchronously, in the area around the circle a new shot may become



(a) fade-out

(b) dissolve

Figure 6.2: A fade-out and a dissolve in contemporary (TRECVID) video material.

visible. The reverse process of iris-in is referred to as *iris-out wipe*. Figure 6.3 shows different examples of iris-out wipes in historic film material.

The automatic detection of shot cuts and gradual transitions for shot segmentation is a well-researched topic. In the context of the TRECVID benchmark initiated by the National Institute of Standards and Technology (NIST) in 2001 numerous methods for shot boundary detection have been developed [198]. Additionally, the TRECVID benchmark has made a large amount of annotated video material for shot boundary detection available that enables the objective comparison of different approaches. Since 2008 the detection of gradual transitions is declared to be solved by the TRECVID organizers [197, 198]. However, the TRECVID material mostly contains high-quality (color) video, consequently the conclusion above cannot be transferred to low-quality and monochromatic archive film material.

For shot cuts, we have shown in Chapter 5, that reliable and satisfactory detection is possible in the context of archive film by using robust low-level features and a late fusion strategy (see Section 5.3.4). The main challenges in gradual transition detection are (i) the large number of different gradual transition types that exist, (ii) the varying duration of gradual transitions, and (iii) object- and camera movements which are easily confused with gradual transitions [82, 123, 242, 244]. Archive film material poses additional challenges to gradual transition detection:

(a) centered iris-out of an intertitle (over a black frame).

(b) asymmetric iris-out from bottom left corner (over a black frame).

(c) iris-out (over another shot).

Figure 6.3: Different examples of iris-out wipes in historic material. Note the artifacts, for example the low contrast in (c) that makes this transition difficult to detect even for human observers.

1. The gradual transitions in archive material are longer than in contemporary material. According to Table 6.1 the mean duration of gradual transitions in historic material is approximately three times the mean length in TRECVID material (30.7 versus 11.83 frames).

2. The number of different gradual transitions types is larger for historic material than for TRECVID material (8 versus 3), see Table 6.1. Examples are given in Figures 6.3 and 6.4.

3. The archive film material contains artifacts that impede gradual transition detection, such as flicker and low contrast originating from uneven exposure of the

Type of Material	Gradual Transition Type	#Frames per GT		
		Min	Max	Mean
Historic	Dissolve	15	134	31.8
	Iris in (from black)	10	52	26.3
	Complex transition	25	93	49.6
	Iris in (not to black)	23	34	30.9
	Bar wipe	25	53	36.8
	Iris out (to black)	6	21	12.5
	Iris out (not to black)	29	47	38.0
	Fade out	20	25	22.5
	All	**6**	**134**	**30.7**
TRECVID	Dissolve	1	22	2.9
	Fade in/out	7	16	10.9
	Other	4	107	21.7
	All	**1**	**107**	**11.83**

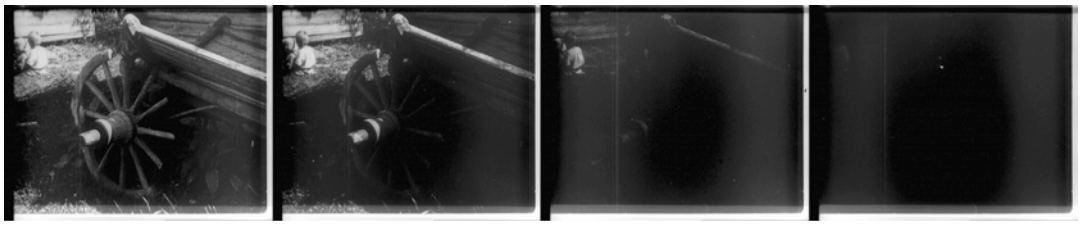
Table 6.1: Gradual transition (GT) types and their durations in historic and TRECVID material. The gradual transition types are sorted by descending occurrence frequency in historic and TRECVID material, respectively.

filmstrip (see Section 3.2). Such artifacts create patterns that are easily confused with gradual transitions and consequently produce false positives.

4. Archive film material is usually monochromatic, which means that color information cannot be exploited. Many existing approaches rely on color and thus cannot be applied.

6.2 Related Work

There are principally two different types of approaches for gradual transition detection: *specialized approaches* and *unified approaches*. In specialized approaches a separate specialized detector is developed for each type of gradual transition (e.g. a dissolve detector, a fade-in detector, etc.) A large number of specialized approaches has been proposed in literature [105, 243, 249]. A popular example is the twin comparison method by [249] which was later extended by [243]. Most methods rely on color information (e.g. color histograms) and employ thresholds for the detection of gradual



(a) a fade out (note the uneven distribution of intensity during fade out).

(b) a dissolve sequence.

(c) a variant of a bar wipe (a barn door vertical open wipe).

(d) a combination of fade-out and iris-out. The intertitle is faded out and simultaneously an iris-out proceeds.

Figure 6.4: Different examples of transitions that demonstrate the rich diversity of gradual transitions in historic film material.

transitions. According to Yuan et al., a threshold's value highly depends on the genre of the video and thresholds cannot exploit information about the shape of a peak or valley (e.g. if a peak is sharp or not) in a signal [244]. Consequently, thresholds are not robust to different types of film and video material and lack in expressiveness.

Liu et al. overcome the problems with thresholds by applying machine learning for decision making [128, 131]. Their method delivered the best shot boundary detection performance in TRECVID 2006 and 2007. The authors propose for example a specialized dissolve detector based on the change of variance of color histograms during a dissolve. They assume that a dissolve is a linear mixture of two shots and thus the change of the color variance during a dissolve follows typical curves. Classification of dissolves is performed with finite state machines and support vector machines (SVM).

Specialized approaches are not well-suited for gradual transition detection in archive film material. *First*, most approaches employ color features and thresholds. Color features are not applicable to archive material and thresholds are usually not robust to the artifacts and distortions in archive material. *Second*, specialized approaches require one detector for each gradual transition. The individual detectors are usually highly optimized which requires a large amount of training data. Archive film material contains a large number of differing gradual transition types which would result in a large number of individual detectors. Furthermore, archive material might not contain enough examples to train each specialized detector (e.g. if a particular gradual transition appears only a few times in the films). A shot boundary detector for archive material should be able to detect gradual transitions even if it has *not* been trained for it.

Unified approaches better fulfill the requirements of gradual transition detection in archive film material. Unified approaches are more general than specialized approaches, since they incorporate only one detector for all gradual transitions. This allows a broader applicability and makes the approaches less data dependent. Consequently, they are able to detect transitions they have not been trained for. However, for unified approaches the expected detection rate is lower than for specialized approaches because unified approaches do not exploit a priori knowledge about the different gradual transitions types.

Different unified approaches have been proposed in literature. Bescos et al. introduce a *unified* approach for gradual transition detection which is based on inter-frame comparison with different temporal distances. They use RGB color values as

features and detect gradual transitions by thresholding [22]. The method relies on inter-temporal comparison of all frames with *one* frame only. Consequently, we expect a lack of robustness for low-quality film material where single frames may be highly disturbed.

Yuan et al. utilize a similarity matrix that represents inter-frame similarities [244]. The approach is based on the fact that due to the varying lengths of gradual transitions, a gradual transition does not leave a pattern as clear as a cut in the similarity matrix (see Figure 5.2 in Section 5.3.2, Chapter 5 for an example of the clear checkerboard pattern produced by abrupt shot cuts). The authors compute a self-similarity matrix with a lower temporal resolution, e.g. by decreasing the frame rate in order to avoid this problem. In this low-resolution similarity matrix a gradual transition leaves a clearer pattern (more similar to those of a shot cut). The employed features in the approach are global and block-based color histograms and classification is performed by an SVM.

Cooper et al. propose an approach for shot boundary detection based on a self-similarity matrix, as well [47]. The self-similarity matrix is constructed from the pair-wise comparison of global and block-based color histograms of successive frames. The authors extract intermediate features from the similarity matrix. The intermediate features represent the neighborhood of a frame and are input to a K-NN classifier.

The approaches of Yuan et al. [244] and Cooper et al. [47] rely on self-similarity matrices that compare all frames of a sequence with each other. This property makes them more robust to distortions. Both approaches perform comparably well on the TRECVID material. We expect these two approaches to be superior to the approach of [22] for archive film material.

6.3 Robust Gradual Transition Detection

The proposed approach for gradual transition detection is based on the robust shot cut detector presented in Chapter 5 and the approach of Cooper et al. [47]. We extend the approach of Cooper et al. [47] by integrating robust features of the shot cut detector from Chapter 5 and by exchanging the K-NN classifier by an SVM which is more robust and less dependent on data.

According to Yuan et al. a shot boundary detector consists of three processing steps: (i) visual content representation, (ii) construction of the continuity signal and

(iii) classification [242]. We add a fourth step of (iv) verification to detect and reject false positives. In the following sections we describe the four basic processing steps in detail.

6.3.1 Visual Content Representation

Visual content representation refers to the extraction of features which are meaningful for gradual transition detection. The features should provide a compact and robust representation of the visual content of a film. A basic requirement is that the features are invariant to object motion and camera motion to a high degree. In the context of archive film material, the features have to fulfill additional requirements, such as invariance towards flickering, scratches, mold and dust.

Since, it is a priori unknown, which features or feature combinations perform best on archive film material, we extract a representative set of features that captures different aspects from the visual signal. In the systematic evaluation in Section 6.4 we evaluate the performance of the single features and feature combinations. As the historic material is black and white, we extract luminance histograms. Additionally, we employ global DCT coefficients and MPEG-7 edge histograms as in Chapter 5 since they have already performed well for shot cut detection. We extract the luminance histograms as well as the edge histograms globally and block-based in order to capture local and global information.

6.3.2 Construction of the Continuity Signal

The next step is the computation of a continuity signal which is an indicator for changes in the visual signal. Changes may indicate shot boundaries (e.g. dissolves or fades) or other global changes (e.g. camera movements, movements of large objects, and illumination changes). We construct the continuity signal in three steps: self-similarity matrix construction, intermediate feature extraction, and fusion.

Self-similarity matrix construction

We first normalize each feature component separately (over all frames) by a min-max normalization that transforms all values into the range between 0 and 1. Next, we construct a self-similarity matrix as described in Section 5.3.2 for shot cut detection. We

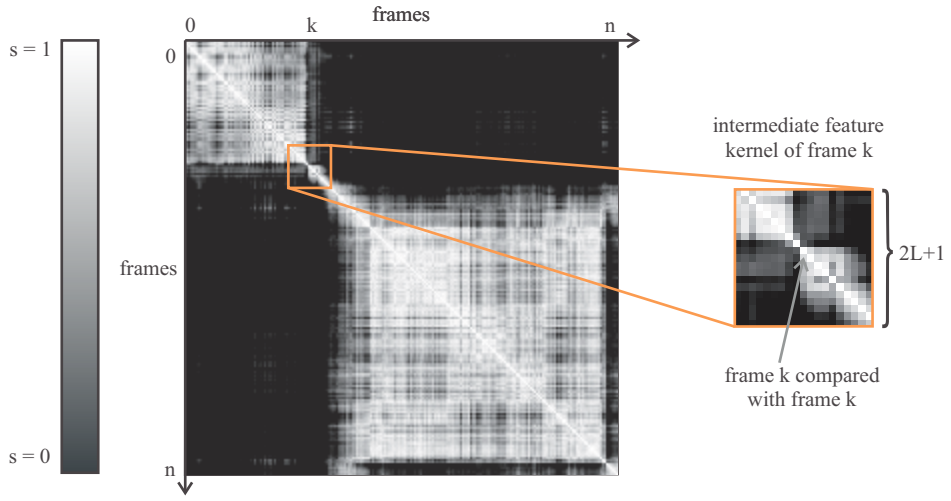


Figure 6.5: Similarity matrix of a dissolve with the intermediate feature kernel of frame k. Dark areas in the similarity matrix indicate low similarity s and bright areas indicate high similarity.

alternatively employ three different similarity and distance metrics for the construction of the matrix: L2 distance, Cosine similarity and χ^2 distance. Figure 6.5 shows a resulting similarity matrix of a dissolve. We observe two bright squares which correspond to the two shots that are connected by the dissolve. The area along the diagonal between the two bright squares in the similarity matrix represents the dissolve.

Intermediate feature extraction

The intermediate features represent the temporal neighborhood of a frame in the similarity matrix. The size of this neighborhood (the intermediate feature kernel lag L) is the number of past and future frames of a frame k that are considered for the intermediate feature. The intermediate feature of a frame k consists of the portion of the similarity matrix that corresponds to the frames $k - L$ to $k + L$ (see Figure 6.5 for an illustration).

Since the similarity matrix is symmetric, we select one half of the intermediate feature kernel as intermediate features. We call this a *full similarity kernel*. To avoid the curse of dimensionality during classification it is useful to reduce the dimension of the intermediate feature vectors. For this purpose, Cooper et al. identified the most relevant components of the intermediate feature vectors by greedy feature selection

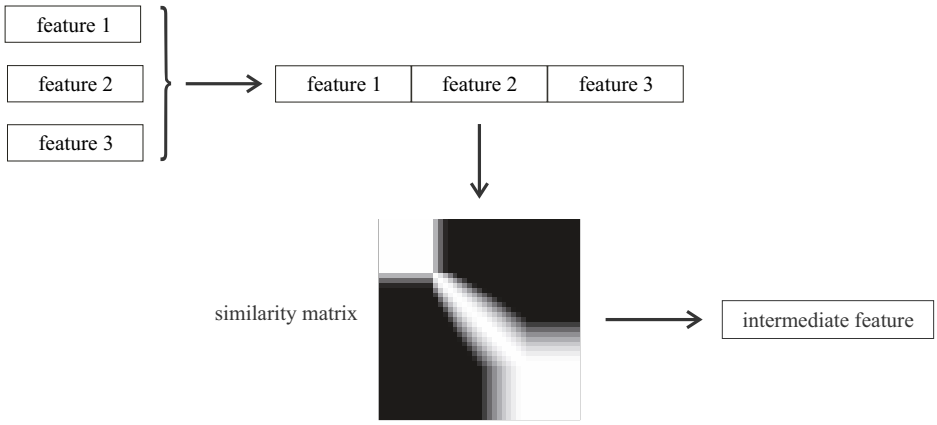


Figure 6.6: Schema of feature combination with early fusion. In early fusion the feature vectors are first concatenated and subsequently used to compute the similarity matrix.

which results in a reduced *greedy kernel* [47]. We alternatively extract the full kernel and the selected (reduced) kernel of [47] for the evaluation.

Fusion

The information captured from the frames can be maximized by the combination of different (complementary) features. There are two ways to combine information from different features in this framework: early fusion and late fusion. In *early fusion*, we concatenate different feature vectors and use the resulting vector as input to the calculation of the similarity matrix. The result is a single similarity matrix as illustrated in Figure 6.6). The intermediate feature vector for each frame is derived from this similarity matrix as described above.

For *late fusion* we compute a similarity matrix for each feature separately. Next, we extract intermediate features from each resulting similarity matrix. Finally all intermediate feature vectors are concatenated as shown in Figure 6.7.

6.3.3 Classification

The intermediate feature vectors (independent of the type of fusion) represent the continuity signal for the analyzed sequence. During classification each frame in the sequence is classified as either being part of a gradual transition or not. This is performed by classifying the intermediate feature vector of each frame. We use a Support Vector Machine (SVM) for classification, see Section 2.5.3. The SVM is able to process high

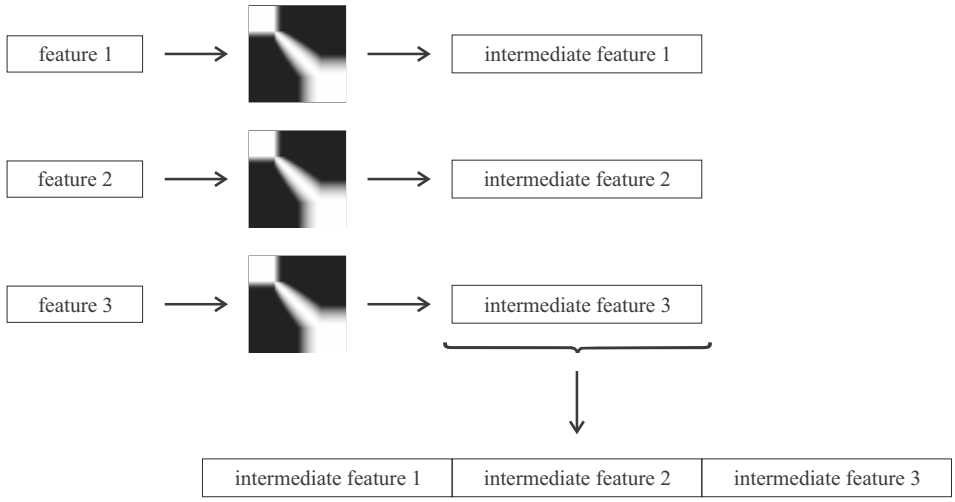


Figure 6.7: Schema of feature combination with late fusion. In late fusion for each feature a separate similarity matrix is computed. Intermediate features are derived from each similarity matrix in parallel and are finally concatenated.

dimensional feature vectors (which is necessary in case of the full similarity kernel) and is less prone to overfitting than the K-NN classifier originally employed in [47]. In the experiments we evaluate the performance of the SVM with different SVM kernels.

The result of classification is a binary label for each frame that indicates whether or not a frame has been classified as being part of a gradual transition or not. Note that, since the intermediate features represent the temporal neighborhood of a frame, the classifier implicitly incorporates temporal information.

6.3.4 Verification

We expect many false positives, outliers, and gaps in the classification results due to quality of the historic material. We smooth the labeling obtained by classification with a temporal median filter in order to eliminate outliers and to fill gaps. Furthermore, we propose begin-end matching and KLT verification for the identification (and subsequent elimination) of false positives that might occur due to camera or object motion and abrupt illumination changes.

The goal of *begin-end matching* is to remove false positives. We assume that a low similarity between the frames at the beginning and at the end of a candidate gradual transition indicates a high likelihood that a gradual transition actually occurred. We

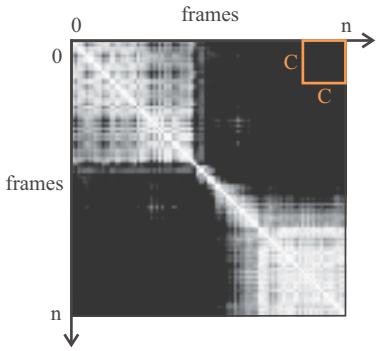


Figure 6.8: Schema for begin-end matching. The mean of the similarity values in the $C \times C$ square indicate whether or not the beginning and the end of a candidate transition are similar.

use the similarity matrix to calculate the similarity between the beginning and the end of a candidate transition. We take a square of size $C \times C$ frames from the upper right corner of the similarity matrix of the candidate transition (see Figure 6.8 for an illustration. These values represent the similarity between the beginning and the end of the transition. Next, we calculate the mean of the $C \times C$ similarity values. A low value (low similarity) indicates a high likelihood that a gradual transition actually occurs. A high value indicates a false positive.

KLT verification aims at identifying false positives caused by camera and object motion. We assume that all objects in a scene must disappear across a gradual transition. That means, we cannot track the objects continuously across the transition. We use the KLT feature tracker to detect and track motion [194]. In case we find KLT feature trajectories that persist through a sequence of frames classified as gradual transition, we conclude that the sequence is a false positive. If more than a certain number of trajectories does not break off across the candidate transition, we mark the sequence as false positive. Theoretically, we expect this threshold to work perfectly with a value of one, as already one continuous trajectory falsifies a gradual transition. In practice, higher values of the threshold give higher confidence of the falsification.

6.4 Systematic Evaluation

We perform a systematic evaluation of the individual processing steps and parameters of the proposed method. We evaluate the method's performance on archive film material

as well as on contemporary (TRECVID) material. In the following, we present the setup of the systematic evaluation and the research questions that are investigated.

6.4.1 Setup for Archive Film Material

Figure 6.9 shows the setup of the experiments with archive film material. We start with the evaluation of the best global and local single features and the best feature combination (using early and late fusion). For these three feature sets, we investigate the influence of different parameters on retrieval performance. In particular we investigate the following research questions:

1. *Which feature delivers the best results?* We evaluate each feature separately.

2. *Which combination of features delivers the best results, and which fusion strategy performs best?* We use single features and combine them. For all combinations we evaluate early fusion and late fusion.

3. *Which similarity measure delivers the best results?* We evaluate the performance of three similarity measures: L distance, Cosine similarity and χ^2 distance.

4. *How does the kernel lag L influence the results?* We use $L = 10$ as default. We evaluate $L = 6$ and $L = 15$ with the previously selected features. Note that the actual kernel size for intermediate feature creation is $2L + 1$.

5. *How do different feature selection kernels for intermediate feature creation influence the results?* We evaluate the performance of the full kernel and the reduced greedy kernel.

6. *Which kernel of the SVM delivers the best results?* We evaluate different kernels for the SVM. The standard kernel used in all experiments is linear. Furthermore, we evaluate a quadratic and a polynomial kernel of third order.

7. *What influence do the verification steps have?* We take the best performing setups of the previous experiments and perform median filtering with filter sizes from 3 to 41 to remove outliers. Furthermore, we perform KLT verification and begin-end matching.

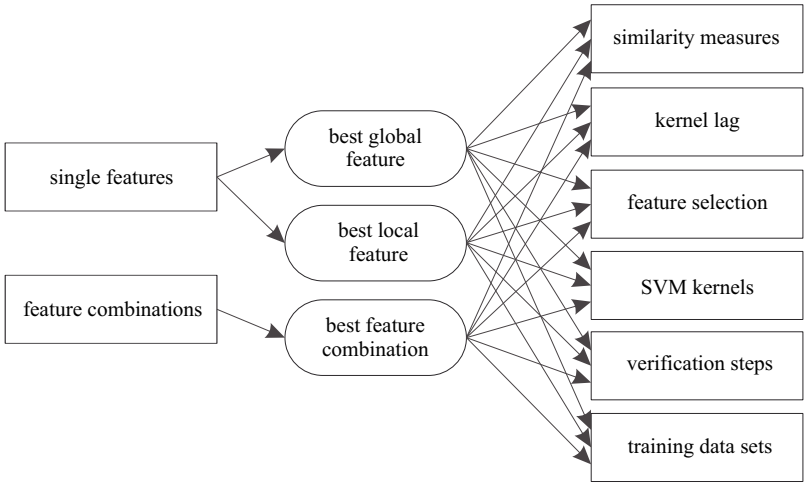


Figure 6.9: Overview of the systematic evaluation with archive film material. We start with the evaluation of single features and feature combinations. For the best features we evaluate numerous parameters of the method (similarity measures, kernel lags, feature selection, SVM kernels, verification steps).

8. *How do results depend on training data?* We perform experiments with three different training datasets which contain randomly and manually selected samples, respectively.

6.4.2 Setup for Contemporary Material

We employ contemporary material from the TRECVID benchmark as reference data to test the validity of our approach. The TRECVID evaluation only distinguishes between shot cuts and gradual transitions, where a shot cut is a shot boundary of length zero and a gradual transition is any other shot boundary with a length greater than zero. This is suitable for our evaluation, since we aim at detecting gradual transitions independently of their type. Note that abrupt shot cuts are not considered in the evaluation.

We use film material from the TRECVID 2006 shot boundary task. The material consists of news magazines, science news, news reports, documentaries and educational programs. At the time of the experiments, the TRECVID 2007 material was available as well. The 2006 material however contains more gradual transitions and thus is better suited for the evaluation.

For contemporary material we evaluate a subset of the parameters investigated for the historic material. Based on the experiences from the previous experiments, we

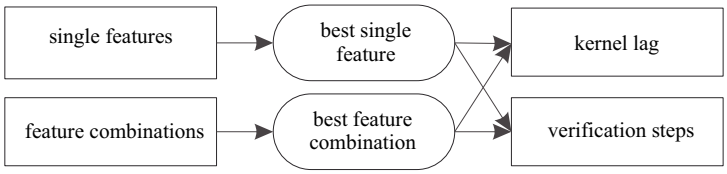


Figure 6.10: To test the validity of our approach, we perform experiments on contemporary reference material from TRECVID 2006. First, we evaluate single features and feature combinations. For the best results from these experiments we evaluate the most important parameters of the proposed method.

focus on those parameters which have the highest influence on retrieval performance. Figure 6.10 gives an overview of the experiments.

In a first step, we investigate the best performing feature and identify the best performing feature combination using early and late fusion. In the next step, we investigate the influence of the median filter in the same way as for the historic material (see Section 6.4.1). Finally, we examine the size of the kernel lag L. We reduce the kernel lag due to the shorter duration of gradual transitions in contemporary material (see Table 6.1). We employ a value of $L = 6$ as the default and additionally $L = 4$ and $L = 10$.

6.4.3 Evaluation

For training the SVM on historic material, we use two randomly selected data sets and one manually selected set (see page 127 for details). For the validation of our approach with contemporary material, we use one randomly selected training data set.

We design the validation for historic and contemporary material to fit the TRECVID evaluation criteria in order to enable a comparison with previous TRECVID results. In the TRECVID 2006 task on shot boundary detection each participating group is allowed to submit up to 10 runs to the evaluation, i.e. 10 detector variants. In our case, the different detector variants correspond to differently trained SVMs. We compare the best result (i.e. the best performing SVM model) to the best results of the TRECVID evaluation.

We use frame recall f_r and frame precision f_p as performance measures in the evaluation. Frame recall and precision are defined as in Section 2.6, where a *document* refers to a *frame* in this evaluation. Consequently, frame recall f_r is defined as:

Abbreviation	Feature
GLH	Global luminance histogram
LH2x2, LH3x3, LH4x4	Local luminance histograms extracted from 4, 9 and 16 blocks.
GEH	Global edge histogram
EH2x2, EH3x3, EH 4x4	Local edge histograms extracted from 4, 9 and 16 blocks.
DCT	Local DCT coefficients

Table 6.2: Features employed in this study and their abbreviations.

$$f_r = \frac{|\{frames\ correctly\ assigned\ to\ gradual\ transitions\}|}{|\{frames\ belonging\ to\ grad.\ trans.\ in\ ground\ truth\}|}\ , \tag{6.1}$$

and frame precision f_p is:

$$f_p = \frac{|\{frames\ correctly\ assigned\ to\ gradual\ transitions\}|}{|\{all\ retrieved\ frames\}|}\ . \tag{6.2}$$

Additionally, we employ the f_1 score (see Section 2.6) obtained from frame recall and frame precision for evaluation. The usage of these measures makes our results fully comparable with the TRECVID results.

6.5 Experimental Results

6.5.1 Archive film material

An overview of the systematic evaluation is given in Figure 6.9. First, we present and discuss the performance of single features and the performance of feature combinations with different fusion strategies. Table 6.2 contains a description and the abbreviation of the features we extract from each frame. We conduct all feature related experiments with the same training data set. We utilize the χ^2 distance to construct the similarity matrices and employ a full similarity kernel with lag $L = 10$. The SVM is trained with a linear kernel and verification is skipped.

Single features

The evaluation of single features in historic material shows that the local edge histogram with 16 blocks (EH4x4) performs best (see Figure 6.11). We observe, that block-

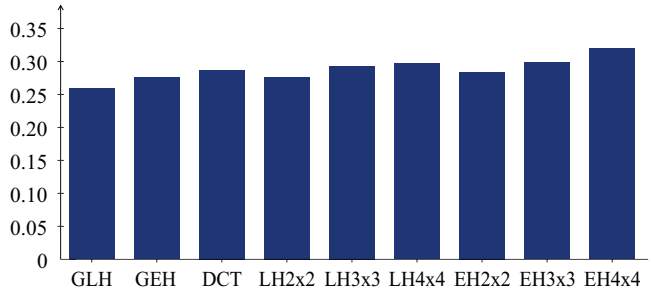


Figure 6.11: Performance of single features for historic material in terms of the f_1 score. The best performing single feature is the block-based edge histogram with 16 blocks (EH4x4).

based features perform better than global features, and that a larger number of blocks increases performance. This applies to luminance histograms (GLH, LH2x2, LH3x3 and LH4x4) as well as edge histograms (GEH, EH2x2, EH3x3 and EH4x4). From the experiments, we learn that structure and shape information represented by the edge histograms are more important (and more robust) for gradual transition detection than the intensity distribution provided by the luminance histograms. This contradicts earlier findings based on contemporary material [47, 242].

Feature fusion

Figure 6.12 shows the results for different feature combinations compared to the best single feature. Early fusion decreases quality in most of the cases. The combination of features with late fusion tends to improve results. The combination of all features performs best. We observe that all late fusion combinations, that perform significantly better than the best single feature, utilize DCT in combination with at least one local feature. The late fusion of DCT with a local and a global feature yields the largest performance improvement. We assume that the improvement is due to the fact that the combined features represent complementary information.

Similarity measures

Table 6.3 summarizes the results of different similarity measures for the best global feature (GEH), the best local feature (EH4x4), and the best feature combination (GLH, DCT, GEH, EH4x4, LH4x4 with late fusion). For the evaluation of the similarity measures we fix the kernel lag at 10, use a full kernel for the generation of intermediate

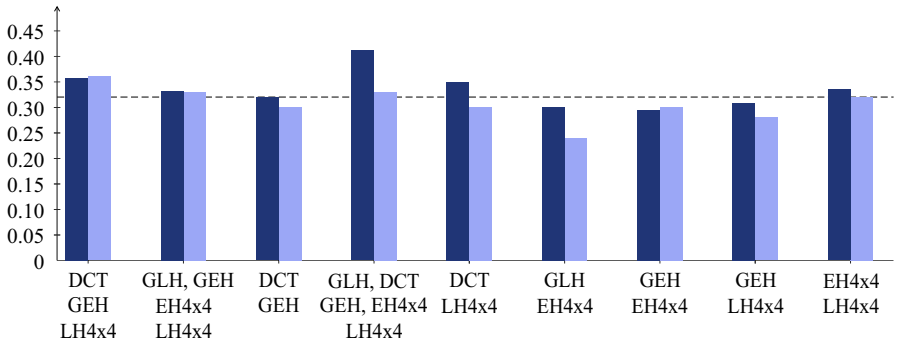


Figure 6.12: Performance of feature combinations with different fusion strategies in terms of f_1 score. The dark bars represent results with late fusion and the brighter bars are results with early fusion. In the majority of cases late fusion yields better performance than early fusion. Note that not all feature combinations improve performance compared to the best single feature (horizontal dashed line).

Feature	Similarity measures		
	χ^2 distance	L2 distance	Cosine similarity
GEH	0.28	0.28	0.24
EH4x4	0.32	0.32	0.30
Combination	0.41	0.39	0.38

Table 6.3: f_1 values for different similarity measures for the best local and best global feature and for the best feature combination.

features, and apply a linear SVM. We observe, that the χ^2 distance slightly outperforms L2 distance and Cosine similarity with all evaluated features and the feature combination. The L2 distance performs comparably well to the χ^2 distance only with single features (GEH and EH4x4). The Cosine similarity performs suboptimal in all three experiments. We select the χ^2 distance for the further experiments.

Intermediate feature kernel lag

We further evaluate the kernel lag for the creation of the intermediate features. For this purpose, we employ the χ^2 measure, use a full feature selection kernel, and a linear SVM. We evaluate the kernel lag for the best global feature, the best local feature, and the best feature combination.

We observe that a larger kernel lag generally leads to a better result (see Table 6.4). A lag of 10 performs better in any case than a lag of 6. This is especially true for

Feature	Kernel lag L		
	$L = 6$	$L = 10$	$L = 15$
GEH	0.26	0.28	0.24
EH4x4	0.29	0.32	0.32
Combination	0.30	0.41	0.48

Table 6.4: f_1 values for different kernel lags for the best local and best global feature and for the best feature combination. Note that the actual kernel size is $2L + 1$.

the best feature combination with late fusion where the increase of the lag yields an improvement of more than 25% (from 0.3 to 0.41). The increase of the lag from 10 to 15 further improves the performance of the feature combination. For the best single and the best global feature however the increased lag does not improve performance.

Intermediate feature selection

We observe in the experiments that greedy feature selection as proposed by Cooper et al. in [47] in most cases decrease performance. While the result with the best performing single feature (EH4x4) is relatively stable, the result for the feature combination decreases significantly (by 7%) when greedy feature selection is applied. We conclude, that greedy feature selection is not appropriate for the historic material because the original greedy kernel has been trained using contemporary material in [47].

SVM kernels

We evaluate three different SVM kernels: a linear kernel and two polynomial kernels of order two and three. In the majority of experiments the linear kernel outperforms the polynomial kernels. The quadratic kernel achieves slightly lower performance than the linear kernel throughout the experiments. For the polynomial kernel of order three results even partly degenerate. Overall, the linear kernel is more stable in the experiments than the other kernels.

Verification

Figure 6.13 shows the performance of the best performing setup with different kernel lags L and with different median filter sizes (from 0 which means that no median filter is applied to 43). The application of a median filter significantly improves results for

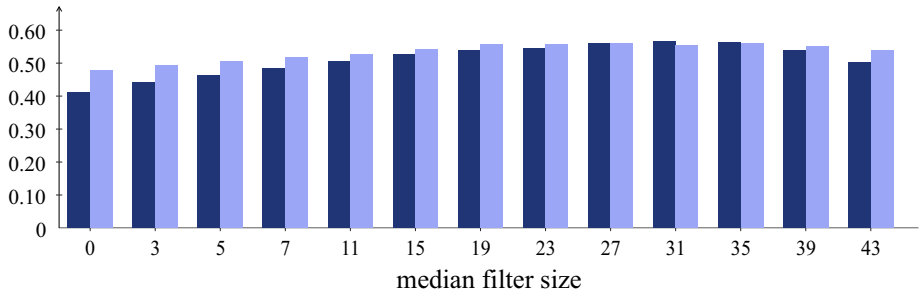


Figure 6.13: f_1 scores for the best performing setup with two kernel lags $L = 10$ (dark bars) and $L = 15$ (bright bars). The median filter improves results for both kernel lags because it filters outliers that result from classification.

both setups. The highest scores obtained for both setups with the median filter are equally good. We assume, that the median filter is a possible substitute for a larger intermediate feature kernel since the median filter as well as a larger kernel compensate for the distortions (artifacts) in the historic material. In both setups, the median filter is most effective with a window size of 31 frames. Note that this value correlates with the median length of the gradual transitions in the historic material (see Table 6.1 in Section 6.1).

We have further analyzed the performance of the two verification procedures (begin-end matching and KLT-verification). Both procedures do not improve retrieval performance. One reason is that the feature trajectories obtained by the KLT feature tracker often break off due to the large amount of noise in the data. That means that false positive detections cannot be identified, because there are no trajectories that persist through the entire sequence. Instead the trajectories break off even when no gradual transition occurs. This contradicts with the assumptions made for KLT verification (see Section 6.3.4).

Dependence on training data

We investigate the influence of the training data on the results. We compare the results for three different training sets. The first two training sets employ randomly selected training data. We evaluate if the features deliver comparable results for both sets and thereby if the performance is independent from the training data set. Note that these training sets do not necessarily contain samples of all gradual transition types. The third training set is manually selected and contains frames from *all* gradual transitions

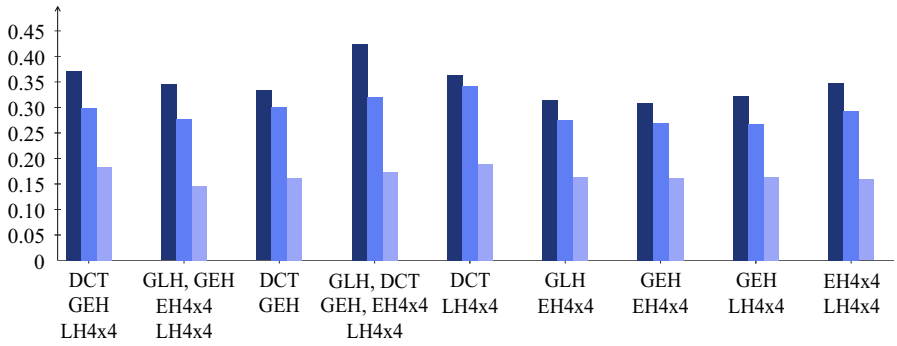


Figure 6.14: The performance (in terms of f_1 score) for three different training data sets. The dark and the medium bars correspond to results for the first and second randomly selected training set. The bright bars represent results for the manually selected data set. The training data influences the method's performance. However, the best feature combinations yield the best results consistently.

types. With the third training data set we evaluate if a manual optimized selection of the training data outperforms the random selection.

We observe that the results are not independent from the training data set (see Figure 6.14). However, the results are consistent over all evaluated feature combinations. The three best feature combinations we identified above are the three best combinations in each of the experiments. Furthermore, we observe that the manual selection of training data (that includes examples of all gradual transition types) does not improve the method's performance (see Figure 6.14). This indicates that the presented approach is able to detect gradual transitions even if no or only incomplete samples of that transition type are included in the training set.

6.5.2 Contemporary film material

The goal of the experiments with contemporary material is to demonstrate the general validity of the approach. Table 6.5 shows a comparison of the best result obtained in the experiments with those of the TRECVID 2006 shot boundary detection task (only the results of unified approaches were selected). The results show, that the performance of the approach lies in the range of the TRECVID results. Note that the method has not been optimized for contemporary material.

In the following we investigate the performance of different components of the method (single features, feature fusion, kernel lags, and median filtering) and investi-

Approach	f_p	f_r	f_1
TRECVID 2006 best (unified)	0.80	0.87	0.84
TRECVID 2006 mean of all (unified)	0.69	0.79	0.72
Proposed approach	**0.52**	**0.62**	**0.56**
TRECVID 2006 worst non-zero (unified)	0.32	0.80	0.46

Table 6.5: Comparison of the approach's performance to TRECVID results.

Abbreviation	Feature
GUH	Global histogram of the U channel
GVH	Global histogram of the V channel
GRH	Global histogram of the R channel
GGH	Global histogram of the G channel
GBH	Global histogram of the B channel
UH4x4	Local histogram of U channel extracted from 4 blocks.
VH4x4	Local histogram of V channel extracted from 4 blocks.
RH4x4	Local histogram of R channel extracted from 4 blocks.
GH4x4	Local histogram of G channel extracted from 4 blocks.
BH4x4	Local histogram of B channel extracted from 4 blocks.

Table 6.6: Color features employed in this study and their abbreviations.

gate differences in behavior of the approach between historic material and contemporary material.

Single features

Since the contemporary material contains mostly color video, we add color features in the evaluation. For this purpose, we compute a histogram for each RGB color channel. Additionally, we transform the frames into the YUV color space and extract additional histograms for the U and V channel. The color features together with their abbreviations are summarized in Table 6.6.

Figure 6.15 shows the performance of single features. We observe, that the global features perform better than the local (block-based) ones in most of the cases. The best performing global as well as the best performing block-based feature is the histogram based on the green color channel (GGH, GH4x4). For luminance histograms (GLH, LH4x4) and the red color channel histograms (GRH, RH4x4) the performance is only

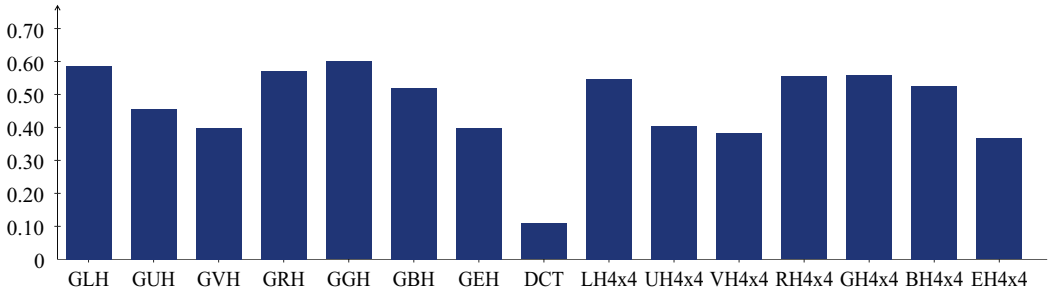


Figure 6.15: Performance of single features (f_1 score) with contemporary material. Additionally to the features employed for historic material (see Table 6.2) we extract global and block-based YUV and RGB histograms.

marginally lower than for the green color channel histograms. The features derived from the blue color channel yield significantly lower results. We assume that this performance difference is explained by the proportion of red, green, and blue in the luminance Y: $Y = 0.299R + 0.587G + 0.114B$. The green channel contains most of the luminance information; the red and green channel combined contain almost 90%. Blue has the smallest influence.

From the experiments we observe that (pure) color information seems to be of limited importance, since the U and V histograms (which do not contain intensity information like the histograms from the RGB channels) perform poorly. In contrast to the experiments with historic material, the DCT coefficients and the edge histograms (GEH, EH4x4) yield poor performance. The best features in the study are the global luminance histogram and the global histogram of the green channel.

Feature fusion

Early fusion is clearly outperformed by late fusion in our experiments with contemporary material. Consequently, we focus on late fusion in the following. Figure 6.16 depicts the results for late fusion with contemporary material. We observe, that no feature combination improves the result compared to the best single feature (GGH). The feature combination that is closest in performance, contains the global histograms of the red and green channel. This means that the information in the red channel weakens the result (the performance of the combined red and green channel histograms is lower than that for the green channel histogram alone). The same applies to the blue channel. The combination of the global R, G and B histograms (GRH, GGH,

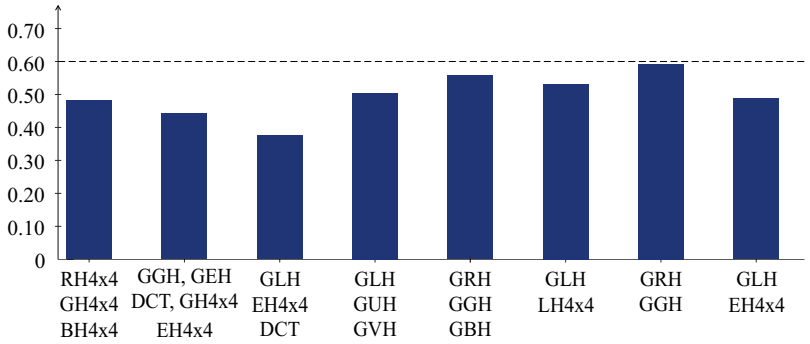


Figure 6.16: For contemporary material the combination of features (with late fusion) does not improve performance (in terms of f_1 score). The horizontal dashed line represents the performance of the best single feature.

GBH) is also outperformed by the green channel histogram. Overall, we observe that a combination of features has no benefit on the method's performance for contemporary material.

Intermediate feature kernel lag

The standard kernel lag L has a value of 6. The experiment with the global luminance histogram resulted in $f_1 = 0.588$ (see Figure 6.15). We conduct two further experiments with values for $L = 4$ and $L = 10$. The resulting f_1 values are 0.570 and 0.565. We observe, that the kernel lag has no significant influence on the result. Even the smallest kernel performs well. We assume, that this is due to the short mean duration of the gradual transitions in contemporary material (see Table 6.1).

Verification

Figure 6.17 shows the effect of a median filter with different sizes on the performance. In contrast to the application of the median filter on historic material, we achieve no performance improvement. Since the contemporary material does not contain artifacts as the historic material, we conclude that the median filter is not necessary for high-quality material.

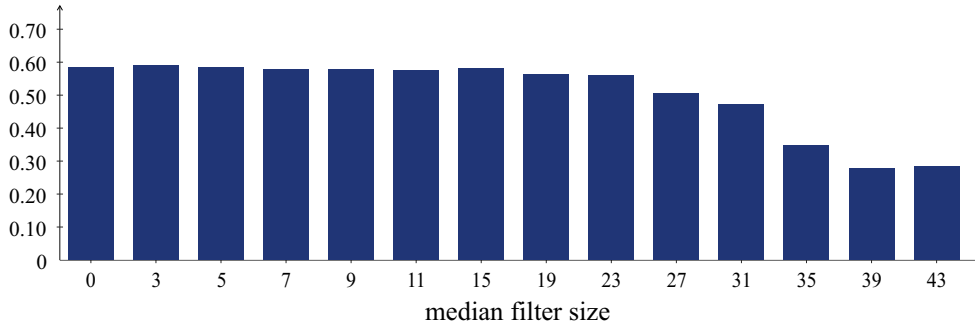


Figure 6.17: The application of the median filter has no positive effect on the f_1 score for contemporary material.

6.6 Summary

In this chapter we have presented and evaluated a method for the detection of gradual transitions in historic film material. The method follows the approach of [47] and extends it by robust features, a robust classifier and a verification step to remove outliers which is necessary for highly distorted film material.

We perform a systematic evaluation of the method and its components for historic and contemporary material. We evaluate different types of features, similarity measures, feature combinations, feature selection, and different system parameters. Additionally, we compare two different schemes for feature fusion in the framework. Different methods for verification based on motion, begin-end matching, and temporal median filtering are evaluated. The main findings from this evaluation are:

- Due to flickering and brightness variations in the historic material different features than for contemporary material are required. Global and local edge histograms robustly describe structure and shape information in the visual signal and are well-suited for this purpose.

- In historic material local edge histograms even outperform luminance histograms. In contrast to this, in contemporary material luminance histograms outperform edge histograms. Additionally, we observe that color features have only limited influence on performance.

- The combination of more than one feature with late fusion significantly improves the results for historic material. Late fusion generally outperforms early fusion. In contemporary material both, early and late fusion do not improve performance.
- For historic material a larger kernel for the intermediate features are necessary than for contemporary material because of the longer duration of gradual transitions in historic material.
- Due to the artifacts in the material, classification produces numerous outliers and gaps. A temporal median filter with a size close to the mean length of the gradual transitions is well-suited to remove outliers and fill gaps. The simple temporal median filter is more effective in removing false detections than more complex approaches based on motion (KLT trajectories) and begin-end matching.

The results obtained for gradual transition detection ($f_1 \approx 0.56$) are significantly lower than those for shot cut detection (f_1 between 0.85 and 0.94) in Chapter 5. The reason for the lower performance lies in the duration of gradual transitions in historic material. Due to the long duration, the transitions generate only slight differences in successive frames which make them difficult to detect. For the detection of long transitions the size of the analysis window (kernel lag) has to be adapted accordingly. However, the longer the analysis window the higher the probability that a camera or object motion is confused with a gradual transition. Additionally, the likelihood that artifacts occur and disturb the detector increases for large windows. The experimental study presented in this chapter shows that gradual transition detection for historic material is still a challenging task that is far from being solved.

Chapter 7

Segmentation of Scenes

In Chapters 5 and 6 we have presented methods for the detection of abrupt shot cuts and gradual transitions. These methods enable the segmentation of a film into its basic structural units, the shots. The next higher temporal unit in a film are *scenes*. A scene principally represents a series of consecutive semantically related shots. Scenes are an important structural as well as semantical higher-level concept in films. On the one hand, scenes structurally partition a film into parts with different topics and on the other hand, the topic of a scene usually has a particular semantic meaning important for transporting the message of a film. In this chapter, we investigate scene segmentation in the context of archive film material. In Section 7.1 we present the specific requirements for scene segmentation introduced by the archive films. We review related approaches in Section 7.2. Section 7.3 presents an extensible multimodal framework for scene segmentation that does not require a priori knowledge about compositional rules and thus is applicable to arbitrary audio-visual content. We perform a systematic evaluation of the framework (see Section 7.4) and investigate the performance and behavior of features, similarity measures, multimodality, and the framework's parameters. Results of the systematic evaluation for both archive film and contemporary film are presented in Section 7.5.

7.1 Introduction

In modern fiction films, such as Hollywood films, scenes usually depict activities related to the same dramatic incident or location [20]. This definition is not applicable

to documentaries and especially not to artistic archive documentaries. In archive documentaries, shots constituting a scene are related on a higher abstraction level. For example, in a fiction film, a scene may show two people driving in a car and talking to each other. All the shots depicting this conversation form the scene. In an archive documentary, a scene for example may consist of shots that show how electricity is brought to a village. Shots of the scene show someone installing a power line, peasants using an electrical thresher and several houses of the village with electrical lighting. All these shots are recorded at different locations and at differing times. The cohesion of the shots is generated purely on a semantic level without spatio-temporal relations. Due to the low cohesion of shots there is only little a priori knowledge (e.g. about composition rules from film grammar) that can be incorporated into the segmentation process. The most important clue for scene segmentation in archive documentaries is the repeated appearance of visually and auditory similar shots and motifs.

We present a framework for scene segmentation that solely relies on the repeated occurrence of similar auditory and visual content. The most important characteristics of the framework are:

- It allows the integration of arbitrary visual and auditory features and respective similarity and distance measures. The larger the number of features, the more different auditory and visual aspects can be taken into account for scene segmentation.
- A common fusion process combines all similarities found by different features. Since the importance of the single features is unknown a priori each feature is weighted equally during fusion.
- A two-stage process improves the quality of the segmentation. In the first stage we identify core scenes. In the second stage we refine the segmentation and determine the final scene boundaries.
- The framework is applicable to arbitrary film material (e.g. contemporary films) since it solely relies on audio and visual similarities.

In this chapter, we present the proposed framework and perform a systematic evaluation of different parameters of the framework. In the systematic evaluation we first estimate adequate similarity measures for the different features. Next, we evaluate the

performance of different features and feature selections and the benefit of multimodality for scene segmentation. Additionally, we investigate the behavior of the framework under variation of the framework's parameters. Ultimately, we investigate the benefit of the refinement stage of the framework. All experiments are carried out for archive films and contemporary films in parallel in order to allow for performance comparisons.

7.2 Related Work

Existing techniques for scene segmentation usually target at contemporary films. There are a few methods that exploit specific film editing techniques. Tavanapong and Zhou for example exploit the human perception of continuity. They construct a visual feature vector based on (key-)frame regions which are important for the human perception of continuity in narrative films. They use this feature vector at several stages of their algorithm, for details see [212]. Other scene segmentation approaches that exploit specific film editing rules were devised by Truong et al. [216]. The authors analyze film grammar to identify rules and conventions that are applied for the creation of scenes by filmmakers. These rules and conventions are then exploited for the segmentation of films into scenes.

Scene segmentation methods that work for non-narrative films, such as archive documentaries, cannot rely on specific film editing techniques and composition rules. These methods have to employ auditory and visual similarities as the only clues. First steps towards scene segmentation based on visual similarity are presented by Yeung et al. [240]. Yeung et al. introduce the complete-link method for clustering shots based on visual (color and luminance) similarity under a temporal constraint. They require shots to be temporally close to be considered as part of one cluster. Using the results of the clustering process, the authors build a scene transition graph. Individual scenes are identified by finding the cut-edges that partition the graph into disjoint sub-graphs. These sub-graphs represent the final scenes. Another method that is based on similarity and a temporal constraint is introduced by Hanjalic et al. [84]. The authors segment movies into Logical Story Units (LSUs) which are approximations of scenes. The method analyzes the visual dissimilarity of shots in a sliding time window. The dissimilarity computations rely on keyframes representing the shots. For each keyframe the average color in the L^*u^*v color space is extracted and used for the dissimilarity

computation. If the dissimilarity between two shots S_k and S_n where $k < n$, exceeds the dissimilarity threshold, a LSU break is recognized. If the dissimilarity between S_k and S_n is lower than the threshold a so called *overlapping-link* is detected and all shots inbetween S_k and S_n are grouped to belong to the same LSU containing the shots S_m where $m \in [k, n]$. Based on the principles of overlapping links and visual similarity a number of scene segmentation methods have been developed [146, 235, 251]. Wang et al. perform an iterative backward and forward search for similar shots in a video. Shot similarity is defined as the combination of visual similarity and consistent motion characteristics (motion similarity). The visual similarity of shots is expressed in terms of the maximum similarity between all combinations of the first and the last frames of two shots. The frame similarity computation is based on color histograms. Wang et al. extract the motion similarity by comparing the accumulated motion intensities of two shots. Eventually, the motion characteristics and the visual similarity are weighted and summed to arrive at the value for shot similarity. Zhu and Liu [251] integrate texture and *gray information variance* as the visual features for scene segmentation in their framework. Gray information variance is obtained by dividing the frame into $8 \times 8 = 64$ image blocks and computing the variance of the gray values within each block. Additionally, the variance of the first-level coefficients of a Haar wavelet transform and the corresponding average of the entire frame represent texture information.

There exist techniques that incorporate auditory similarity in addition to visual similarity. One such method was devised by Pfeiffer et al. [166]. They use audio and visual features to classify shots as belonging to three types of scenes: dialogs, settings and similar audio. Starting from shot boundaries they extract features for the shots and compute shot distance tables. Pfeiffer et al. distinguish background and foreground audio using a loudness feature, they argue that background sounds set the atmosphere and are less loud than foreground sounds. Additionally, they identify audio cuts measuring the changes in the distribution of frequency intensities over time. Visual features include a frontal face detector, color coherence vectors and an orientation feature. All features are input to the computation of the shot distance tables. Similarity clustering of shots is performed on the distance tables. Finally, the three different scene types are merged to obtain a scene segmentation for the entire film. Sundaram and Chang [206] propose a multimodal method that is based on the separate detection of auditory and visual scenes. First, the authors extract a set of auditory features

including cepstral flux, spectral flux, zero crossing rate, energy, cepstral features, and cochlear decomposition. Second, they compute a correlation function based on the Euclidean metric. Finally, they identify audio scene boundaries using the minima of the correlation function. The video scene boundaries are identified using color histograms and a human perception model. Eventually, Sundaram and Chang merge the lists of audio scene boundaries and video scene boundaries using a time-based nearest neighbor criterion. The merged list is the final scene segmentation.

The framework proposed in this chapter also takes visual and auditory information into account. However, in contrast to existing techniques we abstract from the two different modalities and treat all features whether they originate from the audio track or the visual signal equally. This reduces the number of assumptions necessary for the fusion and combination of information derived from the auditory and visual domain to a minimum and makes the framework generally applicable to the segmentation of audio-visual signals. Additionally, the proposed framework solely bases on auditory and visual similarities and does not require models of auditory and visual scenes.

7.3 Multimodal Scene Segmentation Framework

An overview of the proposed framework is provided in Figure 7.1. In a first step the shot boundaries (shot cuts) are detected in the visual track. For each shot, we extract a key frame and compute different visual content-based features $f_1^V, f_2^V, ..., f_N^V$ for each keyframe. From the audio track, we first extract auditory features continuously for the entire film, resulting in features $\phi_1^A, \phi_2^A, ..., \phi_M^A$. These continuous features are then aggregated for each shot as described in Section 7.3.2 resulting in shot-based auditory features $f_1^A, f_2^A, ..., f_M^A$. The audio and visual features (representative for single shots) form the input of shot grouping. The result of shot grouping are separate groupings for each audio and visual feature. Next, the individual groupings are fused together which results in a sparse segmentation of the film into core scenes. Eventually, the refinement and pruning stage determine the final scene boundaries.

7.3.1 Feature Extraction

Prior to feature extraction we locate the positions of the shot cuts in the investigated film by the approach presented in Chapter 5. Then, we select the first frame of each

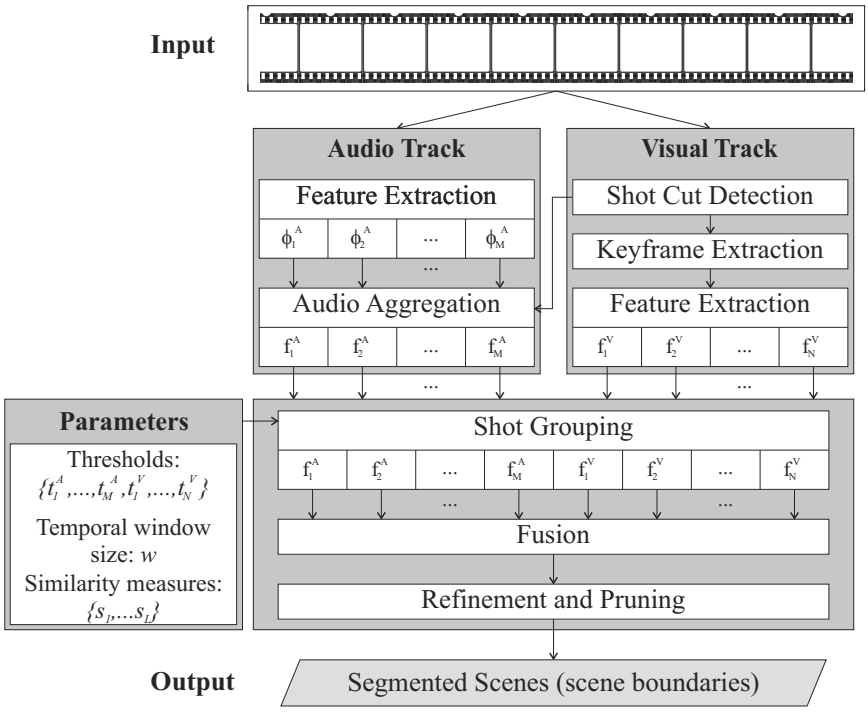


Figure 7.1: An overview of the proposed framework.

shot as a keyframe. The next computational step is the extraction of audio and visual content-based features. Arbitrary features can be integrated into the framework. For the visual description for example histograms, texture descriptors and local feature point descriptors may be employed. An important factor is however, that the extracted features represent complementary information. Complementary features enable to capture a larger spectrum of visual similarities and thus a larger variety of visual aspects from the film. The same applies to auditory features. Auditory features are extracted continuously for the entire sound track of the film. For this purpose a small analysis window is moved across the sound track and at each window position features are extracted. The auditory features have to be aggregated over entire shots in order to obtain a compact description of a shot's audio content. The aggregation of auditory features is presented in the next section.

7.3.2 Audio Aggregation

In contrast to the visual features which are extracted for a keyframe of a shot, the auditory features have been extracted for the entire audio signal of a shot. Consequently,

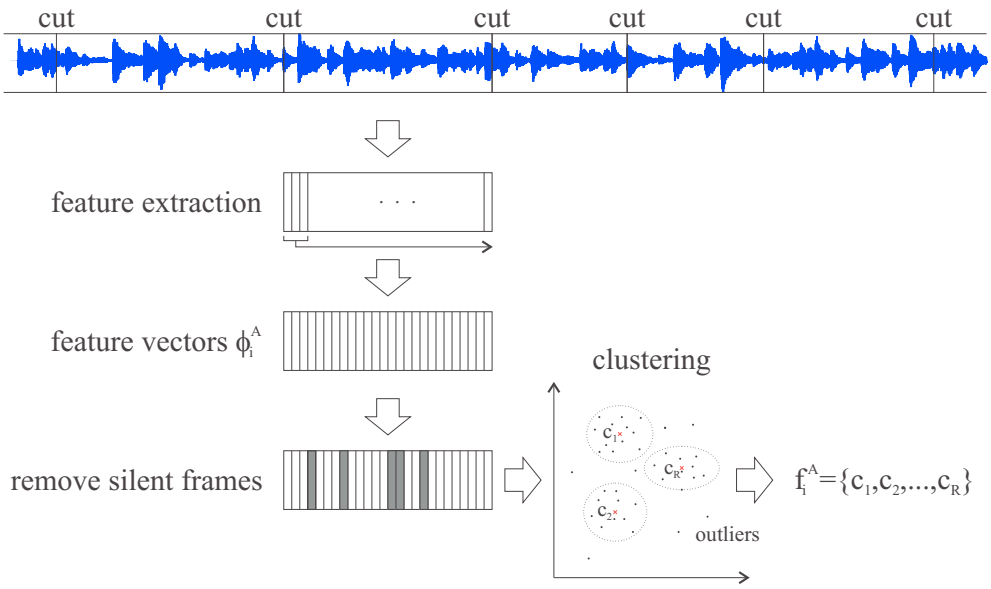


Figure 7.2: Aggregation of auditory features by clustering.

they must be aggregated over the shot in order to get a compact description of the audio content of the entire shot. Different schemes for the aggregation of audio exist. A common practice for feature accumulation is to compute statistical measures, such as mean, variance, median, etc. for each feature component over time [41]. However, the mean and variance of a feature component over an entire shot are too coarse measures to represent the audio content of an entire shot reasonably. We apply a more elaborate scheme for feature aggregation that is based on the clustering of feature vectors. An overview of the scheme is shown in Figure 7.2.

After feature extraction, we first remove feature vectors of silent audio frames. Feature vectors that capture silence do not represent discriminatory information and would distort similarity comparisons. We extract a loudness feature (specific loudness sensation, see Section 2.2.4) and sum up all components of the feature in an audio frame. The resulting value approximates the loudness in the audio frame. If the value is below a threshold, the audio frame is considered silent and is excluded from further analysis.

Next, the remaining audio frames are clustered on a shot basis. The feature vectors of all non-silent audio frames of a shot are input to mean-shift clustering [71]. Mean-shift generates a variable number of clusters and returns the respective cluster

centers $c_1, c_2, ..., c_R$ which have the same dimension as the input feature vectors. We employ the returned cluster centers as representation of the audio content of a shot. The presented aggregation scheme is applicable to arbitrary auditory features that can be extracted for short audio frames (short-time features).

7.3.3 Shot Grouping and Fusion

We employ the *overlapping links method* for shot grouping [84]. The overlapping links method compares subsequent shots in a temporal analysis window to each other based on a particular feature and similarity measure. We limit the similarity computations between the shots to a time window of several preceding and following shots. In literature this is often referred to as temporal constraint, basically the temporal constraint ensures that shots that are temporally far apart are not assigned to the same scene. This constraint is necessary because otherwise two audio-visually similar shots, one at the beginning and one at the end of a film would result in severe under-segmentation. There would be a single scene comprising the greater part of the film.

The framework allows for the integration of arbitrary similarity and distance measures for the comparison of shots represented by audio or visual features. We evaluate a number of similarity measures in the context of the framework, see Section 7.4.1. Similarity measures can be directly applied to the visual features which comprise *one* feature vector per shot. For the aggregated auditory features similarity measures cannot be directly applied since the features contain *several* vectors per shot. We propose an appropriate scheme for the comparison of aggregated auditory features in the following.

The similarity between two shots h_u and h_v represented by the i-th auditory feature (with P and Q cluster centers, respectively) $f_i^A(h_u) = \{c_{u_1}, c_{u_2}, ...c_{u_P}\}$ and $f_i^A(h_v) = \{c_{v_1}, c_{v_2}, ...c_{u_Q}\}$ is the maximum similarity between all possible pairs of cluster centers of both shots:

$$sim_i^A(u, v) = \max_{\substack{p=1,...,P \\ q=1,...,Q}} s_j(c_{u_p}, c_{v_q}), \tag{7.1}$$

where s_j is the similarity measure. The proposed comparison scheme enables the application of arbitrary similarity and distance measures for the comparison of aggregated auditory features.

After assigning each feature an adequate similarity measure, the overlapping links method is applied. The principle of the overlapping links method is the following: First,

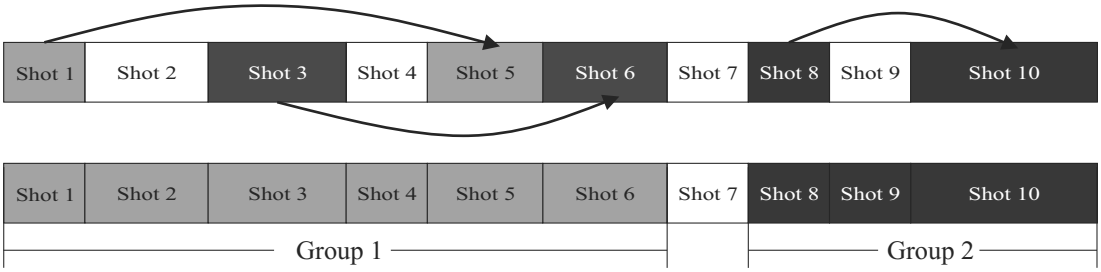


Figure 7.3: The similarity computations show which shots belong together (indicated by the arrows and shading). Shots that have no similarities (e.g. shot 9) but exist between matching shots are assigned to the group defined by the surrounding matching shots (shot 8 and shot 10).

for each feature in the framework a threshold is defined for similarity comparisons. If the similarity between two shots for a particular feature f_i exceeds threshold t_i, the two shots are considered similar given feature f_i. Next, we compare all shots inside a temporal window of w shots with each other and *group* shots with a similarity higher than the threshold together. Additionally, shots that are between two similar shots are assigned to the group of the surrounding similar shots. Figure 7.3 illustrates this process. Note that the size w of the temporal analysis window is at least 5 in this example to allow establishment of the first link between shot 1 and shot 5.

We apply the overlapping links method for each audio and visual feature separately and obtain one shot grouping for each content-based feature. The different groupings are combined into a first segmentation by a simple fusion scheme: We merge the different segmentations using the set operation *union*. For an illustration see Figure 7.4. Consider for example shots 2 to 5 for the three groupings in Figure 7.4. Shot 3 and shot 4 are part of one group according to feature f_1 (first row in Figure 7.4). Shots 2 and 3 are part of one group according to feature f_2 (second row in Figure 7.4) and shots 4 and 5 are part of the same group according to feature f_3. The union operation combines these overlapping sets of shots into one segment that contains the shots 2, 3, 4, and 5. The output of this procedure are the so called *core scenes* (last row in Figure 7.4). We call the shots that are not assigned to any core scene *loose shots* (e.g. shots 6 and 7 in Figure 7.4).

An important issue in the context of the selected fusion scheme is the dependence on the preceding shot grouping process. If the shot groupings are too tolerant, the union

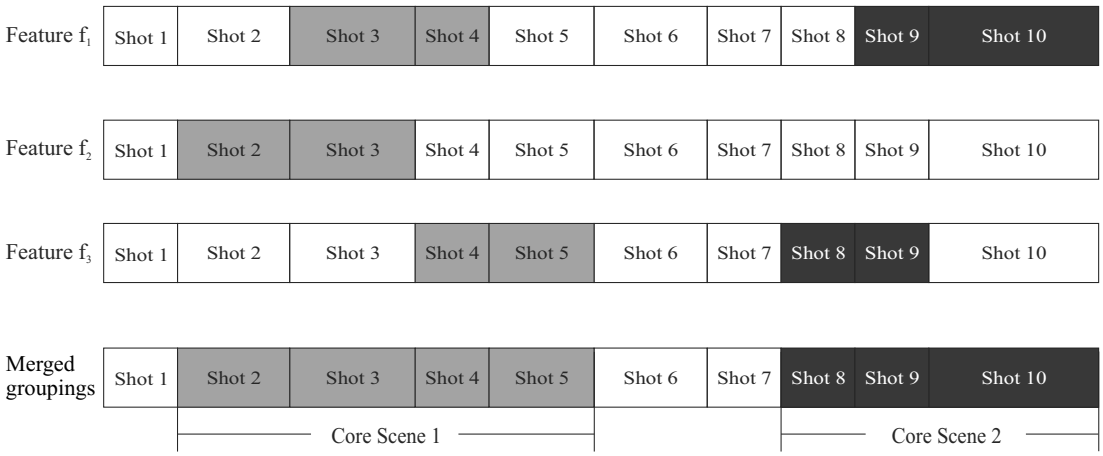


Figure 7.4: The three groupings obtained by the content-based features f_1, f_2, and f_3 are combined. The result of this combination are the core scenes.

operation generates an under-segmentation of the film. The shot grouping should be strict enough so that shots are grouped together (considered similar) only if they have a *significant* similarity. A stricter grouping can be obtained easily by increasing the similarity comparison thresholds.

In this investigation, we focus on performance evaluations of different features and feature combinations in the framework. We therefore do not implement a sophisticated fusion scheme. The framework however allows the integration of arbitrary, more complex schemes.

7.3.4 Refinement and Pruning

After fusion we obtain core scenes and numerous remaining loose shots inbetween the core scenes. This over-segmentation is reduced by assigning the loose shots to neighboring core scenes. The assignment of loose shots to core scenes is again performed solely based on visual and auditory similarities and does not require a priori knowledge. Additionally, the assignment scheme minimizes the likelihood that a loose shot is assigned to the wrong (less similar) neighboring core scene. The principle of the assignment of loose shots is illustrated in Figure 7.5. First, a loose shot (h_{L1} in the Figure) is compared to all shots from the neighboring core scenes A and B. Next, the maximum similarities between h_{L1} and all shots from core scenes A and B are estimated based on a feature f_i. If the maximum similarity with core scene A is higher than for B by

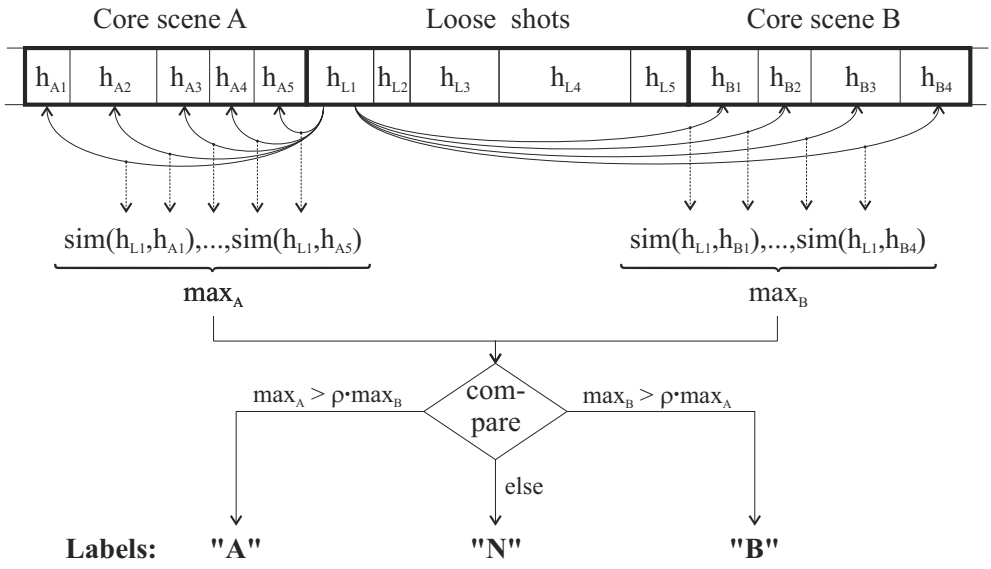


Figure 7.5: The process for the labeling of a loose shot.

a factor ρ, the loose shot is labeled "A". In the opposite case the loose shot is labeled "B". If both conditions are not fulfilled, the shot cannot be assigned uniquely to one of the core scenes and gets the neutral label "N". The factor ρ assures that a loose shot is only assigned to a core scene if the shot is sufficiently more similar to one core scene than to the other. By this approach, we avoid the assignment of a loose shot if no safe decision for either core scene can be made.

Figure 7.5 illustrates the labeling of one loose shot for *one* content-based feature only. If several features are employed in the framework, we repeat the entire process for each feature and obtain a label for each feature. The obtained labels for each feature may be heterogeneous. We apply a majority voting on the labels. If there are more decisions for scene A then for scene B the loose shot is labeled "A". In the opposite case, we label the shot with "B". If the voting is undecidable (same number of labels "A" and "B") the shot is labeled neutral ("N").

Loose shot assignment as described above is performed for all loose shots between two core scenes. The result is one label for each loose shot. The final goal of the refinement step is to find a position in the set of loose shots where we can join the two neighboring core scenes. An important constraint in finding this position is to minimize the likelihood of a wrong assignment of a labeled loose shot. If the sequence of labels for

five loose shots is for example "AAABB", the assignment of the loose shots is possible without error. The neighboring core scenes are simply joined after the third loose shot: "AAA|BB". In general however the sequence of labels cannot be split without errors, which is for example the case in the sequence "AABBA". There is no split position where all loose shots can be assigned according to their label.

Finding an optimal split position in the label sequence is a simple optimization problem. We linearly scan the sequence and split the sequence at each possible position. For each split position we assign costs to the loose shots. If the label of a loose shot matches the neighboring core scene, we assign costs of 0. If the label of a loose shot does not match the neighboring core scene, we assign costs of 1. If a loose shot has label "N" it always gets assigned costs of 0. This assures that shots labeled neutral have no influence. The overall costs for the current split-position is simply the sum of costs over all loose shots (which is the number of falsely assigned labels with the weights selected above). We evaluate the overall costs of each possible split position and select the position with the minimum costs and thus with the minimum number of wrong assignments.

The final result of the refinement is a dense segmentation of a film into scenes. Refinement reduces over-segmentation by removing gaps with loose shots between core scenes. We further reduce over-segmentation in a *pruning* step that removes scenes which contain less than four shots since such segments are too short to represent a scene. After removing these short scene candidates the resulting gap between the neighboring scenes is closed by positioning the new boundary between both scenes in the middle of the gap.

7.4 Systematic Evaluation

The presented framework is a well-suited basis for the investigation of different research questions in the context of automatic scene segmentation. We perform a systematic evaluation of the different components and parameters in the framework in order to answer the following questions:

1. How different is scene segmentation in archive film material from scene segmentation in contemporary material? Are the optimal features, feature selections, parameter values, etc. similar for both types of material or are there differences?

The latter would indicate that archive film material requires specialized methods for scene segmentation.

2. How do different features and feature selections perform? Do the more sophisticated SIFT features outperform simple features, such as intensity histograms and edge histograms? Which is the best single feature? Does the combination of different features improve results?

3. Does scene segmentation benefit from multimodal (audio-visual) processing or do the two modalities impede each other? How well do the two modalities perform on their own?

4. How large is the influence of the similarity and distance measures? Which similarity measures are most appropriate for the evaluated features?

5. How does the temporal window size influence the performance of scene segmentation? How does the choice of the similarity comparison thresholds influence results? Are the optimal threshold values similar for different films? Are there dependencies between the temporal window size and the similarity comparison thresholds?

6. Do refinement and pruning improve the scene segmentation? How large is the performance gain?

Questions from 2 to 6 are investigated for archive film material and contemporary film material separately in order to answer the questions in 1.

7.4.1 Experimental Setup

We employ a number of archive and contemporary films for the systematic evaluation. The films and their characteristics are summarized in Table 7.1 which provides the film name, the name of the director, the year of release, the available modalities (Mod.), the number of shots and the number of ground truth scenes. In the evaluation we distinguish between three groups of films: (i) *archive silent films*, (ii) *archive sound films*, and (iii) *contemporary films*. The groups are analyzed separately and the results are later compared. In the group of archive silent films we additionally incorporate the two archive sound films ("Enthusiasm" and "Three Songs of Lenin"), however without

Film name	Director	Year	Mod.	#shots	#scenes
The Eleventh Year	D. Vertov	1928	V	660	25
Man with a Movie Camera (V)	D. Vertov	1929	V	1782	55
Enthusiasm (Restored)	D. Vertov	1931	V+A	612	30
Three Songs of Lenin	D. Vertov	1934	V+A	817	37
Top Gun (TGun)	T. Scott	1986	V+A	2121*	52
Pulp Fiction (PulpFn)	Q. Tarantino	1994	V+A	1276	60
Run Lola Run (Lola)	T. Tykwer	1998	V+A	1654	97

Table 7.1: Archive and contemporary films employed in the evaluation and their characteristics. The films above the dashed line are archive films, while the films below are contemporary films.
* The number of shots was evaluated automatically since no ground truth was available.

analyzing their soundtrack. This increases the size of the group and allows a more robust evaluation.

Prior to systematic evaluation we select audio and visual features that have shown to be robust and discriminatory in the context of the investigated film material. The features and the parameters necessary for their computation are summarized in Table 7.2. We evaluate three visual features: (i) block-based intensity histograms (BBH), (ii) MPEG-7 edge histograms (EH) and (iii) SIFT keypoints (SIFT). For the BBH we divide the image uniformly into 16 blocks and compute a 16 bin intensity histogram for each block. A partition into 16 blocks represents a good tradeoff between expressiveness of the features and robustness (e.g. against motion). The BBH compactly summarizes the gray-value distribution while it preserves a certain amount of spatial information. The EH feature describes the number of edges in horizontal, vertical, 45° and 135° direction as well as non-directional edges. It represents the texture of the frame independent of its intensity values. Again we split the image uniformly into 16 blocks and compute one edge histogram for each block. Additionally, we compute a global edge histogram for the entire frame. All resulting edge histograms are concatenated into one feature vector. Finally, SIFT provides the positions and compact descriptions of salient points in the frame. We extract SIFT features using Vedaldi's and Fulkerson's VLFeat [229] library with default parameters. We filter out SIFT keypoints with small scales, since they show low expressiveness.

Modality	Feature	Parameters
Visual	BBH	16 sub-images, 16 bins
Visual	EH	16 sub-images + global histogram, 5 edge directions
Visual	SIFT	default parameters by [229], neglect scales ≤ 6
Audio	BFCC	frame size 30ms, overlap 20ms, 13 coefficients

Table 7.2: Audio and visual features in the evaluation and their parameters.

The three selected visual features represent orthogonal information, namely intensity, edges, and salient keypoints. Consequently, by combining the features in our framework we are able to capture a larger spectrum of visual similarities and thus a larger variety of visual aspects can be considered for scene segmentation. Furthermore, the features have the potential to mutually compensate for weaknesses. For example in situations where keypoints can hardly be detected (e.g. in a shot that mainly shows homogeneous areas like sky, the intensity histograms may provide a more accurate description).

Additionally to the visual features, we extract Bark-frequency cepstral coefficients (BFCCs, see Section 2.2.6) for the entire audio track of the film. BFCC are short-time features and are extracted for analysis frames of 30 ms. BFCCs represent the coarse spectral envelope (frequency distribution) of the underlying audio signal. Auditory features in the framework are aggregated over entire shots, in order to get descriptions that allow the comparison of different shots (see Section 7.3.2).

In a systematic evaluation the number of possible system configurations to be evaluated grows exponentially with the number of parameters and degrees of freedom. We first evaluate the performance of the similarity measures for the features. Based on this evaluation, we select the most appropriate similarity measure for each feature to reduce degrees of freedom and to keep the number of possible system configurations in a reasonable range.

An exception is the SIFT feature. The matching for the SIFT features is computed as proposed by Lowe [133]. The similarity between two shots based on SIFT is expressed by the number of matching keypoints. We add a spatial constraint to the matching procedure that restricts the maximum distance of two compared SIFT feature points.

For all other features (BBH, EH, and BFCCs) we have to evaluate an appropriate similarity measure. Distance and similarity measures evaluated comprise of L1 and

L2 distance, Chebyshev distance, canberra distance, χ^2 distance, Cosine similarity, Pearson correlation and histogram intersection. Distance measures are transformed into similarity measures by inverting them linearly. Details on the different measures are given in Section 2.4. For similarity comparisons in our framework we define four thresholds: t^V_{BBH}, t^V_{EH}, and t^V_{SIFT} for the visual features and t^A_{BFCC} for the auditory feature.

We evaluate the performance of each similarity measure (in dependence of the features) systematically. For each combination of feature and similarity measure, we perform scene segmentation and validate the results against the ground truth. For scene segmentation, we have to set at least two parameters: the comparison threshold for the similarity measure and the temporal window size w. We *vary* both parameters in order to allow for an evaluation of the similarity measures *independent* from the parameters. This means that for each combination of feature and similarity measure several scene segmentations are computed, each with different values for w and the comparison threshold. Refinement and pruning is skipped in this evaluation. Ultimately, we select *the* similarity for a given feature that achieves (i) high f_1 scores over all films and (ii) high f_1 scores for a broad range of parameter values. The second condition demands for a certain degree of independence from the parameter values. If a similarity measure yields a high score for only *one* particular combination of w and a comparison threshold the result is not representative and the similarity measure is neglected. The final result of the evaluation is the identification of an adequate similarity measure for each feature.

The goal of the following systematic evaluation is to analyze the influence of the single parameters, the dependencies between the parameters, and the effectiveness of particular processing steps. The number of possible system configurations is near infinite since the framework incorporates a number of numeric parameters (e.g. the similarity comparison thresholds). We subsample the value range of all numeric parameters which reduces the number of system configurations significantly. All evaluated parameters together with their possible values are shown in Table 7.3.

The first four parameters are flags that control the inclusion or exclusion of single audio and visual features for the generation of different feature and modality selections. The next four parameters are the similarity comparison thresholds of the four features. The minimum and maximum values for the thresholds are evaluated manually. The

Parameter	Possible parameter values
useBBH	true, false
useEH	true, false
useSIFT	true, false
useBFCC	true, false
t^V_{BBH}	0.5, 0.57, 0.64, 0.71, 0.78, 0.85
t^V_{EH}	0.5, 0.55, 0.61, 0.66, 0.71, 0.77, 0.82, 0.87, 0.93, 0.98
t^V_{SIFT}	5, 10, 15, 20, 25
t^A_{BFCC}	0.7, 0.76, 0.81, 0.87, 0.92, 0.98
w	5, 10, 15, 20, 25, 30, 35, 40, 50
doRefinement	true, false
doPruning	true, false

Table 7.3: Parameters and their possible values in the systematic evaluation.

values between minimum and maximum are linearly spaced. Parameter w is the size of the temporal analysis window for shot grouping (in units shots). The last two parameters control whether or not refinement and pruning are applied.

The systematic evaluation is performed as follows: We successively change the parameter values to generate novel system configurations and systematically evaluate all possible configurations. The systematic evaluation is performed for each film in Table 7.1. Results are aggregated for the three investigated groups of films (archive silent films, archive sound films, and contemporary films) to enable a comparison of the groups.

Results are expressed in terms of recall, precision, and f_1 score. Recall represents the number of correctly retrieved scene boundaries divided by the total number of scene boundaries in the film (according to the ground truth). Precision represents the portion of correctly retrieved scene boundaries in the set of all retrieved boundaries. We integrate a tolerance of 4 shots into the evaluation according to [83]. The f_1 score is computed as described in Section 2.6.3

The systematic evaluation produces a large number of retrieval results for each film (in the order of 10^5 for sound films and 10^4 for silent films). A significant portion of system configurations does not produce meaningful results, because the combination of the selected parameter values performs poor. For the evaluation, we first aggregate the

results for the three investigated groups of films and then we select only the best 5% of the results. This subset of the results represents the set of system configurations with the highest performance. In the following, we call this set the "*quasi-optimal result set*". We consider all results in this set as equivalently good which allows to analyze the distribution of parameter values in this set. If for example, most system configurations in such a result set incorporate a feature f_i, this indicates that this feature is beneficial to a high degree. Consequently, this feature is well-suited for scene segmentation of films of the current group and should be selected.

7.5 Experimental Results

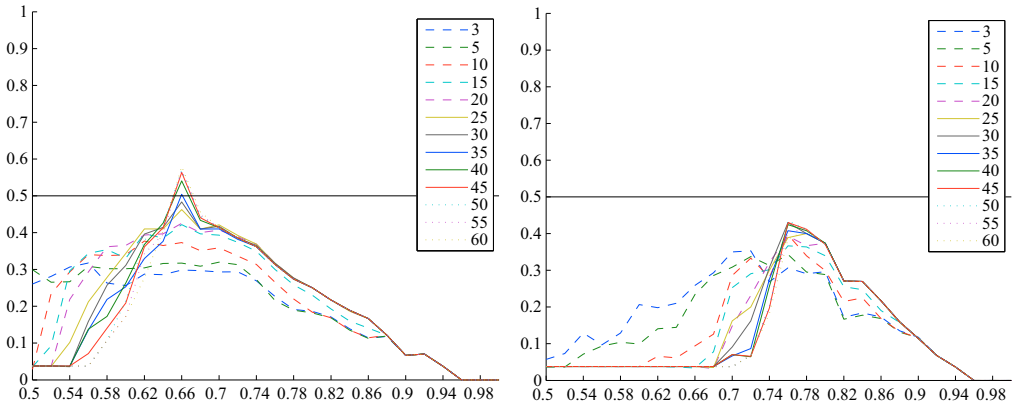
In this section, we discuss the results of the experiments. We first discuss the performance of the similarity measures and then present the results of the systematic evaluation of features, feature combinations, parameters, and processing steps. The organization of the section follows the computation process in the framework.

7.5.1 Similarity Measures

We evaluate the distance and similarity measures for BBH, EH, and BFCC as described in Section 7.4.1. The evaluation reveals significant performance differences between the similarity measures. Figure 7.6 shows the performance figures for the BHH feature with L1 and L2 distance, respectively. Performance is shown for all evaluated parameter combinations. The value of the comparison threshold runs along the x-axis. The y-axis represents the f_1 score and each curve in the diagram represents a different window size w.

We observe that the L1 measure yields higher f_1 scores than the L2 measure. Additionally, the L1 measure yields higher scores for a larger number of different parameter combinations. This can be observed from the shape of the curves for the L1 measure which are broader and higher than for the L2 measure. Since the L1 measure yields good results for all films and outperforms the other evaluated similarity measures in most cases, we employ the L1 measure for similarity comparisons of the BBH in the following.

For the EH we observe even stronger performance differences between the different similarity measures. We present an example in Figure 7.7 which shows the performance



(a) L1 measure

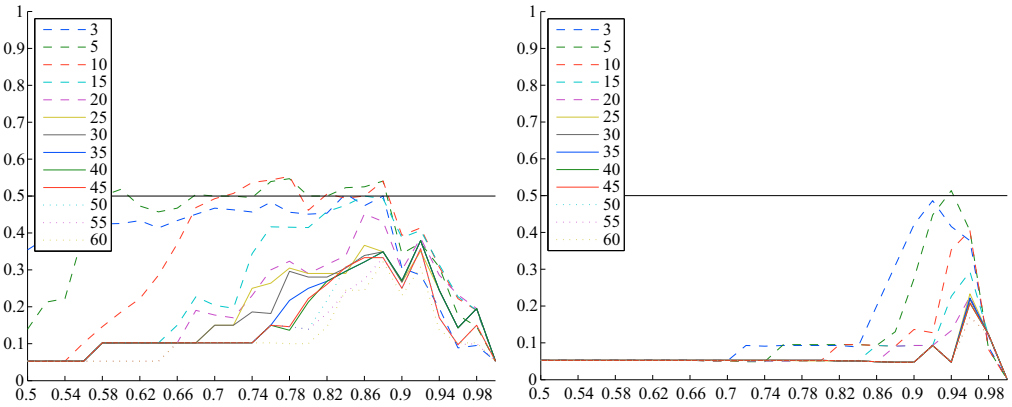
(b) L2 measure

Figure 7.6: Performance of L1 and L2 measure for BBH and film "Top Gun". The x-axis represents the comparison threshold (from 0.5 to 1), the y-axis the f_1 scores. The different curves represent experiments with different window sizes w (from 3 to 60).

of the Cosine similarity and the χ^2 distance in the context of EH. The Cosine similarity yields higher f_1 scores nearly independent from the comparison threshold (e.g. for window sizes from 3 to 10). The corresponding curves show higher f_1 scores for a broad range of threshold values. The χ^2 distance yields higher f_1 scores only for two distinct combinations of w and the comparison threshold ($w = 3$, $t^V_{EH} = 0.92$ and $w = 5$, $t^V_{EH} = 0.94$). For all other combinations results degenerate. This shows that the χ^2 distance is not appropriate for the EH feature. The Cosine similarity yields the most robust results for the EH feature over the evaluated films. As a consequence, we select the cosine similarity for the EH feature.

Finally, we evaluate the similarity measures for the auditory feature BFCC. Again we observe strong performance differences. Figure 7.8 shows the performance figures of BFCC with Cosine similarity and L2 measure. While the L2 measure obtains higher f_1 scores for different combinations of w and the similarity threshold, the Cosine similarity obtains higher f_1 scores only for one "magic" threshold value of 0.98. This means that the cosine similarity is highly dependent on the data (film) and consequently no good choice for scene segmentation of arbitrary films.

We further investigate the behavior of the similarity measures by analyzing the similarity judgments made by both measures. We compute a matrix that contains the pair-wise similarities of all shots in a film for each similarity measure. The two resulting

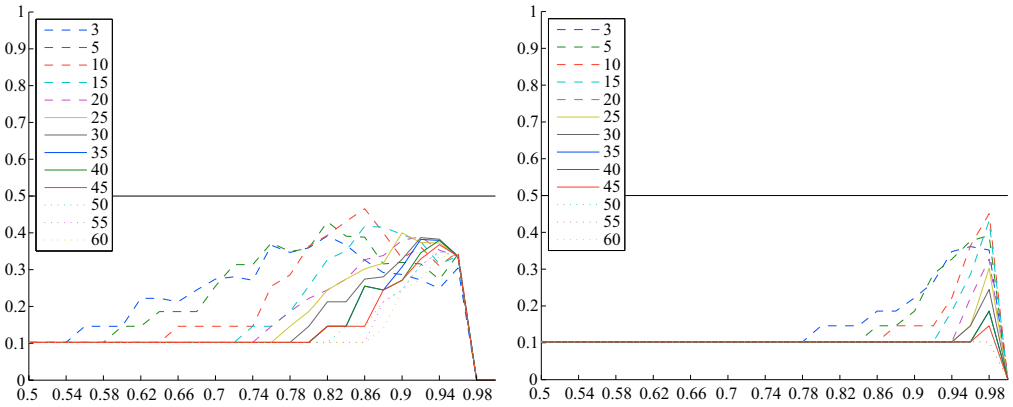


(a) Cosine similarity (b) χ^2 distance

Figure 7.7: Performance of Cosine similarity and χ^2 distance for EH and film "Three Songs of Lenin". The x-axis represents the comparison threshold (from 0.5 to 1), the y-axis the f_1 scores. Each curve represents a different window size w (from 3 to 60).

matrices are shown in Figure 7.9. We observe that the similarity matrix generated with the L2 measure has a larger variance than the matrix obtained by the Cosine similarity. In the similarity matrix of the cosine measure nearly all similarities lie in the range $[0.9, 1]$. This is also the value range in Figure 7.8(b) where the only higher f_1 scores are obtained. The generally high similarity judgments obtained by the cosine similarity and the low variance show that the cosine similarity lacks discriminability for the feature. Nearly all shots in the film are considered highly similar to each other. The judgments of the L2 distance show a higher variance of values. The value range of possible similarity judgments is utilized better than by the cosine similarity. We further observe more distinct structures in the similarity matrix of the L2 measure. Since the L2 measure is more discriminative and also outperforms the other similarity measures (apart from the cosine similarity), we select the L2 distance as similarity measure for BFCC.

From the evaluation of the similarity measures we learn that the choice of the similarity measure significantly influences the performance of the overall system. The choice of an adequate similarity measure for a given feature is a crucial factor in the design of the scene detector. We further observe that the f_1 scores for the single features do not exceed 0.6 significantly. In the following systematic evaluation we evaluate the



(a) L2 measure (b) Cosine similarity

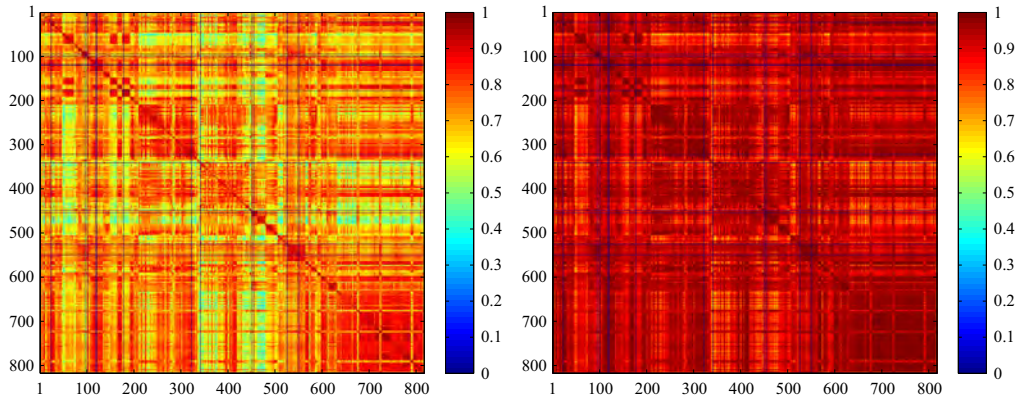
Figure 7.8: Performance figures for L2 measure and Cosine similarity for BFCC in "Three Songs of Lenin". The x-axis represents the values of the comparison threshold (from 0.5 to 1), the y-axis the obtained f_1 scores. The different curves represent experiments with different window sizes w (from 3 to 60).

influence of feature combination, multiple modalities, and refinement on the overall performance.

7.5.2 Features and Multimodality

We systematically evaluate all possible feature combinations which includes also the single features and all possible combinations of modalities. As described in Section 7.4.1 we select only the set of the best 5% of the evaluation results for further analysis. In this quasi-optimal result set we analyze the occurrence frequency of the different features and feature combinations. Each of the three groups of films (archive silent films, archive sound films, and contemporary films) is evaluated separately.

For the group of silent archive films the distribution of feature combinations is shown in Figure 7.10(a). There are 7 possible combinations to arrange the three features BBH, EH, and SIFT. Among the 5% best results most of the system configurations (64%) employ all (visual) features together. 32% of the system configurations (bars 2-4) use two features and only 4% of the results are obtained by a single feature. We draw two basic conclusions from this result. *First*, by employing several features we increase the probability to obtain a good result. This follows from the observation that the large number of system configurations which employ all features, *all* differ in the remaining



(a) L2 measure

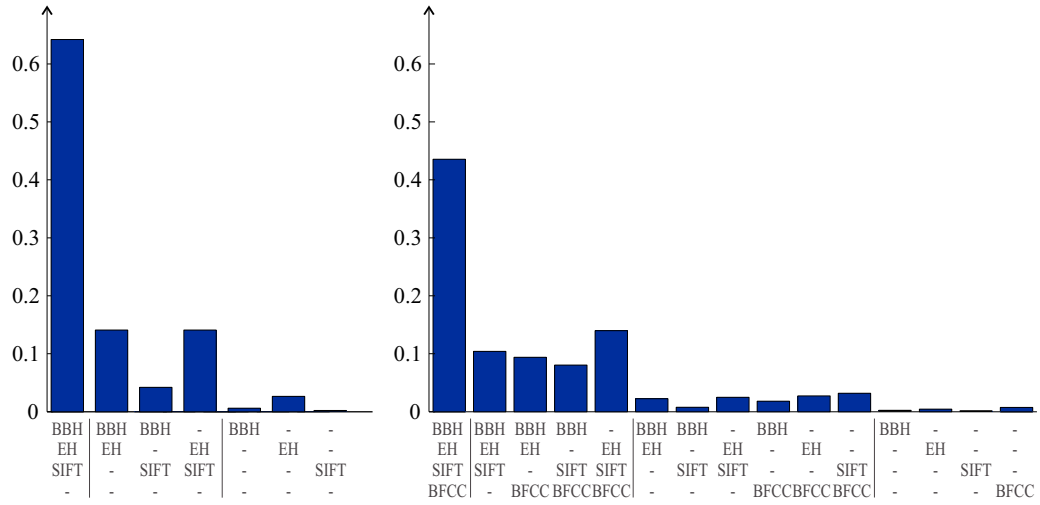
(b) Cosine similarity

Figure 7.9: Similarity matrices for BFCC with L2 measure and Cosine similarity for the film "Three Songs of Lenin". Each pixel represents the pairwise similarity between two shots of the film.

parameter values. This means that for many different parameter values near-optimal results can be obtained if all features are used in combination. The system becomes less dependent on the remaining parameters such as window size and similarity thresholds when all features are used. *Second*, we observe that system configurations that employ single features are also able to obtain near-optimal results. However, the number of different system configurations which obtain near-optimal results with a single feature is small (4%). High performance with a single feature is obtained only for carefully chosen parameter values. This shows that such system configurations are highly dependent on the underlying data (overfitting) and do not generalize well for several films.

The set of quasi-optimal results for archive silent films contains all possible feature combinations which can be seen in Figure 7.10(a). We further observe that all feature combinations in the set achieve similar f_1 scores. Consequently, there is no clear winner among the different features and feature combinations. Each feature and each feature combination is able to obtain near-optimal results, however with different generalization abilities.

From the distribution of *single* features (BBH, EH, and SIFT) we observe that there is only one system configuration in the quasi-optimal set that employs solely the SIFT feature. Five configurations exist that employ the BBH feature and 22 configurations



(a) archive silent films

(b) archive sound films

Figure 7.10: The distribution of all possible feature combinations in the quasi-optimal result set for the groups of archive silent films and archive sound films. Each bar corresponds to one feature combination. The y-axis represents the portion of system configurations in the quasi-optimal result set that employs the corresponding feature combination. All bars together sum up to one.

are based only on EH. These results indicate that the more sophisticated SIFT feature is less effective for scene segmentation than EH and BBH.

For the group of archive sound films we observe a similar distribution of feature combinations in the quasi-optimal result set as for the archive silent films, see Figure 7.10(b). For archive sound films there are 15 possible feature combinations due to the additional auditory feature BFCC. Similarly to the silent films, most system configurations employ all features together (44%). The system configurations with three features make up 41% of the configurations. 12% of the system configurations employ only two features and only 3% of the configurations rely on a single feature only. We draw similar conclusions for the sound films as for the silent films. Again, using more features increases the independence from the other parameters in the system. The configurations with single features are again highly dependent on the parameters and thus overfit on the data.

The results for single features are also similar to that of the silent films. Among the visual features again the EH most frequently appears as the only feature among the near-optimal solutions followed by BBH. SIFT is employed in only 3 system con-

figurations as the only feature. We further observe that the performance obtained by SIFT is slightly lower than that of the other features. Surprisingly, the application of the auditory feature BFCC alone outperforms all of the visual features in the number of system configurations as well as in retrieval performance.

We observe a relatively wide range of f_1 scores in the quasi-optimal result set for the archive sound films that ranges from 0.44 to 0.62. Consequently, we cannot consider the performance of all system configurations to be equivalent as stated in Section 7.4.1 for this group of films. The reason for the wide range of f_1 scores is the following: System configurations that incorporate only visual features are not able to exceed an f_1 score of 0.52. By incorporating the auditory modality a performance gain of up to 10% is achieved. *Both* modalities are necessary to obtain an optimal result. We conclude that both modalities complement each other and that multimodal processing is beneficial in the context of the archive sound films.

Additionally to the archive film material, we evaluate the scene segmentation framework for the group of contemporary films. We first investigate the distribution of all possible feature combinations in the quasi-optimal result set, see Figure 7.11. The distribution of feature combinations is similar to that of archive sound films, see Figure 7.10(b). 50% of the system configurations in the quasi-optimal set of results contain all features and thereby also both modalities. 41% of all system configurations employ three features. The portion of system configurations with one or two features together is only 9%. We conclude that the behavior of the framework for different feature combinations is similar for contemporary material and for archive film material. System configurations with a larger number of features are more robust against variations of the remaining parameters and thus do better generalize the data.

We further investigate the performance of the single modalities for contemporary material. In the set of quasi-optimal results there is no system configuration that solely relies on the auditory modality. Consequently, the auditory modality by itself yields only suboptimal results. The system configurations that are based on the visual modality only yield f_1 scores between 0.51 and 0.53 for single features and scores between 0.54 and 0.55 for combinations of visual features. Higher scores are only obtained in combination with the auditory modality. Multimodal system configurations yield f_1 scores from 0.58 to 0.61. By combining both modality a performance gain of up

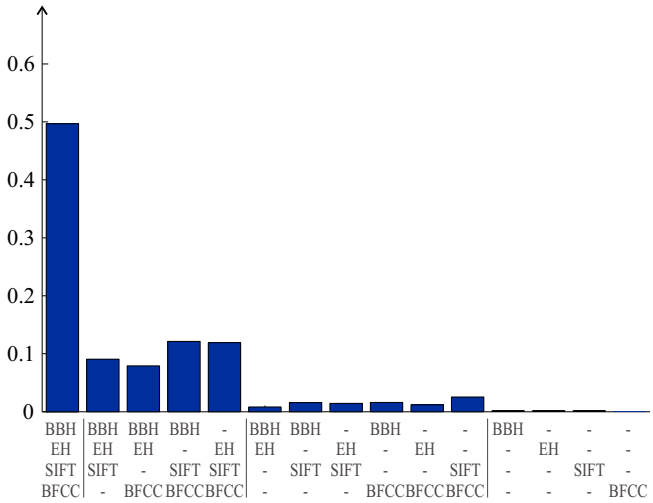


Figure 7.11: The distribution of all possible feature combinations in the quasi-optimal result set for the group of contemporary films. Each bar corresponds to one feature combination. The y-axis represents the portion of system configurations in the quasi-optimal result set that employs the corresponding feature combination.

to 6% is achieved which shows that multimodal processing is beneficial for contemporary material.

In conclusion, the systematic evaluation of features and feature combinations shows that large feature combinations are more likely to produce high performance than single features. Additionally for both, archive and contemporary films, we observe that multimodal processing is beneficial for scene segmentation. There is no difference in the optimal feature selections for archive and contemporary material. In both cases all features together yield the best scores. We conclude that features that are already robust to archive film material also perform well on contemporary material.

7.5.3 Temporal Window Length and Comparison Thresholds

The quasi-optimal result set is the basis for the analysis of the distributions of the temporal window length and the similarity comparison thresholds. Table 7.4 summarizes the median values and the ranges for the thresholds and the window length for all investigated films. We observe that all possible values of the temporal window length w produce results in the quasi-optimal result set. We conclude that w is no critical parameter for the performance of scene segmentation. Similarly to the temporal window length, all possible values of the comparison parameter t^V_{SIFT} produce results in the

Film	w		t^V_{SIFT}		t^V_{BBH}		t^V_{EH}		t^A_{BFCC}	
	med.	range	med.	range	med.	range	med.	range	med.	range
11th	5	5-50	15	5-25	.71	.57-.85	.82	.66-.98		
MMCV	10	5-50	10	5-25	.78	.64-.85	.87	.61-.98		
EsmR	5	5-50	15	5-25	.71	.57-.85	.82	.55-.98	.92	.81-.98
3SoL	5	5-40	15	5-25	.78	.64-.85	.87	.61-.98	.92	.81-.98
TGun	30	5-50	15	5-25	.71	.57-.85	.87	.77-.98	.92	.87-.98
PulpFn	15	5-50	15	5-25	.71	.64-.85	.87	.71-.98	.92	.92-.98
Lola	10	5-50	15	5-25	.64	.57-.85	.87	.66-.98	.92	.87-.98

Table 7.4: Summary of the distributions of parameter values used in the experiments that lead to results in the quasi-optimal result set. The column *med.* contains the median while the column *range* contains the minimum and the maximum values.

quasi-optimal result set. This is not true for the other similarity comparison thresholds (t^V_{BBH}, t^V_{EH}, t^A_{BFCC}) whose optimal ranges are smaller than their possible ranges. The optimal threshold values for BBH range from 0.57 to 0.85, for EH the optimal range is 0.71 to 0.98 and for BFCC the threshold is in the range 0.87 to 0.98. From Table 7.4 we observe that the variation of the thresholds' medians is low over the films or even constant (in case of BFCC). This indicates that the dependence of the thresholds' values on the analyzed films is limited.

The above information is useful for automated parameter optimization but it does not reveal if there is a relation between the temporal window size w and the similarity comparison thresholds (t^V_{SIFT}, t^V_{BBH}, t^V_{EH}, t^A_{BFCC}). In order to investigate this relation, we set the temporal window size to three values within its possible value range ($w = 10, 20, 30$) and analyze the effects on the similarity comparison thresholds. Tables 7.5, 7.6, and 7.7 summarize the similarity comparison thresholds for fixed w.

We observe a trend regarding the median and the lower bound of the value ranges: The larger the window size w, the higher the values of median and lower bound of the comparison thresholds. Stricter similarity requirements have to be met when a larger window size w is taken into account. The combination of larger windows and more tolerant similarity comparison lead to under-segmentation because too many shots are grouped into a small number of scenes. Correspondingly, short temporal windows lead to over-segmentation if they are combined with a strict similarity comparison because in this case shots actually belonging together are not detected as similar and consequently

Film	w fixed	t^V_{SIFT} med.	t^V_{SIFT} range	t^V_{BBH} med.	t^V_{BBH} range	t^V_{EH} med.	t^V_{EH} range	t^A_{BFCC} med.	t^A_{BFCC} range
11th	10	20	10-25	.64	.64-.85	.82	.71-.98		
MMCV	10	5	5-25	.78	.71-.85	.87	.77-.98		
EsmR	10	15	5-25	.78	.57-.85	.82	.71-.98	.92	.87-.98
3SoL	10	15	5-25	.78	.71-.85	.87	.71-.98	.92	.87-.98
TGun	10	15	5-25	.71	.57-.85	.87	.77-.98	.87	.87-.98
PulpFn	10	15	5-25	.71	.64-.85	.87	.71-.98	.92	.92-.98
Lola	10	15	5-25	.71	.64-.85	.87	.77-.98	.92	.92-.98

Table 7.5: Summary of the distributions of the similarity comparison thresholds with the fixed temporal window size $w = 10$. The column *med.* contains the median while the column *range* contains the minimum and the maximum values.

not grouped into scenes. From the experiments we conclude that (i) there is a strong relation between the temporal window size and the similarity comparison thresholds and (ii) for all temporal window sizes we obtain quasi-optimal results. The window size and the comparison thresholds compensate for each other. This means that even with a fixed window size quasi-optimal results can be achieved if the comparison thresholds are chosen accordingly.

7.5.4 Refinement and Pruning

We further evaluate the performance of the refinement and the pruning step of the scene segmentation framework. For this purpose, we determine the optimal performance figures (f_1 scores) with and without refinement and pruning separately. The performance difference is measured by a value Δf_1 which is positive if performance is influenced positively by refinement or pruning and negative if performance decreases. Additionally, we investigate how many system configurations in the quasi-optimal result set employ refinement and pruning. From the resulting distribution we compute the median for both postprocessing steps separately. The median is 1 if the majority of system configurations employs refinement or pruning, respectively and otherwise 0.

The results for the three investigated groups of films and the individual films are summarized in Table 7.8. For the three groups of films refinement and pruning always improve segmentation performance. Since the results for the groups of films are av-

Film	w fixed	t^V_{SIFT} med.	t^V_{SIFT} range	t^V_{BBH} med.	t^V_{BBH} range	t^V_{EH} med.	t^V_{EH} range	t^A_{BFCC} med.	t^A_{BFCC} range
11th	20	20	10-25	.85	.78-.85	.82	.77-.98		
MMCV	20	15	10-25	.71	.71-.85	.93	.82-.98		
EsmR	20	15	5-25	.78	.71-.85	.87	.77-.98	.92	.92-.98
3SoL	20	15	5-25	.78	.71-.85	.93	.82-.98	.92	.92-.98
TGun	20	15	5-25	.71	.64-.85	.87	.82-.98	.92	.87-.98
PulpFn	20	15	5-25	.71	.64-.85	.87	.77-.98	.92	.92-.98
Lola	20	15	5-25	.71	.64-.85	.93	.82-.98	.92	.92-.98

Table 7.6: Summary of the distributions of the similarity comparison thresholds with the fixed temporal window size $w = 20$. Compared to Table 7.5 the parameter's value ranges tend to become smaller. The column *med.* contains the median while the column *range* contains the minimum and the maximum values.

eraged over the respective films we conclude that refinement and pruning in average improve scene segmentation. For the individual films there are a few cases where the performance is slightly decreased, e.g. for "Enthusiasm" and "Man with a Movie Camera (V)" refinement decreases results by 1%. However, in comparison to the potential performance gain of up to 7% the decrease of performance is negligible. The same applies to the pruning step. In most cases pruning improves results (in the best case by 7%). In the worst case performance is decreased by 2%.

Table 7.8 further shows the medians for refinement and pruning in the set of quasi-optimal results. In most cases the median is 1 which means that the respective processing step is employed in the majority of system configurations. From these distributions and the performance figures we conclude that both, pruning and refinement are beneficial for scene segmentation.

7.5.5 Retrieval Results

Finally, we present the best retrieval results obtained in the systematic evaluation for the individual films. Table 7.9 shows the best result in terms of f_1 score together with the corresponding recall and precision for each film. In scene segmentation the importance of recall and precision is usually not weighted equally. Generally, a slight over-segmentation that returns some false scene boundaries (false positives) is favored over an under-segmentation where actual scene boundaries are missed. In other words,

Film	w fixed	t^V_{SIFT} med.	t^V_{SIFT} range	t^V_{BBH} med.	t^V_{BBH} range	t^V_{EH} med.	t^V_{EH} range	t^A_{BFCC} med.	t^A_{BFCC} range
11th	30	20	10-25	.85	.78-.85	.87	.82-.98		
MMCV	30	15	10-25	.78	.78-.85	.87	.82-.98		
EsmR	30	15	5-25	.78	.71-.85	.93	.77-.98	.98	.92-.98
3SoL	30	15	5-25	.85	.71-.85	.93	.82-.98	.98	.87-.98
TGun	30	15	5-25	.71	.64-.85	.87	.82-.98	.92	.92-.98
PulpFn	30	15	5-25	.71	.64-.85	.87	.77-.98	.92	.92-.98
Lola	30	15	5-25	.71	.64-.85	.93	.87-.98	.92	.92-.98

Table 7.7: Summary of the distributions of the similarity comparison thresholds with the fixed temporal window size $w = 30$. Compared to Table 7.5 and Table 7.6 the parameter's value ranges tend to become smaller for higher w. The column *med.* contains the median while the column *range* contains the minimum and the maximum values.

this means that recall is favored over precision. We provide retrieval results with optimized recall in columns 5-7 in Table 7.9. For the sake of completeness, we also provide results with optimized precision in columns 8-10. For two films recall always exceeds precision and consequently no optimized result for precision can be provided.

From the distribution of recall and precision in the quasi-optimal result sets we observe that the proposed scene segmentation framework in most cases generates results where recall exceeds precision. This means that the framework tends to produce slight over-segmentations. As mentioned above this behavior is generally more desirable than the generation of under-segmentations. We observe from Table 7.9 that the performance of scene segmentation is higher for the contemporary material than for archive film material. The f_1 scores for contemporary films range from 0.67 to 0.72 while for archive material the results are generally lower. There are two reasons for the performance differences. First, the quality of the archive film material impedes similarity comparisons and second the scenes in contemporary material are much more coherent and have a simpler structure than the scenes in the archive film material.

For the archive films we observe that scene segmentation for sound films performs better (f_1 scores of 0.65 and 0.66) than for silent films (f_1 scores of 0.45 and 0.58). If we skip the auditory modality for the segmentation of archive sound films we observe a significant decrease in performance. For "Three Songs of Lenin" (3SoL) performance decreases by 8% ($f_1 = 0.57$) and for "Enthusiasm" (EsmR) by 6% ($f_1 = 0.6$). By

Film/Group	Refinement		Pruning	
	median	Δf_1	median	Δf_1
Group of archive silent films	1	+1%	1	+2%
Group of archive sound films	1	+1%	0	+5%
Group of contemporary films	1	+2%	1	+2%
Three Songs of Lenin	1	+2%	0	+7%
Enthusiasm (Restored)	1	-1%	1	+2%
The Eleventh Year	0	0%	1	0
Man with a Movie Camera (V)	0	-1%	1	+1%
Run Lola Run	1	+1%	0	-2%
Pulp Fiction	1	+1%	1	+2%
Top Gun	1	+7%	1	+5%

Table 7.8: Results for the refinement and pruning steps for the groups of films and the individual films. Δf_1 indicates the performance gain and the median shows whether most system configurations employ refinement or pruning (value 1) or not (value 0).

skipping the auditory modality, the results become comparable to those of the archive silent films. From these experiments and from the observations made in in Section 7.5.2 we conclude that the auditory modality facilitates scene segmentation.

7.6 Summary

In this chapter, we have presented a framework for scene segmentation that solely relies on audio and visual similarities. This property makes the framework well-suited for scene segmentation of archive film material where we cannot exploit a priori knowledge such as compositional rules from filmmaking. The framework is based on the detection of similar audio and visual content in a film by different (preferably orthogonal) features. The search range for similar content is thereby restricted to a temporal window in order to avoid under-segmentation. Based on detected similarities shots are grouped together. The shot groupings generated for each feature are fused together and result in a segmentation of the film into core scenes. Finally, a refinement step closes the gaps between the core scenes and a pruning step removes short scenes.

We evaluate all combinations of the parameters of the framework in a systematic evaluation and investigate the dependencies between parameters. First, we evaluate

Film	Best f_1 score			Recall favored			Precision favored		
	Rec.	Prec.	f_1	Rec.	Prec.	f_1	Rec.	Prec.	f_1
3SoL	0.60	0.71	**0.65**	**0.78**	0.50	0.61	-	**-**	-
EsmR	0.80	0.56	**0.66**	**0.87**	0.51	0.64	0.60	**0.64**	0.62
11th	0.48	0.43	**0.45**	**0.64**	0.34	0.45	0.44	**0.46**	0.45
MMCV	0.62	0.54	**0.58**	**0.82**	0.42	0.56	-	**-**	-
Run Lola Run	0.73	0.66	**0.70**	**0.75**	0.64	0.69	0.63	**0.73**	0.67
Pulp Fiction	0.71	0.72	**0.72**	**0.77**	0.66	0.71	0.62	**0.76**	0.69
Top Gun	0.65	0.69	**0.67**	**0.71**	0.61	0.65	0.54	**0.72**	0.62

Table 7.9: Performance figures for all investigated films. For each film we provide the maximum performance in terms of f_1 score and a result with optimized recall and (if possible) one with optimized precision.

similarity and distance measures and identify an adequate measure for each feature. Next, we evaluate different feature combinations and benefit of multimodal processing. Additionally, we investigate the behavior of the framework for different parameter combinations of temporal window size w and the similarity comparison thresholds and investigate their dependencies. Finally, we evaluate the benefit of the refinement and pruning steps.

The systematic evaluation provides valuable insights about scene segmentation of archive and contemporary film material. The most important findings from the study are:

- The choice of the similarity measure is crucial for a feature. Different measures have to be evaluated to identify an adequate one.
- Generally, the chance to obtain a good result increases with the number of (complementary) features.
- Multimodal processing significantly increases the performance. The individual modalities achieve only suboptimal results.
- Some thresholds compensate for each other, such as the window size and the similarity thresholds. Consequently, one threshold (the window size) can be fixed in the optimization without losing retrieval quality.

- Refinement and pruning usually improve the scene segmentation
- The framework principally behaves similar for archive and contemporary film material. If robust features are employed, they are applicable to both types of films. However, scene segmentation is generally harder for documentary archive film material due to the low cohesion of the scenes.

The proposed framework is extensible in several ways. First, the framework can easily be extended to incorporate information from several keyframes per shot in order to enable more comprehensive similarity comparisons. Second, additional (and diverse types of) features may be incorporated, such as face recognizers to group shots with recurring faces and object detectors to group shots which show similar objects. Finally, alternative fusion schemes can easily be integrated. For a large number of features for example a fusion scheme based on majority voting introduces additional robustness because it enforces that every decision is supported at least by a majority of features.

Chapter 8

Extraction of Synchronous Montage Sequences

In the last chapter we have presented a method for the segmentation of a film into semantically coherent units (scenes). A related type of semantically coherent units in film are *synchronous montage sequences*. Synchronous montage is a higher-level concept in filmmaking that is related to the synchronous editing of visual and auditory content. Synchronous montage sequences are usually parts of a scene which highlight important events. We first introduce the concept of synchronous montage in Section 8.1. Next, we review work from related research fields in Section 8.2. In Section 8.3 we present a cross-modal approach that extracts sequences from a film with synchronous audio-visual montage. Experiments (Sections 8.4 and 8.5) show that the approach robustly extracts synchronous montage sequences. Furthermore, we observe that the extracted sequences have high semantic relevance for the investigated films.

8.1 Introduction

In synchronous montage the editor purposely synchronizes the soundtrack with the visual montage (the cutting) of a film. Synchronous audio-visual montage enables the filmmaker to accentuate important events and actions and to increase tension and tempo in a scene [27]. Such sequences contain rich semantic context which is important for understanding a film.

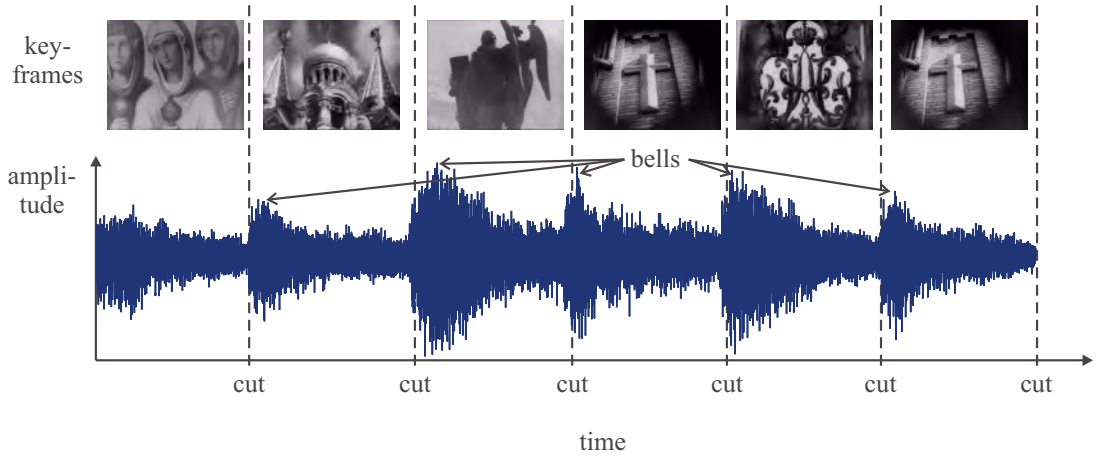


Figure 8.1: A synchronous montage sequence. The keyframes of each shot show different religious symbols. The peaks in the waveform's amplitude at the shot cuts correspond to the church bells.

Synchronous montage has been employed since the early years of sound film and is still employed in contemporary films (e.g. in action scenes and dialogue sequences). A famous example for the synchronous montage technique in film history stems from the film "Enthusiasm" by Dziga Vertov from 1931. "Enthusiasm" is a propagandistic documentary about the first Soviet five-year plan. A central sequence in the film shows several consecutive static shots of different religious and monarchal symbols (e.g. a tsarist monogram, statues of Christ, crucifixes). At each shot cut between two different symbols the director positioned the sound of a church bell in the soundtrack. The synchronous church bells increase the perceptual salience of the sequence and create a threatening and warning atmosphere. According to the film literature, this is a key scene in the film that expresses the rejection of religion and the tsarist regime by the communist regime [63]. An excerpt of the sequence together with the waveform of its soundtrack is shown in Figure 8.1.

Synchronous montage is still a popular technique in contemporary films to emphasize important events (a detailed discussion is provided in [27]). For example, in the feature film "The Hunt for Red October" from 1990 the director exploits synchronous montage in dialogue sequences to emphasize the speakers and the speech. Another example mentioned in [27] is the end scene (the showdown) of "The Last of the Mohicans" where the cutting is coordinated with the musical rhythm.

Due to their rich semantics, synchronous montage sequences are important for (automated) film annotation, interpretation, and summarization. For example, synchronous montage sequences are likely to contain key scenes which should be part of an automated generated summary or trailer. Additionally, film scientists are interested in the extraction of such sequences for film and montage analysis.

The automated extraction of synchronous montage sequences has not been addressed so far. In this chapter, we present a method for the automated extraction of sequences with synchronous audio-visual montage. We develop a cross-modal approach that extracts such sequences by detecting temporally correlating audio and visual events. Unfortunately, the temporal correlation[1] of auditory and visual information on the signal-level differs significantly from the correlation on the perceptual level. Consequently, established methods for the estimation of temporal correlations do not work properly. We propose an approach for the extraction of temporal correlations that are more meaningful and intuitive for the human observer. First, we extract salient audio and visual events by the detection of *onsets*. In general, onsets represent abrupt changes in the underlying signals. Visual onsets refer to abrupt shot boundaries (shot cuts). In the audio domain, onsets are for example musical beats, sudden sound effects, and points in time when an actor starts to speak after a pause. Next, we detect temporally correlated audio and visual onsets by analyzing their coincidences and their temporal neighborhoods. Finally, we extract entire sequences that contain several subsequent correlated audio and visual onsets (synchronous montage sequences). Experiments with different films show that the approach is able to retrieve relevant montage sequences. The results include key scenes with rich semantics.

8.2 Related Work

The audio-visual synchronicity in film montage is a semantically relevant compositional principle that has to our knowledge not been analyzed automatically so far. However, audio-visual synchronicity (correlation) has been studied by researchers in different related domains such as sound source localization [18, 109, 132, 149], talking face detection [92, 196], speech recognition [155], person authentication [31], and

[1]Note that "correlation" in this chapter is not meant in a strict statistical sense. In the context of this chapter it refers to the temporal proximity of auditory and visual events.

surveillance [15]. The computation of temporal audio-visual correlation is performed at different *levels*. Most approaches compute correlation directly between audio and visual features (*feature-level*). Frequently employed correlation measures are Pearson correlation [155] and mutual information [92]. Some methods first reduce dimensionality (e.g. by Canonical Correlation Analysis, CCA and Latent Semantic Indexing, LSI) and then perform correlation computation in a lower-dimensional space [196].

On the feature-level we are able to capture the *natural* correlation that exists between audio and visual signals (from the same source), e.g. speech and the speaker's lip movements. Consequently, it is well-suited for talking face detection and person identification. However, at this rather low level it is difficult to integrate delays, tolerances, and irregularities into the correlation computation. This limits the applicability of such methods for the analysis of film montage since delays and irregularities are sometimes introduced by the filmmaker for stylistic reasons. Additionally, methods that rely on LSI and CCA require a certain amount of training to learn the joint distributions of audio and visual features. Training as in talking head detection is hardly feasible in the domain of film analysis because of the wide range of objects and events and the different types of editing styles.

Other methods (especially from the surveillance domain) compute temporal correlations on the basis of classified high-level decisions (*decision-level*) [14]. Methods at this level learn frequent audio and visual events (atomic events) autonomously and recognize higher-level events (e.g. running, opening a door) by merging co-occurring atomic event classifications [15]. Such methods are usually designed to operate in controlled environments (e.g. a corridor in a building) and require recurring events for learning. Both, recurring events and controlled environments are not available in feature films.

To sum up, methods on the feature-level require a strong and direct correlation in the audio and visual feature vectors and methods on the decision-level require highly controlled environments. For the analysis of audio-visual montage, a method is required that (i) enables flexible temporal correlation assessments and (ii) operates on an uncontrolled (general purpose) set of events. For that reason, we perform the temporal correlation analysis on an intermediate level: the *landmark-level*.

On the landmark-level we operate on salient points (automatically detected peaks and onsets) in the audio and visual feature vectors [163]. This level facilitates the representation of general purpose events and a flexible temporal correlation estimation.

Additionally, psychophysical research points out that the human synchrony perception relies on the matching of salient features (peaks and troughs) in the audio and visual modalities [70].

Only little work on audio-visual correlation estimation on the landmark-level exists. Barzelay and Schechner perform sound localization by correlating audio and visual onsets [18]. The audio onsets are derived from a spectrogram and the visual onsets are extracted from motion trajectories by detecting peaks in the trajectories' curvature. Temporal coincidences of onsets are detected by a likelihood function that yields high values where audio and visual onsets temporally coincide. Similarly, Monaci and Vandergheynst (and later Casanovas et al. [132]) perform general purpose sound localization by correlating onsets in audio and visual feature vectors [149]. From the audio and visual onsets the authors first compute two binary vectors where spikes indicate onset positions. Next, they broaden the spikes with a rectangle function to increase the temporal tolerance. Finally, they combine both vectors by a logical AND to obtain temporally correlated audio-visual onsets. The method is for example applied to talking face detection.

The approaches above are not applicable to the analysis of audio-visual film montage. First, the approaches are designed for correlating sound with *motion*. For the analysis of synchronous montage visual onsets originate from shot cuts and not from motion which is structurally different. Second, the approaches above consider consecutive onsets as independent from each other and neglect their neighborhood relationships. Thereby, information on the temporal distribution of the onsets is lost which is important to evaluate the salience of an onset. Third, both approaches neglect the actual strengths of the onsets (their degree of abruptness). In fact, the strength is a further indicator for the salience and is important to obtain estimates in accordance with human assessment.

8.3 Analysis of Synchronous Montage

An overview of our approach is depicted in Figure 8.2. We first perform onset detection in the visual and audio domains separately. Onsets in the visual signal represent abrupt shot boundaries (shot cuts). Audio onsets correspond to musical beats, sound effects (e.g. explosions, cries, sirens), and speech onsets. Shot cut detection is performed by

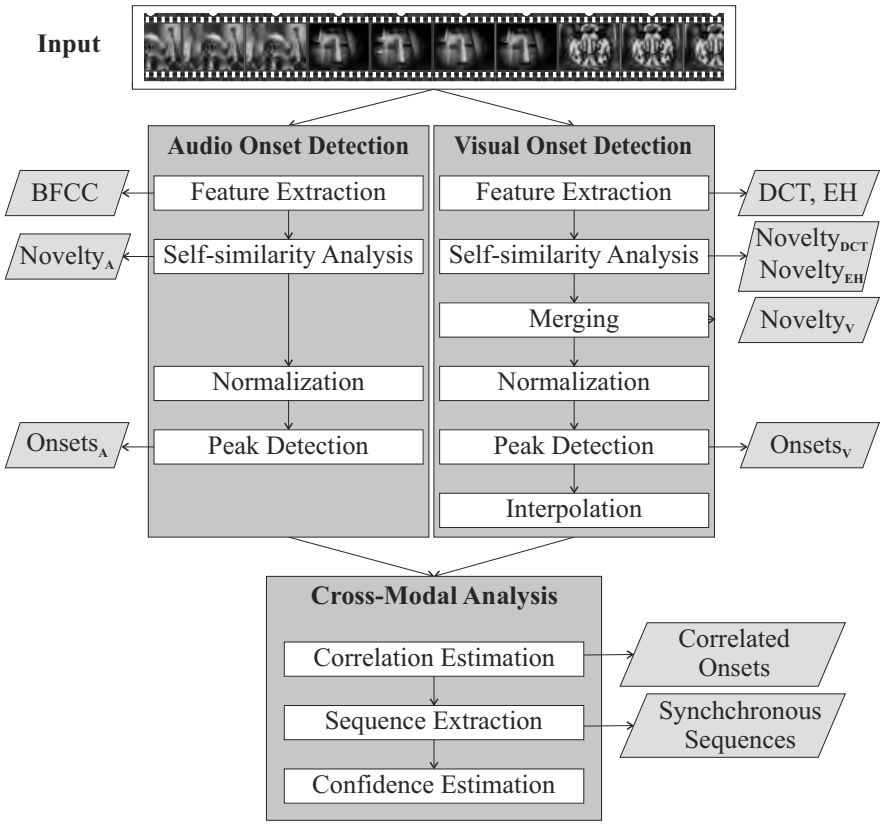


Figure 8.2: Overview of the approach.

the method described in Chapter 5. We use the same analysis framework based on self-similarity analysis for the detection of audio onsets. The result of onset detection are two time series containing auditory and visual onsets. Next, we detect temporally correlating (synchronous) audio and visual onsets (for instance a sudden cry that occurs simultaneously with a shot cut) by a specifically designed weighting function. Finally, we extract entire sequences that contain several consecutive shot cuts with correlated audio onsets with a tolerant segmentation scheme.

8.3.1 Visual Onset Detection

Shots are the most important building blocks of visual film montage. We detect shot cuts (visual onsets) as described in Chapter 5. First, we extract features for each frame (edge histogram, DCT coefficients). Next, we subtract the mean of each feature component (zero-mean features) and perform a temporal self-similarity analysis for

both features and merge the resulting outcomes as described in 5.3.4. The result is a function (novelty curve) which has peaks at positions where the underlying signal changes abruptly. In the case of the visual signal, peaks in the novelty curve indicate shot cuts. The shot cut positions are extracted by a peak detector as described in Section 5.3.3 (after normalization of the novelty curve) and form the set of visual onsets for subsequent processing. Since the frame rate for the visual signal (1/25 seconds) is lower than the frame rate used for audio in the following, the visual onset positions are interpolated (super-sampled) to make them directly comparable with the audio onset positions.

We neglect gradual transitions (e.g. dissolves and fades) since they do not represent distinct events in time that audio onsets can be correlated with. Consequently, they play a secondary role for the detection of sequences with synchronous audio-visual montage.

8.3.2 Audio Onset Detection

For audio onset detection we utilize the same analysis framework as for visual onset detection (based on another feature). For audio analysis, we extract 24 Bark-frequency cepstral coefficients (BFCCs) from audio frames of 30 ms (20 ms overlap). BFCCs employ a psychoacoustically scaled filter bank and compactly represent the coarse spectral frequency distribution in an audio frame (see Section 2.2.6). The BFCCs are first normalized by subtraction of the mean of each component and are then input to a self-similarity analysis like the visual features in Section 8.3.1. The result is again a novelty curve. In the case of audio, peaks indicate abrupt changes (discontinuities) in the audio stream. Such discontinuities occur for example at the beginning of musical beats, speech, and special effects. The stronger a discontinuity the higher is the amplitude of the peaks.

We normalize the novelty curve and extract salient peaks with an adaptive peak detector that uses the median of the novelty curve as threshold. A peak is detected as a salient peak if the amplitude of the novelty curve decreases by more than the median after a local maximum.

Since the novelty curve often contains dense series of peaks (which correspond to the same audio event), we perform a local peak pruning after peak detection. The pruning process moves a sliding window over the signal and compares the heights of

neighboring peaks in the window. Peaks whose heights are smaller than the mean height of all peaks in the window are removed. The pruning step reduces the density of the peaks and leads to a more robust audio onset detection. The positions and heights of the remaining peaks form the set of audio onsets for subsequent processing.

8.3.3 Temporal Audio-Visual Correlation Estimation

The goal of the next step is to find temporally correlated (synchronous) audio and visual onsets which would also be perceived synchronous by a human observer. In general onsets are perceived correlated if they are temporally near to each other. This conforms with the assumptions made in [18], [149], and [70]. In our case, the correlation of onsets means that an audio onset occurs simultaneously with a shot cut. However, we observe that this assumption is not sufficient for the detection of synchronous montage in feature films for two reasons. First, stronger (more salient) onsets catch the viewers attention more than weak onsets. Consequently, we integrate a *salience condition* into the correlation computation that favors stronger onsets (originating from higher peaks). Second, the temporal distribution and the number of audio onsets around a shot cut influence synchrony perception: if many onsets surround a shot cut, they distract the attention of the observer from detecting synchronicity. Consequently, a single isolated audio onset at a shot cut leads to a stronger synchronicity than several audio onsets surrounding a shot cut. Due to the large number of onsets introduced by film music and concurrent background sounds in films, the likelihood is generally high that an audio onset occurs simultaneously with a shot cut accidentally. To take this effect into account, we integrate an *isolation condition* into the correlation computation that favors isolated audio onsets over numerous surrounding onsets.

For temporal correlation estimation we design a weighting function that takes the salience and the isolation condition into account. The weighting function (see Figure 8.3(a)) is centered around a shot cut. The amplitude represents the time-dependent influence of an audio onset for temporal correlation estimation. The function can be partitioned into two different areas.

In area “A” centered around the shot cut the function is positive. Audio onsets that fall within this area influence correlation positively (the nearer the audio onset is to the shot cut the higher is its influence). The weighting function in area “A” models a simple principle of human synchrony perception: Events that are temporally

near to each other are perceived as correlated. With increasing distance the perceived correlation decreases.

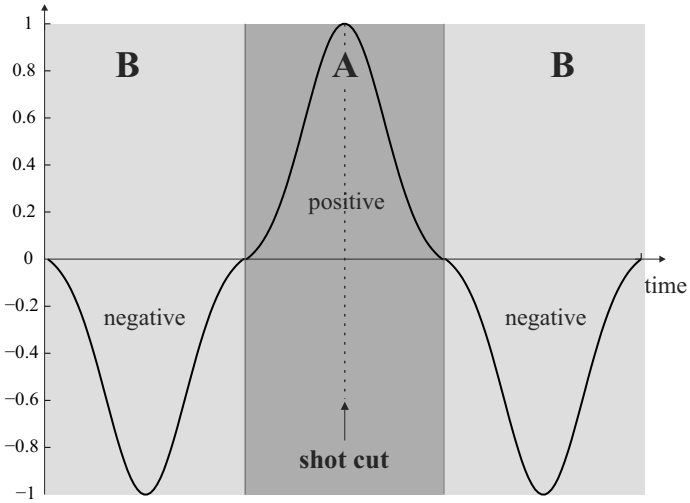
In area "B" the function is negative. Onsets that fall into this area get negative weights. If numerous audio onsets (e.g. originating from different background sounds) surround a shot cut, they contribute negatively to the correlation estimate. An example is shown in Figure 8.3(b). The dashed vertical line marks the shot cut. The spiky curve is the audio novelty and the asterisks mark detected onsets. Even though the central onset (marked with an arrow) is close to the shot cut, the overall correlation at this shot cut is low because the four surrounding onsets have negative weights. This behavior models the isolation condition: the surrounding onsets distract the observer from the central onset which reduces the degree of perceived synchronicity. Figures 8.3(b) and 8.3(c) illustrate the effect of the isolation condition. The shot cut with the isolated onset in Figure 8.3(c) yields a higher correlation c_j than the shot cut with the surrounded onsets.

The correlation computation is performed as follows. Given a set of audio onsets with positions p_i and heights h_i, $i = 1, ..., M$ and a set of visual onset positions (shot cuts) b_j, $j = 1, ..., B$, we center the weighting function w around a shot cut b_j. Note that the weighting function is zero outside of the negative area "B". The correlation c_j at shot cut b_j is the sum of the products of the weighting function w (at position p_i) with the corresponding heights h_i of the audio onsets:

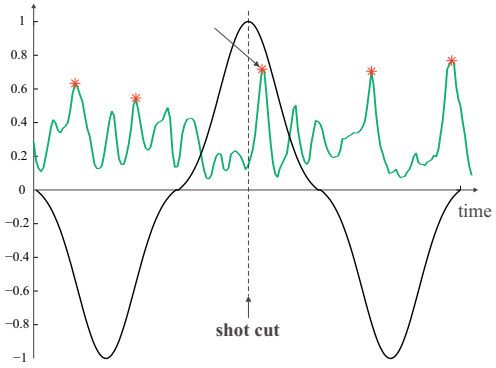
$$c_j = \sum_{i=1}^{M} w(p_i) * h_i. \tag{8.1}$$

By taking the actual onset heights h_i into account we are able to model the salience condition. Higher onsets are more distinctive and influence correlation more than lower onsets. As a result, the correlation measure that takes the isolation condition and the salience condition into consideration.

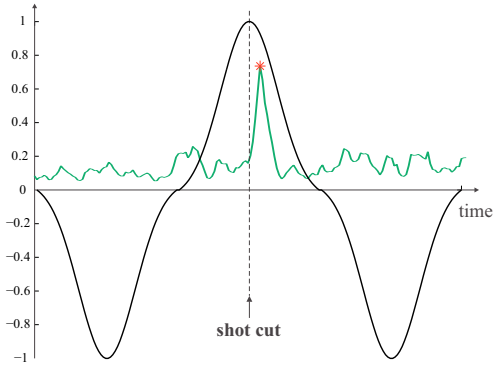
The result of correlation computation is a correlation estimate for each shot cut. In the following, we consider shot cuts with correlation $c_j > 0$ as *synchronously edited shot cuts* and shot cuts with $c_j \leq 0$ as *asynchronously edited*. As a consequence, the example in Figure 8.3(c) is a synchronously edited shot cut and the example in Figure 8.3(b) an asynchronously edited one (see the corresponding correlations c_j).



(a) the weighting function (with temporal partition into areas "A" and "B")



(b) surrounded onsets: $c_j = -0.1$



(c) isolated onset: $c_j = 0.6$

Figure 8.3: The weighting function and examples of positive and negative correlation: the isolated onset yields a higher correlation c_j than a series of onsets that surrounds a shot cut.

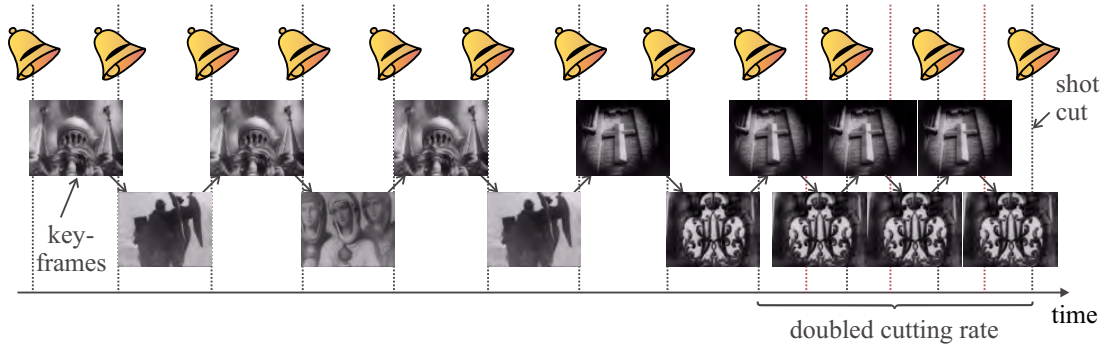


Figure 8.4: The synchronous montage sequence from Section 8.1. The keyframes of each shot show different religious symbols. At the end of the sequence the cutting rate doubles but the frequency of the church bells remains the same. This leads to irregularities (shot cuts without corresponding bell sound, highlighted in red) in the audio-visual correlation.

8.3.4 Extraction of Synchronous Montage Sequences

So far we have analyzed (and classified) whether or not single shot cuts as temporally correlated with significant and isolated audio onsets. In the synchronous montage technique the director makes *repeated* use of synchronously edited shot cuts to attract the attention of the viewer over larger time spans. Consequently, we are interested in the extraction of entire sequences that contain several subsequent synchronously edited shot cuts.

In practice however, such a sequence might contain also some shot cuts that are purposely *not* synchronized with the audio track by the filmmaker (e.g. for stylistic reasons). An example from film history is the sequence of religious symbols from Section 8.1. In the second half of the sequence the cutting rate doubles while the frequency of the church bells remains the same. The result is that only every other shot cut is accompanied by a church bell. The doubling of the cutting rate (halving of the shot lengths) further increases the tension in the scene towards the end. The corresponding part of the sequence is illustrated in Figure 8.4.

Due to such irregularities, for automated sequence extraction on a technical level, we have to search for *possibly interrupted temporal groupings* of synchronously edited shot cuts. For this purpose, we propose a tolerant segmentation scheme consisting of two stages. In the first stage, we search for *neighborhood regions* at each synchronously edited shot cut. The size of the neighborhood regions is maximized on the condition

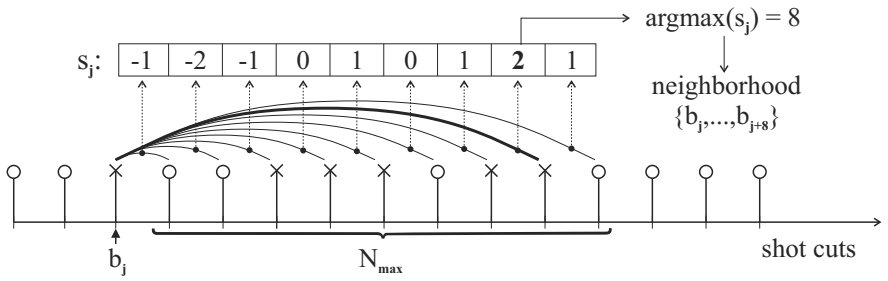


Figure 8.5: The schema for the extraction of a neighborhood region at a shot cut b_j. The maximum sum s_j is obtained for a neighborhood of 8 shots.

that the number of *irregularities* in the neighborhood (asynchronously edited shot cuts) is minimized. In the second stage, we merge the (overlapping) neighborhoods to obtain the final montage sequences.

The first stage is illustrated in Figure 8.5. The extraction of neighborhood regions takes place at synchronously edited shot cuts (marked with "x" in Figure 8.5) only. Asynchronously edited shot cuts marked with "o" can be skipped. Note that this means that a neighborhood region always starts with a positively correlated shot cut. At a given shot cut b_j we position a support window of size n, where n defines the number of neighboring shot cuts that are taken into account. Next, we count the number of positively correlated shot cuts in the support window and subtract the number of negatively correlated shot cuts. This results in a sum $s_{j,n}$ for the support window of size n at shot cut b_j.

We perform the computation of $s_{j,n}$ for different sizes of the support window with $n = 1, ..., N_{max}$ which results in a series of sums $s_j = s_{j,1}, ..., s_{j,N_{max}}$ for the shot cut under consideration (see s_j in Figure 8.5 for the sums of the example sequence). Next, we identify for which window size n the maximum sum is obtained:

$$n_{max} = \underset{n}{\text{argmax}}(s_j). \tag{8.2}$$

In the example in Figure 8.5 the maximum sum is obtained for $n = 8$ (sum is 2). Finally, the region from shot cut b_j to $b_{j+n_{max}}$ is stored as a new neighborhood region. If the maximum sum $s_{j,n_{max}}$ is smaller than 2 no neighborhood region is generated. The process described above is repeated for all synchronously edited shot cuts. The result is a set of (possibly overlapping) neighborhood regions.

In the second stage, we compute the union of all neighborhood regions in order to obtain the final montage sequences. For each extracted sequence we compute a measure that reflects the confidence in the decision that an extracted sequence actually is a synchronous montage sequence. A straight-forward measure is the number of synchronously edited shot cuts in an extracted sequence. The higher this number the higher the likelihood that the extracted sequence is a synchronous montage sequence.

8.4 Experimental Setup

We evaluate the proposed method with both: contemporary feature films as well as historic (archive) film material from the early years of sound film. Especially, the historic material is well-suited for the evaluation of the proposed method because (i) it has low sound and image quality (noise, distortions) and thus allows for the evaluation of the robustness of the method and (ii) the filmmakers of the early sound films intensively experimented with the usage of sound in film montage and as a result the films frequently contain montage sequences with strong audio-visual correlations.

8.4.1 Data

The historic material includes the film "Enthusiasm" by Dziga Vertov from 1931 and "October: Ten Days That Shook the World" by Sergej Eisenstein from 1927. Each of the two filmmakers developed his own montage rules, which are today subsumed by the term *Soviet/Russian Montage Theory* [100]. Soviet montage theory of that time is characterized by very strict and formalistic rules. These montage rules also affect the audio-visual montage which makes the respective films particularly interesting for our evaluation.

"Enthusiasm" is a documentary about the Soviet first five-year plan for economic development (1928-1932) [215]. Vertov deliberately coupled "visible and audible moments" to create a strong tension between sound and visuals which resulted in a revolutionary style of audio-visual montage for that time of filmmaking [63].

The film "October" from Eisenstein is an (originally silent) fictional film in celebration of the 10th anniversary of the October Revolution. In 1966 a soundtrack containing sound effects and music by Dimitri Shostakovich was added. "October" contains highly formalistic visual montage which partly correlates with the later added soundtrack [59].

The contemporary feature films include "The Hunt for Red October" directed by John McTiernan and "Fight Club" by David Fincher. "The Hunt for Red October" was selected because it is a good example of synchronous montage according to [27]. For "Fight Club" no prior information on the montage style was available. The film was selected to broaden the test set and to reduce the bias introduced by the other selected films.

8.4.2 Ground Truth

There is no ground truth available for the performed investigation because this aspect of film montage has not been analyzed automatically so far. Ground truth generation is a time-consuming process and requires the expertise of domain experts. The consequences for our evaluation are twofold.

First, in absence of ground truth we cannot compute recall and precision for a retrieval experiment. Nevertheless, we are able to evaluate the retrieved sequences manually and compute the precision for result sets of different sizes (e.g. for the 3, 5, and 10 sequences with the highest confidence).

Second, we attempt to generate a ground truth for selected material to enable a more comprehensive evaluation of the retrieval performance. We select "Enthusiasm" which makes the most intensive use of synchronous audio-visual montage. Together with domain experts we annotate synchronously edited shot cuts and the synchronous montage sequences in the film.

8.4.3 Parameters

The proposed method requires only a minimum set of parameters. The onset detection is adaptive and free of parameters. This makes the method applicable to a wide range of film material. The correlation computation requires the specification of two parameters: the width of the weighting function w (see Section 8.3.3) and maximum support window size N_{max} (in unit shot cuts, see Section 8.3.4). We experiment with widths of w ranging from 1 to 1.8 seconds. The wider the function the more temporal tolerance is allowed in the correlation computation. Typical values for N_{max} are between 5 and 11 (shot cuts). The larger the values of N_{max} the more irregularities are tolerated during segmentation.

	System P			System A		
Task	Recall	Precision	f_1	Recall	Precision	f_1
#1 Shot cuts	0.67	0.64	0.65	0.88	0.48	0.62
#2 Sequences	0.85	0.72	0.78	0.97	0.41	0.58

Table 8.1: Performance of the two compared system configurations evaluated against the ground truth.

8.5 Experimental Results

We first evaluate the retrieval performance of the proposed method with the generated ground truth. For comparison, we integrate the correlation computation by Monaci et al. [149] into our method. For this purpose, we broaden the onsets with a rectangular filter (to gain temporal tolerance) and combine the audio onsets and the visual onsets by a logical AND as in [149].

We define two different system configurations: the proposed method with the weighting function as correlation estimator from Section 8.3.3, short: "System P" and as alternative system the proposed method with the correlation estimation of [149], short: "System A". Both systems operate on the same audio and visual onsets.

Table 8.1 presents the results of both systems for the film "Enthusiasm". We compute recall and precision for two different tasks: first, the detection of synchronously edited shot cuts (task #1) and second, the extraction of synchronous montage sequences (task #2) which is based on the first task. First, we compare the systems' performance to the theoretical performance attainable by random guessing. The probability for "Enthusiasm" that a shot cut is synchronously edited is 0.36. The probability of occurrence of a synchronous montage sequence is 0.35. This means that for both tasks random guessing would result in a recall of approximately 0.5 and a precision of approximately 0.36 and 0.35, respectively. From Table 8.1 we observe that both systems significantly outperform random guessing in both tasks.

From Table 8.1 we further observe that System A yields a relatively high recall but a precision which is near random. The reason is that nearly all shot cuts are classified as "synchronously edited" and during sequence extraction large sequences are extracted that cover nearly the entire film. This is best illustrated in Figure 8.6 (lower part) which shows the strong under-segmentation produced by the alternative system.

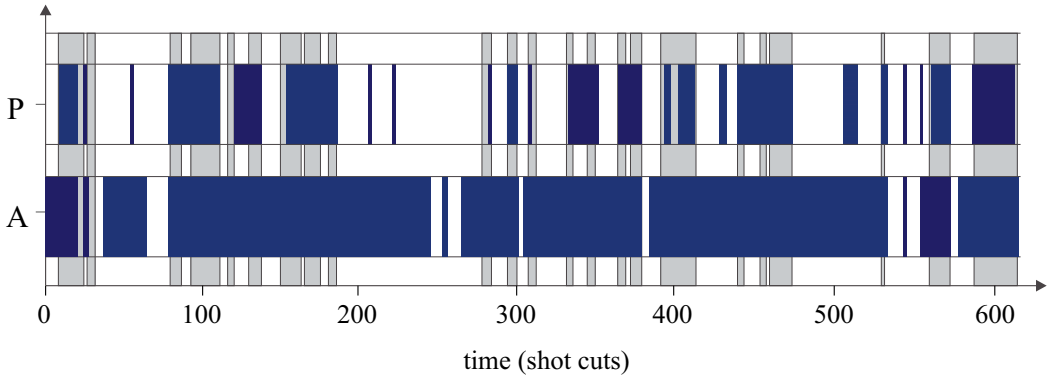


Figure 8.6: Sequence extraction results over time (x axis). The regions in the background (gray) represent the sequences in the ground truth. The regions in the foreground (blue) represent the sequences extracted by the proposed system P and alternative system A. While System A generates a strong under-segmentation, System P achieves a finer and more accurate segmentation.

A finer segmentation requires a better balancing between recall and precision (especially a higher precision). System P yields a higher precision and overall f_1 score. This significantly improves the accuracy of the sequence extraction. Again this is best observed from Figure 8.6 (upper part) where the extracted sequences much better match with the ground truth. Most of the ground truth sequences are partly or entirely retrieved. There are only a few short false positive sequences. The increase of precision is due to the consideration of the isolation and salience condition in the weighting function.

From Table 8.1 we further observe that the proposed method (System P) yields higher recall and precision for sequence extraction (task #2) than for single shot cuts (task #1), although both tasks build upon each other. The reason lies in the tolerance of the segmentation scheme which is able to compensate for falsely classified shot cuts.

Table 8.2 presents the retrieval performance in terms of precision for differently sized result sets (short "Prec@N" for a result set of size N) for the films from Section 8.4.1. We obtain the different result sets by retrieving only the N sequences with the highest confidence. Among the first ten retrieved sequences in average 72% are relevant synchronous audio-visual montage sequences. Furthermore, in the film "Fight Club" where no prior information about the montage style was available we discov-

Feature Film	Prec@1	Prec@3	Prec@5	Prec@10
Enthusiasm	1.00	1.00	1.00	0.90
October	1.00	0.67	0.80	0.50
The Hunt for Red October	1.00	0.67	0.60	0.70
Fight Club	1.00	0.67	0.80	0.80

Table 8.2: Precisions of the proposed method for different result set sizes (1, 3, 5, and 10) and feature films.

ered several synchronous montage sequences. False positives are returned mostly in situations where a lot of background noise is present in the soundtrack.

From Table 8.2 we observe that the performance for historic material is similar to that of the contemporary material. This is remarkable since the historic material contains numerous artifacts in the visual signal (e.g. flicker, shaking, low contrast) as well as in the audio track (e.g. broad-band noise, distortions). There are two reasons for the robustness of the approach. First, we rely on visual and audio onsets which correspond to peaks that are robust to noise to a high degree. Second, even in case of falsely detected onsets, the tolerant segmentation scheme compensates for most of these errors.

The retrieved results include sequences of high semantic interest. For example the top ranked sequence in "Enthusiasm" is the already mentioned sequence of religious symbols from Section 8.1. An illustration of the sequence together with the audio novelty curve is shown in Figure 8.7. The peaks clearly correspond to the church bells at the shot cuts.

An interesting observation concerning the sequence in "Enthusiasm" is made from the results for the film "October". One retrieved sequence from "October" shows a similar sequence of religious symbols which are emphasized by bell sounds at each shot cut. Since "October" was produced before "Enthusiasm", the soundtrack however much later, the presumption comes up that both films mutually influenced each other. This example illustrates that the proposed method is able to hint at correspondences between different films.

For the contemporary material the retrieved sequences contain fast and synchronously cut dialogue sequences (e.g. discussions and arguments between protagonists) and action sequences (fights, shootings, accidents). The proposed method for example

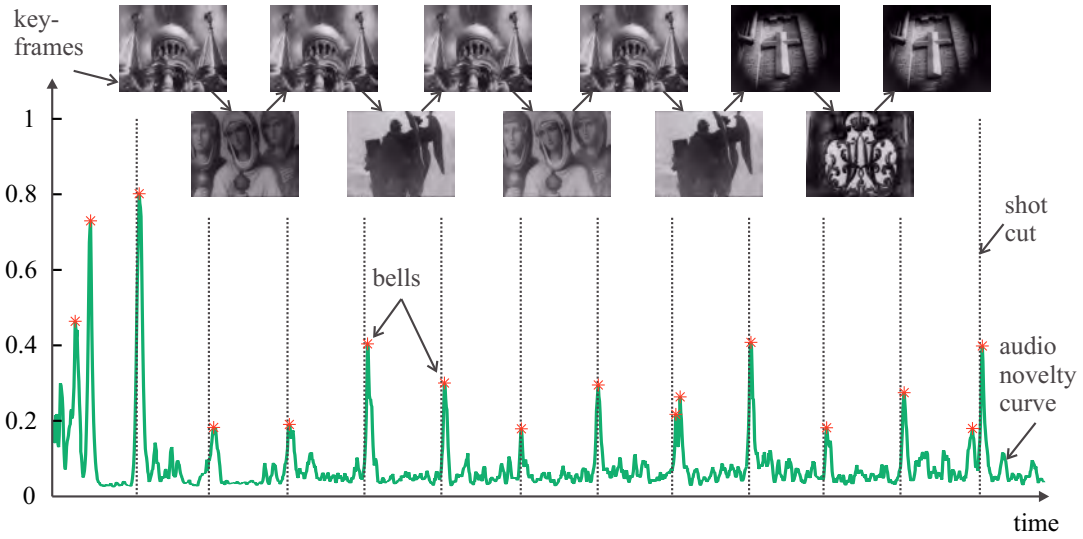


Figure 8.7: A sequence showing different religious symbols with synchronously edited bell sounds at each shot cut.

retrieves a fast and synchronously cut dialogue sequence (between the narrator and Tyler Durden beginning at 1:54:14) from the film "Fight Club". In "The Hunt for Red October" the method retrieves a key scene that shows a parallel montage of a nuclear missile and the main character Jack Ryan who unmasks and shoots a saboteur and thereby prevents the explosion of the missile. Generally, the extracted sequences are semantically important in most cases and may enrich further high-level tasks such as film abstraction, indexing, and summarization.

8.6 Summary

Filmmakers employ the synchronous montage technique to increase the tension of a sequence and to highlight important events. The detection of such sequences enables a new way for the extraction of semantically meaningful information from films. In this chapter we have presented an approach for the automated extraction of such sequences based on a novel method for cross-modal temporal correlation estimation and a tolerant segmentation scheme. We first extract shot cuts and generic audio onsets by a self-similarity analysis of the visual and auditory stream. Next we compute temporal cross-modal correlations between shot cuts and audio onsets in a way that takes synchrony perception of humans into account. The detected synchronously and asynchronously

edited shot cuts are input to segmentation and sequence extraction. Since synchronous montage sequences may contain irregularities (asynchronously edited shot cuts) we develop a tolerant segmentation scheme. The segmentation scheme maximizes the number of synchronously edited shot cuts in a sequence and at the same time minimizes the number of asynchronously edited shot cuts (irregularities).

Experiments with historical and contemporary films show that the retrieved sequences contain rich semantics which makes them suitable for high-level film analysis tasks. The novel method for correlation estimation outperforms a state-of-the-art approach for audio-visual synchronicity detection. The proposed method yields a better balance between recall and precision which results in a significantly more accurate segmentation.

The proposed method is a further step towards the automated extraction of semantically meaningful information for high-level film annotation, interpretation and summarization. The extraction of synchronous montage sequences (similarly to the extraction of scenes) reveals the temporal composition of a film. Another aspect of composition in film is motion. We investigate the analysis and retrieval of motion composition in the next chapter.

Chapter 9

Retrieval of Motion Composition

In the previous chapters we have focused on the retrieval of concepts related to the temporal composition of a film such as shots, scenes, and synchronous montage sequences (in Chapters 5, 7, and 8). In this chapter, we focus on another higher-level concept related to the composition of a film, namely *motion*. We first introduce the role of motion (camera motion and object motion) in filmmaking in Section 9.1. Next, we present a robust method for the extraction of meaningful motion components from a film in Section 9.2. The method clusters motion trajectories into long-time motion segments and provides a compact description of the motion content in a sequence. In a next step, we introduce an intuitive query interface for the description of motion. Based on this interface, we investigate two retrieval scenarios: (i) the retrieval of motion compositions in a shot in Section 9.3 and (ii) the retrieval of motion continuity between two consecutive shots in Section 9.4. For both retrieval scenarios appropriate matching schemes are presented that match the queries with the previously extracted motion components.

9.1 Introduction

Camera and object motion characterize the style of a film and significantly influence the way it is perceived by viewers. Motion controls the tempo in a scene, creates visual rhythm and gives a film temporal continuity. Furthermore, motion gives evidence about the genre of a film and about the director's style. Motion is of special interest to filmmakers and film scientists - however from different perspectives. Filmmakers

employ motion for example to increase the tension in a scene. This may be achieved by using a shaky camera (steadicam) and showing fast moving objects. Film scientists otherwise reverse the process of filmmaking and manually analyze in detail for example the usage of camera and object motion in order to investigate the progression of tension over an entire film or a filmmaker's style.

Automatic methods for motion retrieval support both, film scientists and filmmakers in their creative work by allowing efficient search and retrieval of particular types of motions and motion compositions. Especially film scientists manually analyze motion shot by shot and in great detail which is a tedious, time-consuming and error-prone task. Automatic motion retrieval supports the expert in finding typical motion directions, locations and combinations of interest more efficiently and sometimes also more accurately. Consequently, automatic motion retrieval is a useful device in film studies.

The efficient retrieval of motion and motion compositions from a film requires the robust extraction and segmentation of meaningful motion components. Additionally, an intuitive query interface is needed that allows the user to define motion compositions as search requests. Finally, tolerant matching schemes are necessary, that match the abstract query description provided by the user with the automatically extracted motion components from the film.

9.2 Motion Segmentation by Trajectory Clustering

Humans have the ability to easily interpret and abstract from complex motion patterns. For example, consider a scene where a group of people moves from left to right. The motion that most observers keep in mind is the motion direction of the entire group and not the detailed motion of each individual. Consequently for efficient motion retrieval, methods are required that are able to describe complex motion compositions in terms of a few abstract and meaningful motion components.

Many different methods for the analysis and description of motion have been proposed. Two basically different groups of approaches can be distinguished. The first one aims at the segmentation and tracking of objects (or object regions) to represent the dynamic content of a scene [10, 36, 40, 75, 137]. Approaches in this group usually start with color segmentation of the frames [76] and then perform tracking of the segmented regions over time (e.g., by Kalman filtering [36]). Finally, a trajectory for each object

can be computed by tracking the centroid of each region [50]. However, some types of motion cannot be represented by these approaches because the corresponding objects are difficult to segment, such as groups of objects (e.g., people, cars), motion of water (e.g., rivers), smoke, etc. Furthermore, segmentation-based approaches require high quality (color) video material which is not the case for archive film material.

The second group of approaches extracts a texture pattern from a dense motion field [30, 61, 156] and does not perform object segmentation. This texture pattern (represented by statistical texture descriptors) is characteristic for different types of complex motions (e.g., of crowds, water, grass) [156]. Both groups of approaches are applicable only to a subset of motion types, for example either single objects or special types of objects like water and smoke. For the retrieval of motion in film a method is desired that is able to describe many different types of motions.

In the following, we present a robust and efficient approach for the segmentation of single object motion as well as motion of groups of objects and camera motion. In contrast to other approaches we extract motion trajectories by feature tracking directly from the raw film sequence and omit object segmentation. The result of feature tracking is a sparsely populated spatio-temporal volume of feature trajectories. Occlusions and the low quality of the film material lead to numerous tracking failures resulting in noisy and highly fragmented trajectories. The novel clustering scheme directly clusters the sparse volume of trajectories into meaningful spatio-temporal motion components belonging to the same objects or groups of objects.

The proposed approach takes the following factors into account:

- Applicability to low-quality monochrome film and video material that contains low contrast, flicker, shaking, dirt, etc.
- The input data is a sparse set of fragmented trajectories that are broken off, have different lengths, and varying begin and end times.
- The analyzed time span may be large (shots up to a few minutes length).
- The number of clusters is unknown a priori.
- The resulting clusters have to be temporally coherent (even if the majority of the trajectories breaks off).

- Motion direction and velocity magnitude may change over time inside a cluster (to enable tracking of e.g. objects with curved motion paths and groups of objects that have slightly heterogeneous directions and speeds).
- Efficient computation.
- Flexible selection of trajectory features and similarity measures in order to enable different clusterings (e.g. to allow also spatial information during clustering).

9.2.1 Related Work

The basis for motion analysis is usually a dense motion field, which can be obtained from an optical flow estimator, as in [39, 40] or directly retrieved from MPEG compressed video [54, 60]. Dense optical flow computation is time consuming and can hardly be applied to a full-length feature film. Furthermore, as experiments with our material have shown, optical flow algorithms (Horn and Schunk, Lucas Kanade) yield highly disturbed motion fields when applied to low-quality material.

Another drawback of optical flow methods is that motion fields obtained from optical flow and compressed video usually represent motion between only two frames. However, the analysis of the motion content of entire shots requires coherent motion information over all or at least a large number of successive frames.

Feature trackers are able to provide motion information over large time scales by tracking feature points over multiple successive frames [194]. Feature tracking has several properties that make it suitable for our task. First, the tracked trajectories provide information over large time scales and have high precision since they are based on distinct feature points. Second, feature trackers capture motion only where it actually appears in a sequence, which reduces the amount of data for further processing significantly in comparison to dense motion fields. Third, feature tracking can be performed efficiently in near real-time [195]. The major drawback of feature tracking is that the resulting feature trajectories are sparse in space and in time. The sparse nature of the motion trajectories impedes the clustering of the spatio-temporal volume. Trajectories have different lengths and varying begin and end times. Consequently, standard methods, such as Mean Shift and K-Means cannot be directly applied.

Methods for trajectory clustering have been introduced mainly in the field of surveillance [36, 122, 137, 170, 230, 233, 234] and video event classification [93]. The methods

have different constraints depending on their application domain. Wang and Li present an approach for motion segmentation based on spectral clustering of motion trajectories [233]. A major limitation of their approach is that all feature points must be trackable in all analyzed frames. This is usually not the case, especially when working with low-quality film material.

Hervieu et al. perform classification of motion trajectories by an HMM framework for video event classification in sports videos [93]. While the HMM framework is able to handle trajectories of different lengths, the method assumes that the trajectories have similar lifetimes and do not break off (e.g., due to occlusions and tracking failures). A similar assumption is made in [122] where the authors track and cluster motion paths of vehicles. They represent trajectories by global directional histograms which require similar motion trajectories to have similar lifetimes. However, this is not provided for broken trajectories.

Veit et al. introduce an approach for trajectory clustering of individual moving objects [230]. Groups of similarly moving objects, as required in our work, are not tracked by the approach. Furthermore, the analysis windows are about one second, which is inadequate for long-term analysis of e.g., an entire shot. Similarly, Rabaud and Belongie cluster motion trajectories on a per-object basis in order to count people in a surveillance video [170].

For clustering a sparse set of trajectories similarity measures are required that take the different lengths and spatio-temporal locations of the trajectories into account. Buzan et al. employ a metric based on the longest common subsequence (LCSS) to cluster trajectories of different sizes [36]. An asymmetric similarity measure for a pair of trajectories of different lengths is proposed by Wang et al. [234]. Their algorithm can handle broken trajectories during clustering due to the asymmetric property of the similarity measure. The measure is not directly applicable in our work because it uses different (spatial) similarity constraints. However, the way we compute similarity between two trajectories during clustering is similar to this method.

There is an important difference between the above mentioned methods and our approach. The presented methods do not consider the temporal location of the trajectories during clustering. They aim at clustering trajectories independently of the time they occur (e.g., similar trajectories of different vehicles are grouped independently of the time they occur) [122]. In this work, we are interested in clustering of motion

trajectories belonging to the *same* object or group of objects. Therefore, we assume corresponding trajectories to occur within the *same* time (and to have similar velocity direction and -magnitude). Note, that we do not require the trajectories to have similar spatial location, which facilitates tracking of large groups of objects and camera motion (in contrast to e.g. [230]).

9.2.2 Trajectory Clustering

The idea behind the proposed scheme is to cluster the entire sparse volume of trajectories directly by iteratively grouping temporally overlapping trajectories. Thus, it is not necessary to split trajectories into sub-trajectories [19] or use global trajectory features [122]. The trajectories are processed in their original representation. Figure 9.1 gives an overview of the approach.

The input of clustering is a sparse set of fragmented trajectories obtained from feature tracking. In a first stage of the algorithm an iterative clustering scheme groups temporally overlapping trajectories with similar velocity direction and -magnitude. Thereby, one trajectory may be assigned to multiple clusters. In the second stage the clusters from the first stage are merged into temporally adjacent and disjoint clusters covering larger time spans. Merging exploits the redundancy (multiple assigned trajectories) contained in the input clusters, see Section 9.2.2.

Iterative clustering

Iterative clustering aims at successively grouping temporally overlapping trajectories. A basic assumption is that trajectories that perform similar motion at the same time belong to the same motion segment. According to this definition, a segment can represent motion of a single object, a group of several similarly moving objects as well as motions of the camera. Consequently, segmentation is not restricted to a particular type and source of motion which is important for the retrieval of arbitrary motion compositions.

A trajectory t is a sequence of spatio-temporal observations $o_j = < x_j, y_j, f_j >$ with $t = \{< x_j, y_j, f_j >\}$, where x_j and y_j are spatial coordinates and f_j is the frame index of the corresponding observation. The input of the algorithm is a sparse spatio-temporal volume which is represented as a set V containing T trajectories t_i of tracked feature points: $V = \{t_i | i = 1, 2, ..., T\}$.

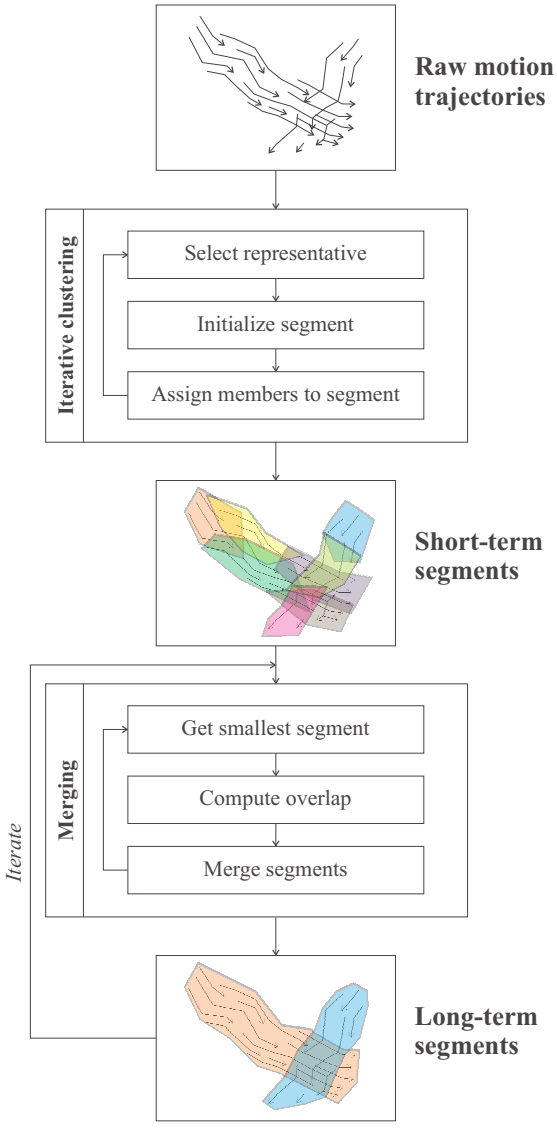


Figure 9.1: The process of motion segmentation.

Clustering starts by the selection of expressive trajectories (representatives of meaningful motion components) for the initialization of clusters. We assume that meaningful motion components span large distances. Therefore, we compute the absolute spatial distance that each trajectory travels during its lifetime. That is the Euclidean distance between the first and last feature point of the trajectory. This measure favors trajectories that belong to an important motion component. Alternatively, the lifetime of the trajectories may be employed as a measure for their expressiveness. However, experiments have shown that the trajectories with the longest lifetimes often represent stationary or slowly moving points, which leads to the selection of inadequate representatives. Consequently, we do not employ the lifetime as an indicator for expressiveness.

We sort the trajectories according to their traveled distances and select the trajectory t_r with the largest distance as representative for the current cluster C_{t_r}. Then all trajectories t_i from the set V are compared to the representative t_r in a pairwise manner. The similarity of trajectories that have no temporal overlap is 0 by definition. Consequently, only temporally overlapping trajectories are compared.

For the pairwise comparison first the temporally overlapping sub-segments of two trajectories t_r and t_i are determined. Following, we extract trajectory features from these sub-segments and a perform similarity comparison. See Section 9.2.2 for the description of the employed trajectory features and similarity measures. The result of the pairwise comparison of trajectories t_r and t_i is a similarity score $s^{r,i}$.

The similarity score is then compared to a threshold λ. All trajectories with a score higher than λ are assigned to the current cluster C_{t_r}:

$$t_i \in C_{t_r} \Leftrightarrow s^{r,i} > \lambda \tag{9.1}$$

The cluster C_{t_r} is then added to the set S of clusters (which is initially empty).

In the next step the original set of trajectories V is updated. All trajectories $t_i \in C_{t_r}$ that lie fully inside the cluster are removed from the original set of trajectories V. Trajectories that are temporally not fully covered by the cluster remain in V. That enables trajectories to be assigned to multiple temporally adjacent clusters in further iterations. This is an important prerequisite for the creation of long-term clusters in the second stage of the algorithm.

After updating the set V the next iteration is started by selecting a new representative trajectory t_r from the remaining trajectories in V. The algorithm terminates

Input: Sparse set of T trajectories $V = \{t_i | i = 1, 2, ..., T\}$
Output: Set of n clusters $S = \{C_1, C_2, ..., C_n\}$
Algorithm:

1. Initialize: $S = \{\}$
2. Sort trajectories $t_i \in V$ by their traveled distance (descending).
3. Select t_r with maximum distance from V, initialize Cluster C_{t_r} as $C_{t_r} = \{t_r\}$, remove t_r from V.
4. For all remaining trajectories t_i in V:
   5. Compute similarity $s(t_i, t_r)$
   6. If $s(t_i, t_r) > \lambda$ then $C_{t_r} = C_{t_r} \cup \{t_i\}$
7. Update: add C_{t_r} to S, remove trajectories from V that lie entirely in C_{t_r}
8. Resume with step 2 until $V = \{\}$.

Figure 9.2: The iterative clustering scheme.

when no more trajectories are left in V. Figure 9.2 presents a compact listing of the iterative clustering scheme.

The result of iterative clustering is a set of n overlapping trajectory clusters $S = \{C_1, C_2, ..., C_n\}$ where each cluster represents a portion of a homogeneous motion component. The temporal extent of the clusters tends to be rather short (it is limited by the temporal extent of the feature trajectories). Consequently, the iterative clustering yields an over-segmentation of the spatio-temporal volume. This is addressed in the second stage (merging), see Section 9.2.2.

Trajectory features and similarity measures

The proposed iterative clustering scheme allows for the use (and combination) of arbitrary features and similarity measures, for example spatial features compared by Euclidean distance, purely directional features compared by Cosine similarity, etc. Additionally, the combination of features requiring different similarity or distance measures is allowed.

We compute features adaptively only for the temporally overlapping segments of the compared trajectories. First, the temporally overlapping segments of the trajectories are determined. Following, feature extraction is restricted to these segments. This is different from other approaches, where features are computed a priori for the entire trajectories.

A straight forward way is to directly employ the spatial coordinates of the trajectories as features. For low-quality material the coordinates of the trajectories are noisy (e.g., due to shaky sequences and tracking failures). Consequently, more robust features are required. For a given segment of a trajectory we compute the dominant direction $\phi = (\Delta x, \Delta y)$ where

$$\Delta x = x_{begin} - x_{end},\ \Delta y = y_{begin} - y_{end}, \tag{9.2}$$

and the distance ρ between the first and the last spatial coordinates of the segment:

$$\rho = \sqrt{(\Delta x)^2 + (\Delta y)^2}\ . \tag{9.3}$$

These features are robust to noise and are location invariant (as required for segmenting motion from the camera and of groups of objects). They represent the velocity direction and magnitude of the trajectories. Dependence on spatial location can easily be integrated by adding absolute coordinates as features.

The presented features require two different metrics for comparison. We employ the Cosine similarity for the directional features ϕ and a normalized difference for the distance features ρ. The corresponding similarity measures s_ϕ and s_ρ for two trajectories u and v are defined as follows:

$$s_\phi^{u,v} = \frac{1}{2} \cdot \left(\frac{\phi^u \cdot \phi^v}{\|\phi^u\| \cdot \|\phi^v\|} + 1 \right),\ s_\rho^{u,v} = 1 - \frac{|\rho^u - \rho^v|}{\max(\rho^u, \rho^v)}\ . \tag{9.4}$$

The Cosine similarity is transformed into the range $[0; 1]$. We linearly combine both similarity measures in order to obtain a single similarity measure $s^{u,v}$ as:

$$s^{u,v} = \alpha \cdot s_\phi^{u,v} + (1 - \alpha) \cdot s_\rho^{u,v} \text{ with } 0 \leq \alpha \leq 1, \tag{9.5}$$

where α balances the influence of the velocity directions and the velocity magnitudes of the two trajectories.

Cluster merging

The presented iterative clustering typically yields an over-segmentation of the trajectories. The goal of cluster merging is to connect clusters that represent the same (long-time) motion component. This is performed by hierarchically merging clusters which share the same trajectories.

The input to this stage is the set of clusters obtained by iterative clustering: $S = \{C_1, C_2, ..., C_n\}$. We start an iteration by sorting the clusters according to their sizes (number of member trajectories) in ascending order. Beginning with the smallest cluster C_i, we search for the cluster C_j which shares the most trajectories with C_i. We merge both clusters when the portion of shared trajectories (connectivity) exceeds a certain threshold μ. The connectivity $c_{i,j}$ between two clusters C_i and C_j is defined as:

$$c_{i,j} = \frac{|C_i \cap C_j|}{\min(|C_i|, |C_j|)} \tag{9.6}$$

The criterion for merging clusters C_i and C_j into a new cluster $C_i^{'}$ is:

$$C_i^{'} = C_i \cup C_j \Leftrightarrow c_{i,j} > \mu. \tag{9.7}$$

After merging the clusters C_i and C_j they are removed from the set S and the new cluster is added into an (initially empty) set $S^{'}$. If no cluster C_j fulfills the criterion for merging then $C_i^{'} = C_i$. Following, C_i is removed from S and $C_i^{'}$ is added to $S^{'}$.

Merging is repeated with all remaining clusters in S, until S is empty and $S^{'}$ contains all combined clusters. This makes up one complete iteration of merging. We perform further iterations by setting $S = S^{'}$ to repeatedly merge newly created clusters until no cluster can be merged any more. Finally, trajectories associated with more than one cluster are assigned to the cluster with the largest temporal overlap. The result of the merging procedure is a smaller set of clusters $S^{'}$ where the clusters represent distinct (long-term) motion components. See Figure 9.3 for a compact listing of the algorithm.

The order in which clusters are merged influences the result significantly. We sort the clusters according to their size and begin merging with the smallest clusters. This supports the merging scheme to successively generate larger clusters out of small ones (fewer small clusters remain). Furthermore, each cluster is merged with the one having the highest connectivity. This facilitates that clusters belonging to the same motion component are merged.

Input: Set of n clusters $S = \{C_1, C_2, ..., C_n\}$
Output: Set of m ($m \leq n$) clusters $S' = \left\{C'_1, C'_2, ..., C'_m\right\}$
Algorithm:

1. Initialize: $S' = \{\}$
2. Sort clusters in S according to size (ascending)
3. For all clusters C_i in S, beginning with the smallest:
   4. Find cluster C_j with highest connectivity $c_{i,j}$ to C_i
   5. If $c_{i,j} > \mu$ then $C'_i = C_i \cup C_j$, remove C_i, C_j from S
   6. Else $C'_i = C_i$, remove C_i from S
   7. Add C'_i to S'
   8. Resume with step 3 until all clusters are processed.
9. Update: $S = S'$
10. Resume with step 2 until no clusters can be merged any more.

Figure 9.3: The cluster merging procedure.

9.2.3 Experimental Setup

In this section, we describe the motion analysis framework that was used for the experiments. The framework includes preprocessing steps (shot segmentation, motion tracking and filtering of the motion field) and some postprocessing steps that were added due to the low-quality of the archive film employed in the experiments.

Film material

The archive film material exhibit twofold challenges, that originate from their technical and from their artistic nature. From the technical point of view, the film material is of significantly low quality due to storage, copying, and playback over the last decades, as described in Section 3.2. Low contrast and flicker impede the process of feature detection and tracking, resulting in noisy and broken feature trajectories. Furthermore, frame displacements and significant camera shakes result in falsely detected motion.

From an artistic point of view, the historic documentaries contain a large number of differing motion compositions. Dziga Vertov used advanced montage and photographic

techniques (e.g., quadruple exposure, reverse filming, etc.) to achieve complex motion compositions, including complex camera travelings, contrapuntal movements, and typical work activities like hammering (see Figure 9.11 in Section 9.3.3 for examples).

Shot segmentation

Prior to motion analysis, we segment the films into shots. The subsequent motion analysis is then performed for each shot separately. For shot segmentation the method described in Chapter 5 may be employed. However, for the evaluation of trajectory clustering, we have determined the shot boundaries manually, in order to assure that the evaluation of our method is not influenced by segmentation errors.

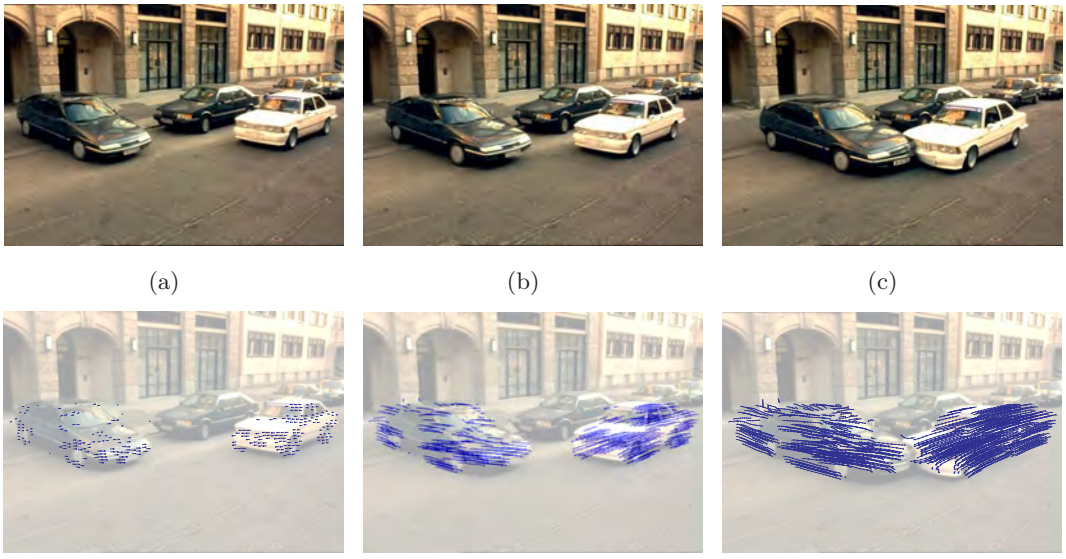
Feature tracking

Feature trackers first select distinct points in a frame (e.g. corners of objects) and then attempt to trace these points over time in subsequent frames. Points that cannot be tracked any further (e.g. because they move out of the frame or get occluded) are usually replaced continuously by new points.

We employ the Kanade-Lucas-Tomasi (KLT) feature tracker because of its efficiency and its ability to track feature points across large time spans [194]. KLT combines feature selection and tracking into a single process. The tracker favors feature points that can be tracked well which improves the expressiveness of the resulting motion trajectories. Figure 9.4 shows the trajectories of tracked feature points for contemporary film material (from the film "Run Lola Run"). The sequence shows two cars that move towards each other and finally crash.

For most parameters of KLT we use the defaults proposed by the implementation in [23]. The search range for tracking is set to 3 which reduces the number of tracking errors significantly with the historic film material. The minimum distance between selected features is reduced to 5 in order to produce denser motion fields especially in areas where tracking is difficult because of low contrast. The number of features is set to 2000 (which is a good tradeoff between available motion information and the computational effort for processing the trajectories). The tracker is configured to immediately replace lost features by new ones.

The output of feature tracking is a fragmented set of trajectories in a sparse motion field. The trajectories are noisy and have low homogeneity due to tracking failures



(a) (b) (c)

(d) (e) (f)

Figure 9.4: Motion trajectories obtained from the KLT feature tracker for contemporary film material.

induced by the low-quality film material. See Figure 9.6(a) for an example of a motion field for the historic material. Some filtering of the motion field is necessary to reduce the amount of noise in the motion field.

Filtering of the motion field

We perform three basic preprocessing steps in order to reduce the noise contained in the motion field. First, we remove trajectories whose lifetime is less than a predefined duration τ ($\tau = 0.5$ seconds in the experiments). This removes a large number of unstable trajectories.

Second, we detect and remove stationary trajectories. For each trajectory, we compute the maximum spatial extent along the x- and y-axis. If both are smaller than a threshold σ, the trajectory is classified as stationary and removed (see Figure 9.5 for an illustration). The threshold σ directly corresponds to the amount of shaking in the sequence and thus can be easily determined. For the employed material σ is approximately 1% of the width of a frame ($\sigma = 7$). We remove trajectories with a spatial extent in x- and y-direction below a threshold σ which directly corresponds to the amount of shaking in the sequence.

Figure 9.5: Three examples for stationary trajectories. The black (solid) rectangles mark their spatial extent. The red (dashed) rectangles show the maximum tolerance σ for stationary trajectories.

(a) (b)

Figure 9.6: The effect of filtering: the motion field before (left) and after (right) filtering.

Third, we smooth the trajectories by removing high-frequency components in the discrete Cosine spectrum of the spatial coordinates x_j and y_j. This dampens the influence of shaking for the remaining trajectories.

Preprocessing reduces the number of input trajectories of up to three orders of magnitude for highly noisy sequences. Figure 9.6 shows the effect of filtering for a motion field accumulated over an entire (noisy) shot. The shot shows an airplane that moves through the scene from left to right and contains a large amount of noise. Filtering removes most of the stationary and noisy trajectories. The remaining trajectories represent the motion of the airplane in the lower right quarter of the frame.

Clustering parameters

The proposed clustering approach requires three parameters to be set (λ and α for the similarity comparison and μ for cluster merging). The similarity score $s^{u,v}$ as defined in

Equation (9.5) in Section 9.2.2 ranges from 0 to 1, where 1 denotes the highest similarity. A value of λ between 0.7 and 0.9 yields satisfactory results in the experiments. The weighting factor α of the combined similarity score $s^{u,v}$ is set to 0.5 which means that direction and magnitude equally influence the similarity computation.

The second parameter μ controls the sensitivity of cluster merging. It specifies the minimum portion of shared trajectories necessary to merge two clusters. Due to the high fragmentation of the trajectories the value of μ is chosen rather low to facilitate cluster merging. Values of μ between 10% and 20% of shared trajectories yield the best results in the experiments.

Postprocessing

We perform two simple postprocessing steps in order to improve the generated motion segments. First, we detect and remove outlier trajectories. Since we ignore spatial information during clustering some clusters may contain outlier trajectories (trajectories which are spatially not connected to the main region of the cluster). For each trajectory of a cluster, we compute the number of trajectories in its neighborhood and remove trajectories without neighbors. This yields more stable clusters with less spatial fragmentation.

Second, we remove small clusters (less than 5 trajectories) since they usually represent noise. In the experiments clusters with less than 5 trajectories are removed. This threshold is set very low, since we want to evaluate whether the clustering algorithm is able to segment even small distinct motions which frequently occur in the material.

9.2.4 Experimental Results

We perform qualitative evaluation by applying our approach to shots with complex motion compositions from the available film material. Some of these shots are highly disturbed by noise.

A number of papers addressing motion analysis report results only for selected sequences [39, 60, 230]. We aim at performing a quantitative evaluation in order to test our method on a large number of shots containing diverse types and motion compositions. Therefore, we apply the algorithm to an entire feature film and evaluate the clustered motion components for each shot.

Qualitative evaluation.

We have selected approximately 50 shots from different archive films for evaluating the quality of our approach. Three test sequences are shown in Figures 9.7-9.9. For each sequence, we provide three keyframes from the beginning, middle, and end. White arrows and ellipses illustrate the dominant motion components. The fourth image shows the clustered feature trajectories and the last image depicts the spatial extent and the primary motion direction of each cluster.

The first sequence (Figures 9.7(a) - 9.7(e)) shows a group of people walking up a hill. The people in the group first move towards the hill (in the lower right quarter), then turn to the left, walk up the hill and finally vanish behind the hill. At the end of the sequence a mule enters the scene at the top of the hill in opposite direction (short arrow in Figure 9.7(c)). From the three keyframes 9.7(a)-9.7(c) we observe a large amount of flicker, additionally some frames contain scratches and dirt as in 9.7(a). Since KLT is sensitive to intensity variations the trajectories frequently break off. However, our approach is able to create temporally coherent motion segments over the entire duration of the shot. The movement of the group of people is represented by segments 1 and 2 (blue and yellow) in Figure 9.7(d). Segment 1 represents the motion of the people away from the camera and segment 2 captures the people walking up the hill. The third segment (red) represents the mule that appears at the end of the scene from the left. The corresponding cluster is small, since the mule is visible only for the last 1.5 seconds. This sequence shows that the approach is able to segment large groups of objects as well as small individual objects. Furthermore, the method supports segmentation of long- as well as short-term motion.

The second sequence (Figures 9.8(a) - 9.8(e)) shows an airplane moving from left to right. The airplane approaches the observing camera and finally passes it. The sequence is shot by a camera that itself is mounted on an airplane, resulting in permanent shaking. Several frames of the shot are heavily blurred (e.g. 9.8(a)) making feature tracking nearly impossible. Clustering of the shot yields three motion segments, shown in Figures 9.8(d) and 9.8(e). The motion of the airplane is represented by segments 1 and 2 (yellow and red). The first segment describes the motion of the airplane from the beginning of the shot to the last quarter of the shot. The second segment continues tracking this motion until the end of the shot. While the two segments are temporally

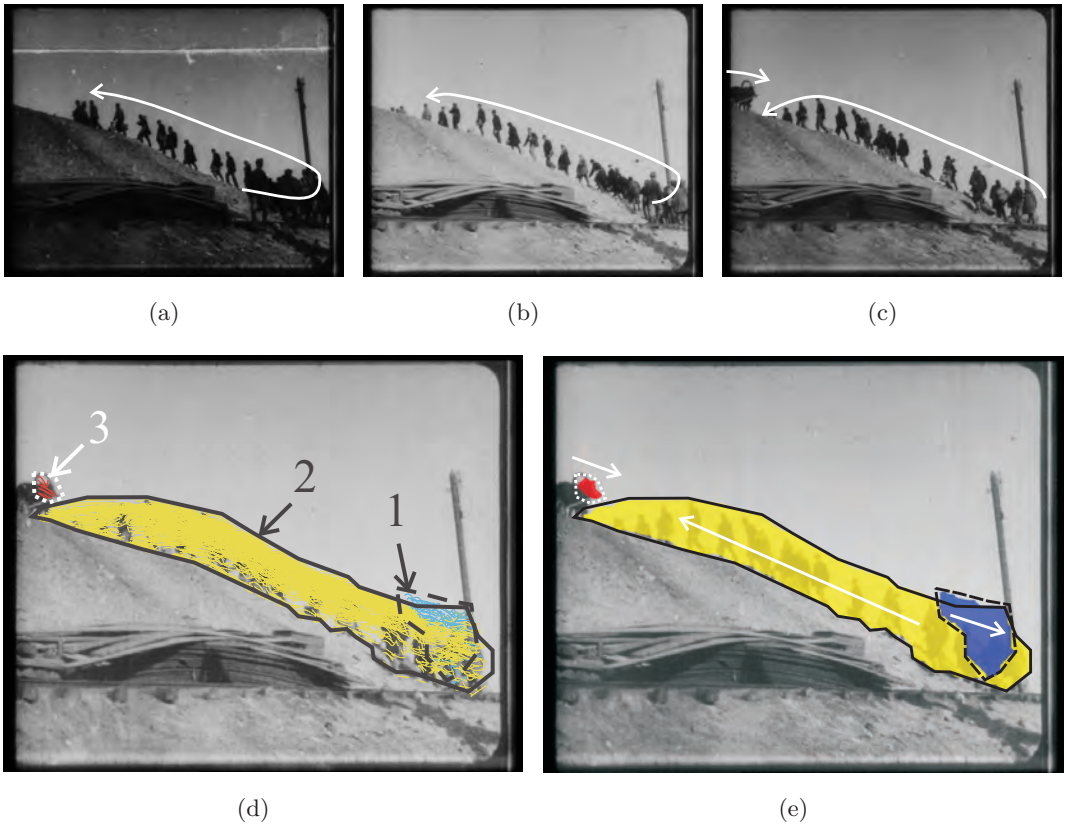


Figure 9.7: Segmentation results of the first sequence. Figures (a)-(c) represent keyframes (white annotations mark the dominant motion components). Figures (d) and (e) show the clustered trajectories and the resulting motion segments with their primary direction.

coherent they are not merged by our algorithm because the (noisy) motion field that connects them is too sparse. The third segment (blue) describes an intense camera shake that is not removed during filtering the motion field. We further observe, that the first segment contains some trajectories originating from the shaky camera. This shot is one of the sequences with the strongest distortions. It demonstrates the limitations for feature tracking and motion segmentation.

The third sequence shows a herd of horses (surrounded by the white ellipses in Figures 9.9(a) - 9.9(c)) moving diagonally into the scene from left to right. At the same time the camera pans to the right (indicated by the dashed arrows). The camera motion can be recognized best by observing the house in the top right corner that moves slowly from right to the left over the three keyframes. Both motion components

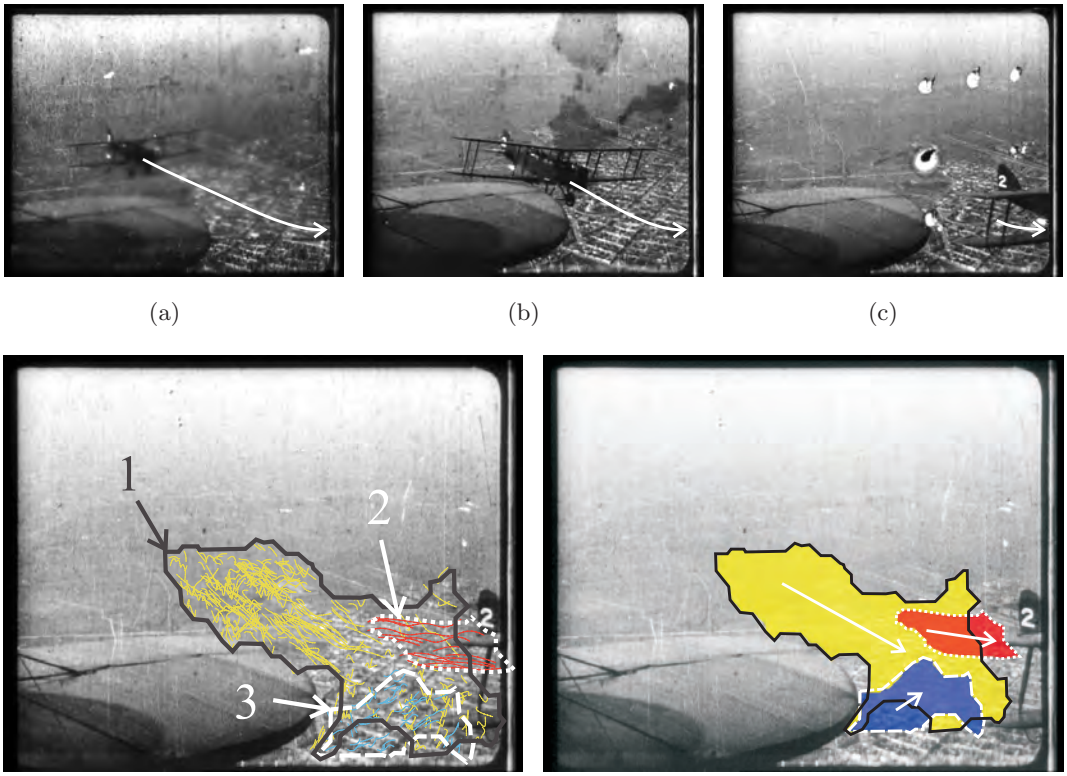


Figure 9.8: Segmentation results of the second sequence. Figures (a)-(c) represent keyframes (white annotations mark the dominant motion components). Figures (d) and (e) show the clustered trajectories and the resulting motion segments with their primary direction.

are tracked and separated from each other by our approach. The spatially distributed segment (segment 1, yellow) in Figure 9.9(d) represents the camera motion, while the second segment (red) describes the motion of the herd. Not all individuals of the herd can be tracked robustly by KLT due to the low contrast between the horses and the background. However, the motion trajectories available from tracking are correctly segmented. Note that the dashed arrows in Figure 9.9(e) represent the motion direction of the camera and not that of the feature points relative to the image plane.

Motion segmentation performs well for the presented sequences. Even under noisy conditions the method robustly segments the motion components. A systematic evaluation of the method on a larger dataset is presented in the next section.

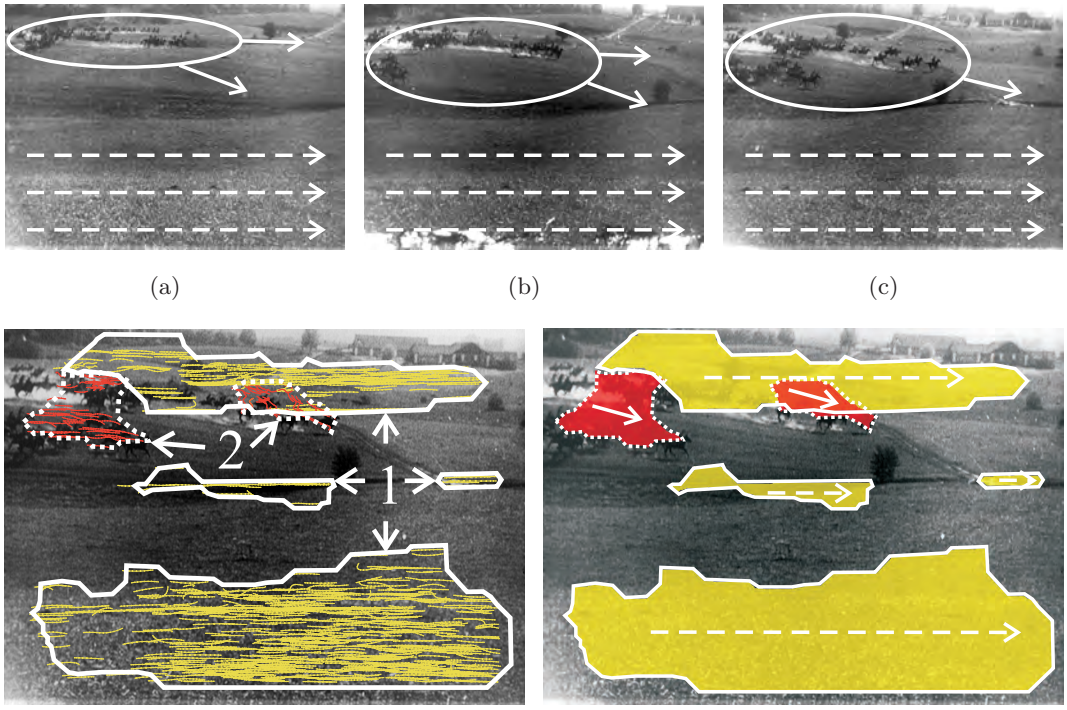


Figure 9.9: Segmentation results of the third sequence. Figures (a)-(c) represent keyframes (white annotations mark the dominant motion components). Figures (d) and (e) show the clustered trajectories and the resulting motion segments with their primary direction.

Quantitative evaluation.

We apply the proposed approach to an entire feature film, in order to perform a quantitative performance evaluation. We select the film "The Eleventh Year" from 1928 because it makes extensive use of motion compositions. The film shows the life of workers of the 1920s and contains a large number of motion studies of physically working people, crowds, industrial machines (e.g., moving pistons), and vehicles (e.g., cars, trains). The film contains 63123 frames that have been manually segmented into (660 shots) and has a duration of approximately one hour (at 18 fps).

Creating a precise ground truth for motion segmentation is a non-trivial task, since it requires the annotation of moving objects along the spatial and temporal dimension. That principally means that the shapes and positions of all moving (possibly non-rigid) objects (or groups of objects) have to be annotated at each time instance. We have

created the ground truth together with film experts manually. For each shot, we count the number of motion components and create a textual description of the objects and their motion activity. We skip shots with a duration shorter than 0.5 seconds (which is the minimum trajectory lifetime τ in our experiments, see Section 9.2.3). This yields a total number of 607 shots. Evaluation is performed manually by comparing the computed motion segments with the ground truth protocol and applying the following rules:

1. A motion component is considered to be correctly detected if one or more clusters exist with similar spatio-temporal locations and similar directions. Otherwise the motion component is considered to be missed.

2. A cluster is considered to be a false positive, if it cannot be assigned to any motion component.

The proposed method is able to segment 60% of all motion components in the film. This low detection rate is a consequence of a poor feature tracking performance. While related literature reports excellent results of KLT for high-quality video [183], the tracker misses 28% of all motion components in the employed film material. The tracker frequently fails for very fast motions, motions in regions with low contrast, and complex scenes of water such as in Figure 9.11(b) in Section 9.3.3. We exclude the motions that KLT misses from the evaluation and yield a significantly higher detection rate of 83% which shows that the proposed method provides high performance when motion tracking is successful.

The false positive rate is relatively high (22%) due to tracking failures and noise. For example, feature points tend to walk along edges resulting in motion components that are wrong but have a significant velocity magnitude. On the other hand, we have configured the system sensitive to small motion components which makes the system prone to noise.

In addition, we test our approach on selected sequences from high-quality film material (230 shots from the feature film "Run Lola Run") and yield a significantly lower false positive rate (3%). The detection rate (for all motion components) is 72% compared to 60% for the low-quality material.

We further evaluate the number of false negatives (motion components that are not correctly segmented) for each shot of the low-quality material. The distribution of false

	0 FN	1 FN	> 1 FN
All motion components	70%	26%	4%
Only trackable motion components	89%	9%	2%

Table 9.1: Percentage of shots containing no false negative (FN), one FN and more than one FN with consideration of all and only the trackable motions, respectively.

negatives is summarized in Table 9.1. The upper row represents the evaluation of the entire system (including the low feature tracking performance) while the bottom row shows the performance of motion segmentation for the trackable motion components *only* (i.e., rectified by the tracking performance).

The approach successfully segments all trackable motion components in 89% of the shots. One trackable motion component is missed in 9% of the shots and only 2% of the shots contain more than one missed component. The greatest potential for improvements lies in the stage of feature detection and tracking. This can be observed from the performance measures for the entire system (including tracking performance) which are significantly lower.

Finally, we measure the computational efficiency of the entire system. We employ a PC with an Intel Core 2 Quad CPU at 2.4 GHz for the experiments. The slowest part is feature tracking. The employed implementation needs 1.3 seconds for tracking features between a pair of frames, resulting in approximately 23 hours for the entire film [23]. An efficient GPU-based implementation would significantly accelerate this process [195]. The proposed clustering method (including filtering of the motion field and postprocessing) is computationally efficient, requiring 10 seconds per shot in average and 110 minutes for segmenting the entire film which corresponds to approximately two times the duration of the film.

9.3 Query-based Retrieval of Motion Composition

The motion segments obtained by the method described in the previous Sections allow for a compact and expressive description of the motion contained in a shot. The segments describe diverse types of motion like camera motion, motion of single objects and motion of groups of objects. The motion segments are a well-suited basis for the development of novel applications that enable the retrieval of motion from films. In

the context of archive film the retrieval of motion compositions is of special interest because they significantly characterize the style of a film. In this section, we present an approach to automated motion retrieval that supports film scientists and archivists in the search for particular motion compositions.

Different approaches for the automatic analysis and retrieval of motion have been proposed in literature. However, they are hardly suitable for the applications of film scientists for different reasons. Most methods are based on the analysis of object motions only. They first segment and track the objects in a shot [36, 40, 76] and then compute a motion trajectory for each object [50]. Retrieval is then performed by matching the object trajectories with trajectories provided by the user [19]. However, some types of motion which are important for motion compositions cannot be represented by these approaches because they are difficult to segment, such as groups of objects (e.g., people, cars), motion of water (e.g., rivers) and smoke. Other approaches skip object segmentation and focus on the retrieval of camera motions only [10, 81]. Thereby, the user coarsely provides the average intensity and direction of motion in predefined spatial regions of the video. Such quantitative representations of motion are too coarse and inaccurate for the retrieval of motion compositions.

We propose a more general approach for motion retrieval that exploits the abstract information extracted by motion segmentation in Section 9.2. First, we define a novel type of query for the description of user-defined motion compositions. The query allows the definition of arbitrary motion compositions in an intuitive way. Based on the query, a tolerant matching scheme extracts those shots from a film which have a similar motion composition. The method enables searching for typical camera motions, object motions and characteristic motion directions.

9.3.1 Query Design

The design of the query is a crucial factor since it defines in which way the user has to specify the motion content of interest. Different types of queries exist in retrieval, such as textual queries, example-based queries and sketch-based queries.

A number of systems incorporating motion for retrieval have been introduced, for example *VideoQ* [40], *MovEase* [6], *Picturesque* [50] and the system in [54]. These systems require the user to define trajectories that represent the motion of the objects contained in the sequence of interest. Trajectory-based motion queries can become very

complex with an increasing number of available degrees of freedom [6]. Each moving object may be described for example by its trajectory together with its projected size, direction, speed and acceleration at each time instant. The definition of such a query can become counter-intuitive and time consuming for complex motion compositions. Tolerant retrieval is difficult due to the large amount of detail contained in the motion description. Furthermore, the definition of such a motion query requires detailed knowledge about the sequences of interest which reduces the exploratory capabilities of the corresponding systems.

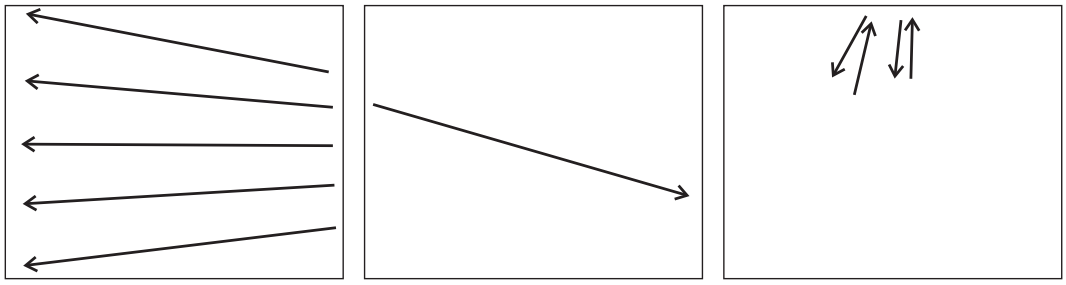
We envision a query that is much easier (and faster) to define and that allows the user to integrate more variability into the motion description. We perform experiments with six test users who are all film experts. They are asked to sketch the motion content of selected shots on a piece of paper. The resulting sketches reveal that the most important information for the participants is the direction of the motion followed by its spatial location. Velocity and acceleration are neglected by most users. Even the size of the moving objects plays a secondary role. Furthermore, most test persons sketch motions of groups of objects as one coherent motion.

Based on these findings, we develop a query that enables sketching the motions of interest as vectors in a sketch-pad window. The absolute position and the length of a vector coarsely specify the region where the corresponding motion occurs (see Figure 9.10 for example queries). For simplicity, we do not consider velocity magnitude and acceleration in the queries.

A query can describe single object motions as well as motions of groups of objects and camera motions. Large moving objects or groups of objects can be specified by drawing several nearby vectors with similar direction. For spatially distributed motion like camera motion, the user simply sketches several arrows with the desired direction(s) distributed over the entire query window[1]. The proposed query allows for expressive motion descriptions and at the same time provides enough freedom for tolerant retrieval.

The numerical description of a query contains the directions of all provided query vectors. Additionally, a region (aligned rectangle or ellipse) around each vector is

[1]Note that for camera motion the direction of the query vectors has to be reversed because the vectors always represent motion relative to the image content. A camera pan to the right for example is represented by several arrows that point to the left because the image content on the screen actually moves to the left.



(a) camera pan (b) diagonal motion (c) hammering

Figure 9.10: Three examples of motion queries. The first query represents a camera pan to the right or the movement of a large object to the left. The second query represents the diagonal motion of an object or a group of objects from left to right. The third example represents (possibly repeated) up and down movements which may originate e.g. from hammering.

extracted that represents the area covered by the corresponding motion. These two parameters (per query vector) are sufficient to represent motion compositions.

9.3.2 Query Matching

The retrieval of motion compositions requires the matching of the query with the previously computed motion segments. We extract representative information from the motion segments that is structurally similar to the parameters derived from the query vectors in order to allow a comparison of the descriptions. For each motion segment two parameters are computed: a representative motion direction and the spatial region covered by the segment.

The segments obtained from motion segmentation comprise a sparse set of fragmented trajectories. The extraction of representative information for such a segment is difficult, since the trajectories have different begin and end times and are spatially distributed. We extract the *median direction* of all trajectories in the segment which is a robust estimate for the dominant motion direction of the segment. The second extracted parameter is the spatial region covered by a segment. Therefore, we compute the polygon (convex hull) that encompasses all trajectories of the segment.

In the next step, a match between the query and the motion segments is established based on the extracted directional and spatial information. Optionally, temporal parameters (start time and duration of a motion) can easily be incorporated as additional

constraints. During matching, all motion segments of a shot are compared with the query. A query Q is defined as a set of N query vectors q_i with directions θ_i and assigned regions $\mathcal{R}_i$. A shot S contains M motion segments m_j with representative (median) directions φ_j and regions (surrounding polygons) $\mathcal{M}_j$. Matching between a query and the motion segments of a shot is performed in three stages.

In the first stage a matching score $s_{i,j}$ between each query vector q_i and each motion segment m_j of the shot is computed as:

$$s_{i,j} = \frac{\theta_i \cdot \varphi_j}{\|\theta_i\| \cdot \|\varphi_j\|} \cdot \frac{|\mathcal{R}_i \cap \mathcal{M}_j|}{|\mathcal{R}_i|} \cdot \left(1 - \frac{|\mathcal{M}_j \setminus \mathcal{R}_i|}{|\mathcal{M}_j|}\right) . \tag{9.8}$$

The first term is the Cosine similarity of the query vector's direction θ_i and the motion segment's median direction φ_j. The second term is the portion of intersection between the region covered by the query vector $\mathcal{R}_i$ and the motion segment's region $\mathcal{M}_j$. The spatial intersection is negatively weighted (penalized) by the area of the motion segment *not* covered by the query vector ($|\mathcal{M}_j \setminus \mathcal{R}_i|$). Matching each query vector with each motion segment yields a set of scores $s_{i,1\ldots M}$ for each query vector q_i. The scores are in the range $[-1; 1]$.

In the second stage, all positive scores for a query vector q_i are summed up:

$$s_i = \sum_{j=1}^{M} \max(s_{i,j}, 0) . \tag{9.9}$$

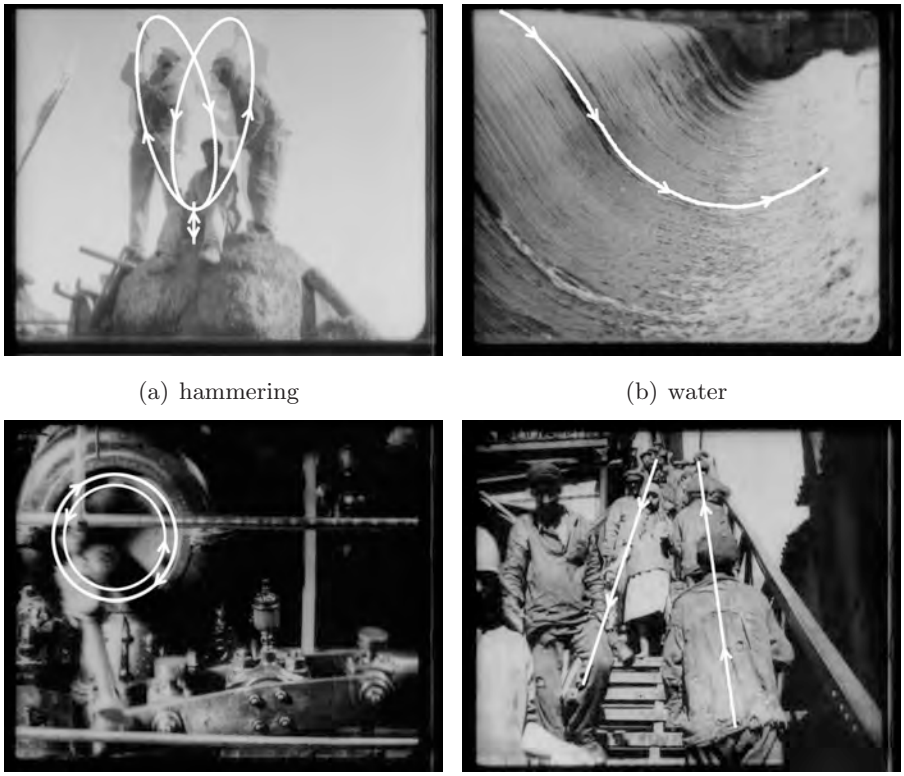
This allows one query vector to score on several motion segments. Negative scores obtained due to negative Cosine similarity are ignored. That means that we exclude scores between query vectors and motion segments when their directions have an angle larger than 90 degrees.

Finally, in the third stage an overall score s is obtained by taking the sum of the scores s_i of all query vectors:

$$s = \sum_{i=1}^{N} s_i . \tag{9.10}$$

The shots with the highest overall scores for the query are returned to the user:

The matching procedure is tolerant since it does not require *all* query vectors to match with a retrieved shot. Additionally, a query vector accumulates scores from all matching motion segments which makes matching more robust under noisy conditions, where motions are sometimes split into multiple motion segments. The amount of



(a) hammering (b) water

(c) rotation (d) group of people

Figure 9.11: Typical motion compositions.

desired tolerance depends on the application. Matching can be performed more strictly by introducing penalties for non- and poorly-matched query vectors.

9.3.3 Experimental Results

Similarly to the experiments on motion segmentation in Section 9.2.3, we select archive films that frequently contain complex motion compositions for the evaluation of the proposed approach. One example is again the film "The Eleventh Year" which contains frequent camera travelings, contrapuntal movements and work activities. Some examples of motion compositions are illustrated in Figure 9.11.

Qualitative results

We evaluate the proposed method with queries representing camera motions, object motions, motions of groups of objects and combinations thereof. Experiments show

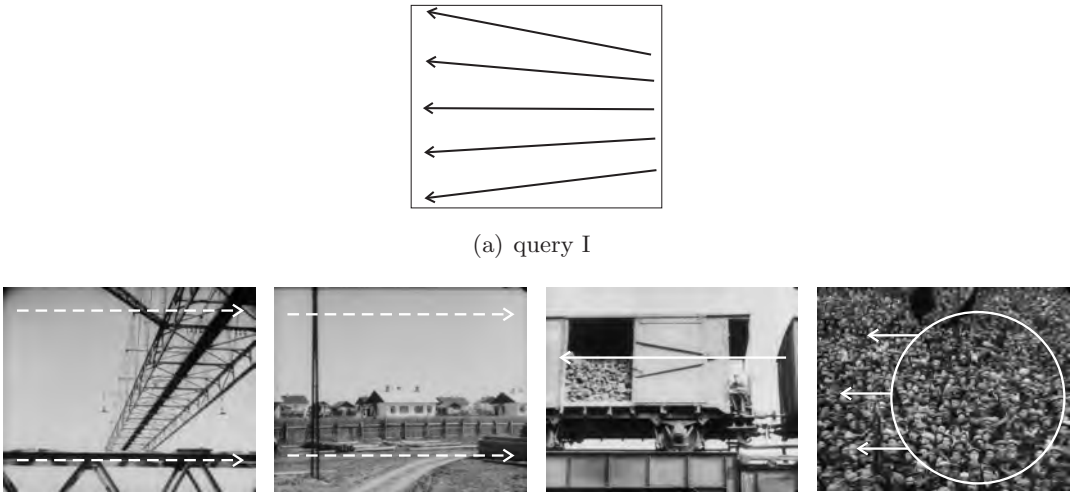
that the method performs well for object motions. However, for the retrieval of spatially distributed motion (e.g. camera motions) which is represented by several query vectors the ranking of the retrieved shots is suboptimal. That means that the most relevant shots are indeed retrieved but do not yield the highest scores. This can be improved by incorporating the total number of scoring query vectors into the matching function from Equation (9.8). We weight the score s with the number of scoring query vectors P_q: $s_{total} = P_q \cdot s$. This increases the score of matches with a larger number of scoring query vectors and generates rankings that better correspond with the user's expectations.

We present the results of three heterogeneous queries in Figures 9.12-9.14. For each query, the figure shows keyframes of four of the returned shots. The first query describes large-scale motion, such as a group of objects moving from right to left or a camera motion (pan or traveling) to the right[1]. The best match (Figure 9.12(b)) is a tracking shot where the camera passes through under a bridge from left to right. Similarly, the result in Figure 9.12(c) is a tracking shot where the camera is mounted orthogonally to the direction of travel and captures the passing by environment. The third shot in Figure 9.12(d) shows a train passing by (from right to left) captured from a static camera. A remarkable result is the fourth shot (Figure 9.12(e)). It captures a large crowd of people which disorderly pushes and shoves to the left.

The second query in Figure 9.13(a) contains a diagonal motion from left to right which is typical for the filmmaker of the analyzed films. The best match is a tracking shot, where the camera is mounted approximately 45° to the direction of travel (Figure 9.13(b)). This yields a dominant motion component in the query vector's direction resulting in a high matching score. The remaining shots show groups of objects (vehicles, people and horses) moving diagonally towards the static camera (Figures 9.13(c) - 9.13(e)). The motion directions in the returned shots slightly deviate from the query vector's direction. This demonstrates the tolerance of the matching scheme.

The proposed method is able to retrieve even more complex motion compositions. The third query represents a combination of opposed (possibly cyclic) vertical motions in the upper region of the frame. The best match is a shot of a trumpeter who moves his trumpet up and down while playing (Figure 9.14(b)). The second retrieved shot shows several workers walking up- and down a stairway (Figure 9.14(c)). The shot in

[1]Note that this query may retrieve camera motions as well as object motions. We do not intend to distinguish between camera and object motion.



(a) query I

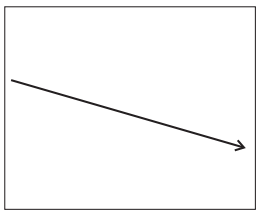
(b) (c) (d) (e)

Figure 9.12: First example query. The query together with keyframes of four top-ranked result shots are shown. Dashed arrows in the keyframes mark camera motions and solid arrows are object motions.

Figure 9.14(d) shows factory workers moving up and down their arms while they pull and push a rod. The last shot in Figure 9.14(e) shows three workers who hammer down a pin into a rock.

Quantitative results

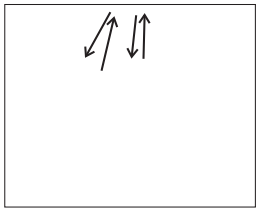
We perform a quantitative evaluation to obtain representative and objective performance measures. For this purpose, we define 17 different motion queries. The queries represent typical motion patterns of camera motions, small and large objects motions, contrapuntal and rhythmical motion compositions. For each query, we assess the 12 top-ranked returned shots as either relevant or not relevant. The portion of relevant shots in this result set gives a performance measure for the accuracy (precision) of the method and can be computed for differently sized result sets (e.g. from 1 to 12). The corresponding measures are termed "prec@1" to "prec@12". These measures enable the evaluation of the obtained retrieval performance as well as the ranking generated by the approach. A good ranking is represented by a high prec@1 (which means that the top-ranked result is most often relevant) and monotonically decreasing precisions for larger result sets (prec@2, prec@3,...). The average precisions (for result set sizes



(a) query II

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

Figure 9.13: Second example query. The query together with keyframes of four top-ranked result shots are shown. Dashed arrows in the keyframes mark camera motions and solid arrows are object motions.



(a) query III

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

Figure 9.14: Third example query. The query together with keyframes of four top-ranked result shots are shown. Solid arrows describe object motions.

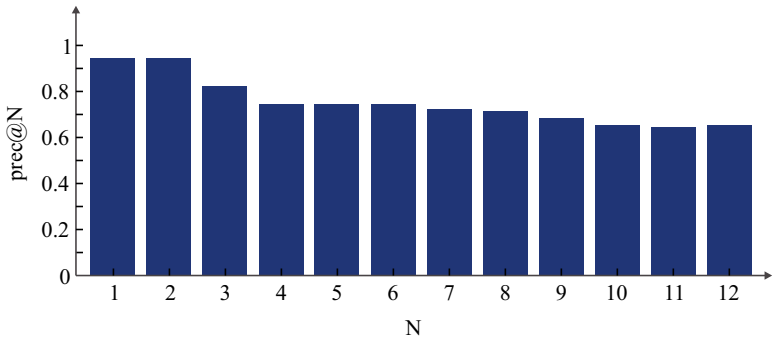


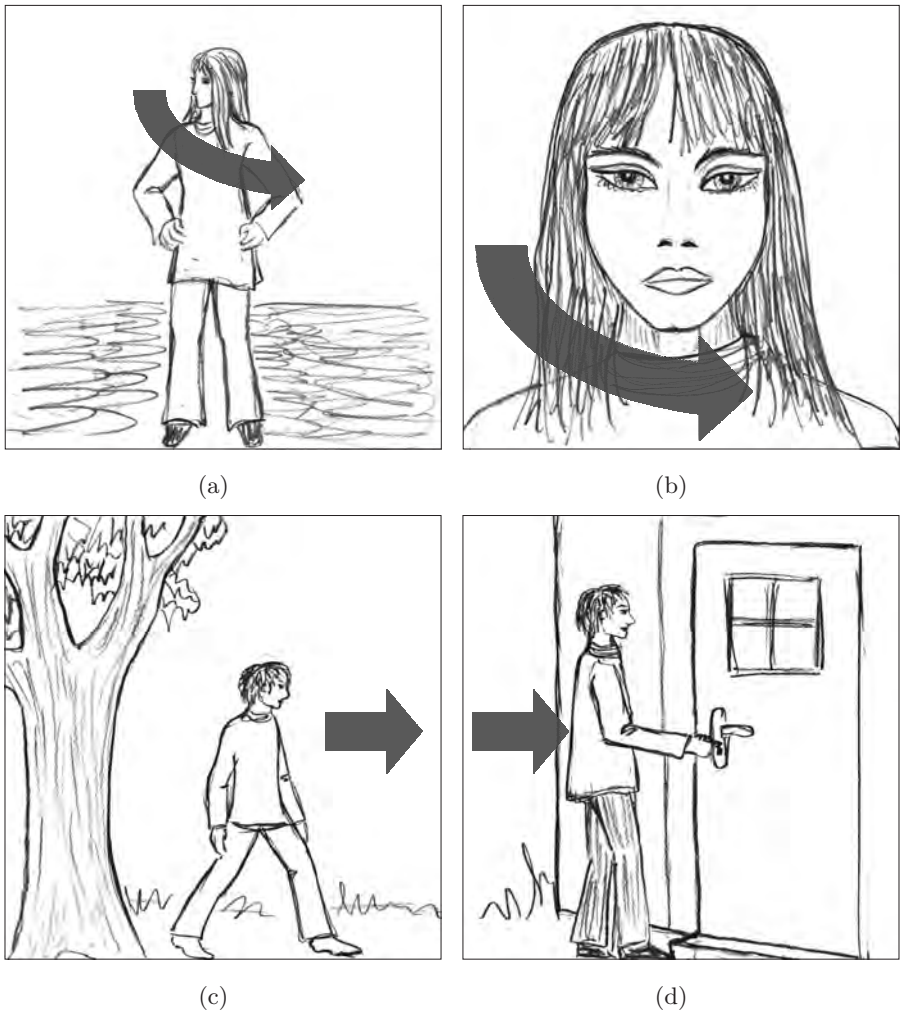
Figure 9.15: Performance of motion composition retrieval: precisions for different result set sizes.

from 1 to 12) over all 17 queries are shown in Figure 9.15. We observe that the precision is generally high and decreases for increasing result set sizes which proves that the ranking is reasonable. The first returned shot is relevant with 94% in average. Among the 12 top ranked shots of the 17 queries in average 65% are relevant to the user.

The evaluation confirms that the generated motion segments adequately represent the motion content of the analyzed film material. The simple and intuitive queries combined with the tolerant matching scheme enable the efficient search for particular motion compositions.

9.4 Query-based Retrieval of Motion Continuity

Continuity editing plays an important role in filmmaking. It "refers to the matching of individual scenic elements from shot to shot so that details and actions, filmed at different times will edit together without error" [20]. Continuity editing assures that consecutive shots in a scene fit seamlessly together and that the conveyed story is presented consistently to the viewer. An important device for achieving continuity is *matching on action* (also cutting on action, cutting on motion). Matching on action aims at keeping the screen direction (the motion direction of objects from the perspective of the camera) between successive shots consistent. Directional continuity is important to avoid confusion for the observer. In scenes presenting a chase for example the motion direction is usually consistent among several shots, e.g. from left-to-right to convey the impression of a continuous action. A single shot that contains right-to-left motion in this scene would confuse the observer's orientation. Two examples of matched actions



(a) (b)

(c) (d)

Figure 9.16: Two examples of matched action.

are shown in Figure 9.16. The first example in Figure 9.16(a) and 9.16(b) shows a character that turns its head from left to right. During this movement the director cuts to a close up of the face. The continued motion between both shots makes the transition between the different shot scales appear seamless. Figures 9.16(c) and 9.16(d) show an example of the entrance-exit pattern [20]. An actor exits the frame at the right side and in the successive shot enters the frame from the left but perhaps at another time and location. This pattern can be utilized to create a continuous transition between different scenes and locations.

In the following sections, we present a method that supports the investigation of a film's motion continuity editing. For this purpose, we extend the method for motion composition retrieval from Section 9.3 by extending the presented query model from Section 9.3.1 and adapting the matching scheme from Section 9.3.2. The resulting method is able to find shots with matching action based on a user-defined query.

9.4.1 Query Design and Matching

An overview of the extended retrieval process is given in Figure 9.17. First, the query window is split vertically into two halves: A and B. In query A the user sketches the motion present at the end of an arbitrary shot and query B contains the continued motion at the beginning of the next shot. For the retrieval of matched motion we define an analysis window (the gray rectangle in Figure 9.17) of a few seconds (2 seconds in the experiments) around each cut. Query A is then matched *only* with the left half of the analysis window (end of shot N) and query B *only* with the right half (beginning of shot N+1). The restriction to the window is necessary to get more accurate results. Matching queries A and B with the entire shots N and N+1 (as in Section 9.3.2) could take motion into account which is temporally not located around the cut and thus is not relevant for continuity. If both queries positively score on the according halves of the analysis window we combine both scores and return shots N and N+1 as a result to the user. All returned results are then ranked according to their combined score.

For the retrieval of matched motions a finer matching scheme is necessary that restricts the comparison to the analysis window only. We adapt the matching scheme from Section 9.3.2 as follows. First, we remove all motion segments that do not coincide with the analysis window. For the remaining motion segments we extract directional and spatial parameters. Since the motion segments may be only partially inside the analysis window, the median direction of the entire segment (as used previously) is not a representative parameter with regard to the analysis window. Instead, we compute the median direction of the segment for *each frame* in the analysis window separately. For an analysis window that covers $2 \cdot D$ frames, this results in D directions $\theta_{i_{1\ldots D}}$ for each segment which allows for a more precise matching. Directional matching between a segment and a query vector is then performed by matching the direction of the query vector with each median direction of the segment. We compute the mean of the Cosine similarities (see first term in Equation (9.11)) between the query vector's direction φ_j

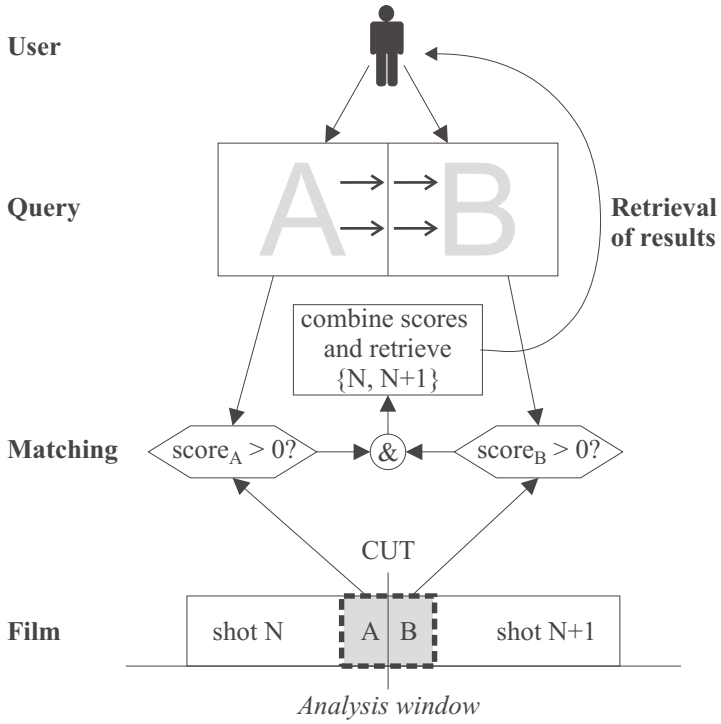


Figure 9.17: Retrieval of matching actions. Each query (A and B) is matched with the corresponding half of the analysis window. In this example the query represents a typical entrance-exit pattern.

and each direction θ_{i_k} of the segment. Spatial matching is performed as in Section 9.3.2. The modified scoring function is defined as:

$$s_{i,j} = \frac{1}{D}\left(\sum_{k=1}^{D} \frac{\theta_{i_k} \cdot \varphi_j}{\|\theta_{i_k}\| \cdot \|\varphi_j\|}\right) \cdot \frac{|\mathcal{R}_i \cap \mathcal{M}_j|}{|\mathcal{R}_i|} \cdot \left(1 - \frac{|\mathcal{M}_j \setminus \mathcal{R}_i|}{|\mathcal{M}_j|}\right). \tag{9.11}$$

We compute overall scores for both halves of the analysis window s_A and s_B as in Section 9.3.2 by summing up the scores over all motion segments and query vectors. Finally, we combine them by taking their product. This measure yields a high overall score only when *both* scores are high which is an important prerequisite in this retrieval scenario. Additionally, the scores are weighted by the portion of scoring query vectors P_q from queries A and B and by the portion of scoring motion segments P_s in the analysis window:

$$s_{total} = s_A \cdot s_B \cdot P_q \cdot P_s. \tag{9.12}$$

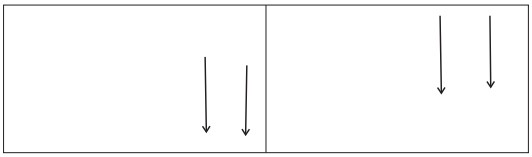
The weighting increases the score where the query vectors correspond particularly well with the actual motion content. This weighting improves the ranking of the retrieved sequences.

9.4.2 Experimental Results

We evaluate the performance of the proposed method with the feature film "Run Lola Run" by Tom Tykwer from 1998. The film is a thriller that makes extensive use of matching on action. There are numerous scenes where motion is consistently carried across several cuts such as chasing scenes and journeys. Many scenes for example show the leading character Lola running through the streets from different viewpoints. They are all connected by matching action to create the impression of a continuous journey. Another frequent pattern are subsequent dolly forward shots joined with matching action which show Lola's view during running. The film further contains characteristic sequences of shots with discontinuous motion. The director connects contrapuntal motions over consecutive shots, for example by alternating dolly forward and backward movements. The film is composed of three episodes that show three different versions of the same story. Consequently, for many scenes there exist three different versions with similar motions but varying content. This makes the material well-suited for the evaluation of the proposed approach.

Prior to the evaluation, film experts manually searched for matching actions in the film and annotated them. We evaluate the retrieval performance for matched actions with different queries. Figures 9.18-9.20 show results for three example queries. The first one in Figure 9.18(a) describes a local motion in the right half of the frame directed downwards that is carried across a cut. This is a variation of the entrance-exit pattern. The first returned result shows Lola's friend Manni in a phone booth talking to Lola. At the end of the shot Manni sinks down in resignation and his head leaves the frame at the bottom. The following shot continues this motion from another camera angle and shows Manni's head moving in from the top of the frame.

The second query (see Figure 9.19(a)) describes a spatially more distributed motion with a screen direction pointing downwards. This can either correspond to downwards motion or motion towards the camera. The query matches well with a scene of shots showing an ambulance approaching the camera that just crashed into a glass plate. While the ambulance comes closer to the camera the director cuts to a wide angle shot



(a) query I

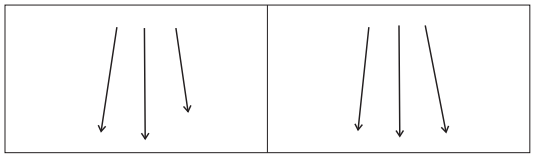
(b) result I

Figure 9.18: Example I: a downwards motion that is continued over a shot cut from "Run Lola Run". Credit: Stadtkino Filmverleih.

that continues the motion (until the ambulance stops) and shows what has happened in the surrounding of the ambulance after the accident (see Figure 9.19(b)). This example demonstrates how matching action can be applied to create a seamless transition between two different scales of a shot.

The third example query in Figure 9.20(a) represents the motion pattern produced by a zoom in or dolly forward motion. The method returns several pairs of shots with a continued dolly forward motion. One example is shown in Figure 9.20(b). The first shot is a medium shot of Manni. While the camera slowly moves towards Manni the director cuts to a medium shot of Lola where the camera continues its movement at the same speed towards Lola. This transition directs the attention towards the two main characters and increases the tension in the scene.

In most cases the returned results match well with the expert annotations. However, there are also results that do not match the underlying query at the first glance. An example are shots captured with a shaky camera (steadicam) which often appear in chasing scenes. In some cases the shaking of the camera produces a pattern of continued motion between two consecutive shots. Such sequences of shots usually do not convey the impression of a matched action. Other sources of confusion are background movements that match well with the query (e.g. a car moving in the background). Such results may surprise the user because the background motion is often not perceived consciously by the viewer. Consequently, the viewer would not recognize the matching motion in this case. However, the proposed method detects such matching



(a) query II

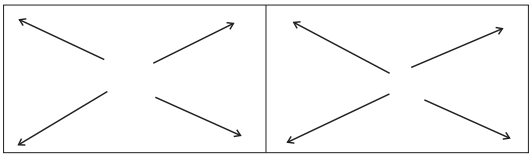
(b) result II

Figure 9.19: Example II: a spatially distributed motion towards the camera continued over a shot cut from "Run Lola Run". Credit: Stadtkino Filmverleih.

background motions since it does not distinguish between foreground and background motion or between "salient" and "non-salient" motion. Although matching background motions may not be recognized by the viewer as examples of matched actions at the first glance, the question arises whether or not the movements are matched on purpose by the director.

We further observe that the proposed method is able to retrieve matching actions that were not recovered by the film experts during annotation. An example is shown in Figure 9.21. It shows an excerpt of cross cut shots between Manni and Lola talking on the phone. The cut between the shots in Figure 9.21 is positioned in a way that the downwards movement of Lola's head is continued seamlessly by the downwards movement of Manni's head. This generates a transition that leads the viewer smoothly from one shot to the next. The matching action lasts for approximately a second which makes it difficult to detect manually. This example shows that the proposed method has the potential to detect patterns of interest that human viewers are likely to overlook. In this manner, the method is able to assist the investigation and exploration of continuity editing in a film.

The analysis of matched motion with *continuous* screen direction is only one possible application scenario. We can for example retrieve sequences of shots with *contrapuntal* motion where shots with different (possibly opposing) motion directions alternate. Such sequences can be employed to create the impression of objects or people moving away from each other. Furthermore, contrapuntal motion is a device for the creation of



(a) query III

(b) result III

Figure 9.20: Example III: a zoom in continued over a shot cut from "Run Lola Run". Credit: Stadtkino Filmverleih.

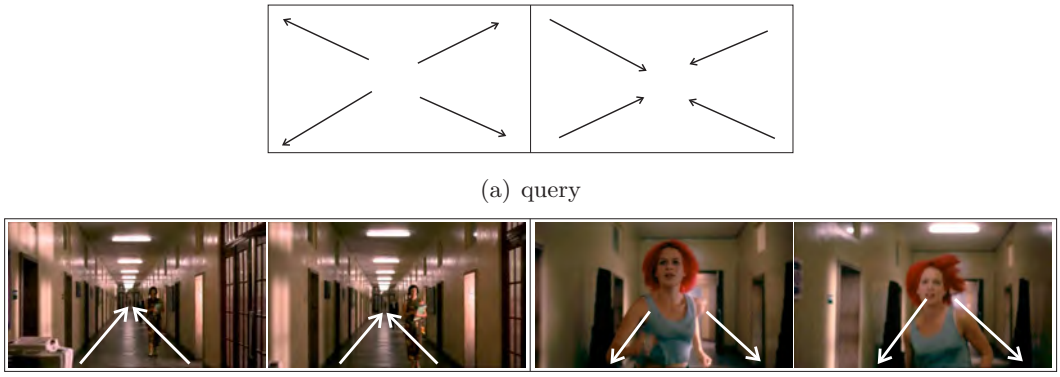
Figure 9.21: A matched action recovered by the proposed method. Credit: Stadtkino Filmverleih.

rhythmic motions. In "Run Lola Run" for example dolly forward movements are frequently followed by dolly backward movements and vice versa. Retrieval results show that this combination often appears in the film in situations where Lola is running (see Figure 9.22 for an example). The first shot of such a combination typically shows the world from Lola's perspective translated in a dolly forward when she is running. In the subsequent shot the camera is positioned in front of Lola and moves backwards showing her running.

Ultimately, we want to point out that the applicability of the method is not limited to the presented examples. The method can be employed for the retrieval of arbitrary combinations of consecutive motions since the definition of the query is up to the user.

9.5 Summary

In this chapter, we have first presented a novel trajectory clustering approach for sparse motion fields. The feature trajectories are highly fragmented and have different tem-



(a) query

(b) result

Figure 9.22: A typical contrapuntal composition of motion that appears repeatedly in "Run Lola Run" together with the according query. Credit: Stadtkino Filmverleih.

poral locations and lengths due to the low quality of the film material. The proposed clustering scheme robustly segments noisy trajectories into meaningful motion components and is computationally efficient. The clustering scheme allows for the flexible selection of trajectory features and similarity measures which makes it well-suited for different types of clusterings and applications. Although the method has been developed for low-quality film material, experiments have shown that it is applicable to diverse video material. The low-quality of the archive film material mainly influences the feature tracking performance. Where motion is trackable, motion segmentation is successful to a high degree.

The extracted motion segments represent an abstract and compact description that is a well-suited basis for motion retrieval. Based on the extracted motion segments, we have developed two applications for motion retrieval. Both applications take simple and intuitive motion queries as input and retrieve sequences with similar motion content by a tolerant matching scheme. The first application retrieves user-specified motion compositions from a film, such as camera motions, dominant motion directions, and combinations of object motions. The second application is an extension of the first one and enables the semi-automatic investigation of motion continuity. An extended query allows the user to specify the motion between two successive shots. The application enables the user to search for continuous motion transitions as well as contrapuntal motion between successive shots.

The proposed query model is easy to understand and does not restrict the user to a predefined vocabulary. Consequently, it does not limit the range of possible search requests and supports not only searching for already known motion patterns but also enables the exploratory search of motion compositions. The developed methods perform well in the investigated retrieval scenarios. In some cases, we even gain new insights about the analyzed films by experimenting with the retrieval system. The presented methods have the potential to assist film analysis and to improve searching movie databases.

Chapter 10

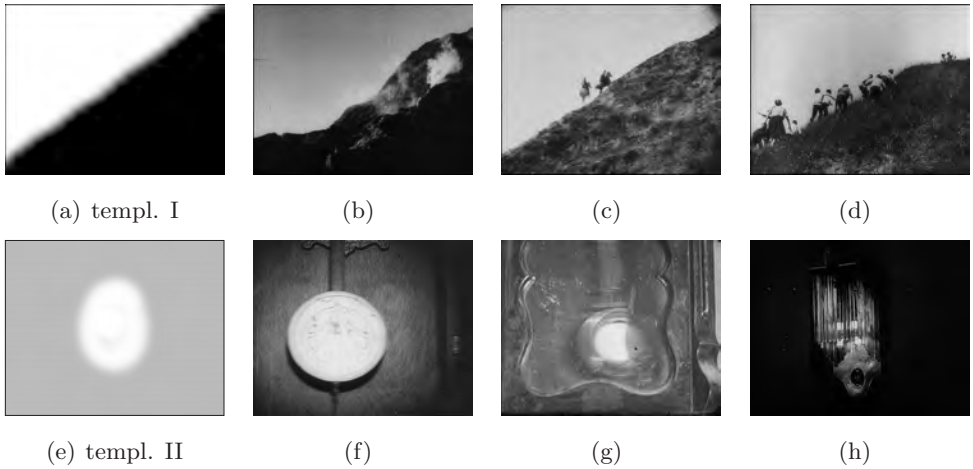
Retrieval of Visual Composition - A User Study

The composition of shots in a film is not solely characterized by motion which has been discussed in the previous chapter but also by *visual composition*. In this chapter, we investigate the retrieval of visual composition in a user study. In Section 10.1 we frame a hypothesis regarding the retrieval of visual compositions and identify research questions. Background information on visual composition is given in Section 10.2. We present the content-based features and similarity measures we employ in the user study in Section 10.3. Additionally, we introduce a novel measure for the expressiveness of content-based features. Section 10.4 describes the experimental setup of the user study, the employed retrieval system and the subjects. We discuss the results of the experiments and answer the research questions in Section 10.5.

10.1 Introduction

The concept of visual composition refers to the spatial arrangement of the visual elements (objects and their shapes) of an image. In painting, the artist arranges the visual elements in a picture to evoke a certain impression. In film, the director arranges the elements in a scene and selects the camera's view [237].

Film experts want to identify recurring visual compositions (see Figure 10.1) because they want to analyze how compositions are used for conveying the message. Currently, there is no accepted method for automated identification of visual composi-



(a) templ. I (b) (c) (d)

(e) templ. II (f) (g) (h)

Figure 10.1: Two composition templates with three frames from different films that share the respective visual composition type.

tions in film. Related work, such as [62], focuses on composition retrieval in news videos which follow much stricter composition rules than film. This is the reason why related work is not applicable to films. It is still unclear whether or not visual compositions, *as understood by film experts*, can be represented and retrieved by low-level content-based features. This is especially true for compositions that are strongly influenced by the semantics of the depicted figures and objects. In this chapter we investigate the applicability of well-understood content-based retrieval methods in the novel domain of visual composition retrieval. For this purpose, we assemble a real world data set with the help of film experts in order to measure the retrieval performance.

We design a system for retrieval of visual composition in film and perform a user study to test and answer the following hypothesis and research questions:

Hypothesis 1 *Low-level features are able to represent visual compositions.*

We pair combinations of features and single features with different proximity measures and let humans evaluate the retrieval results. These relevance judgments serve as a metric for a feature's ability to represent visual compositions. Features that capture visual compositions well will produce better average relevance judgments. Additionally to the hypothesis, we investigate three research questions.

- RQ 1: Which content-based features perform best?

- RQ 2: Which proximity measure performs better?
- RQ 3: Do film experts judge the same retrieval results differently than others?

We derive the third research question from the assumption that subjects with expertise in film studies better recognize the presence of compositions than subjects without this expertise. Consequently, we expect a better assessment of the retrieval results by film experts than by the control group.

10.2 Background on Visual Composition

Visual composition is intrinsic to visual arts and dates back to prehistoric cave paintings and ancient Egyptian papyri. We can safely assume that artists always passed on knowledge regarding visual composition from one generation to the next. In former times this dissemination of knowledge was performed mostly verbally and through imitation. Later, scholars started to put this knowledge down in writing. Today, readily available scholarly work on visual composition dates back to the 19^{th} century [35]. Since then, more and more scientific effort has been directed towards understanding the processes that are used for composition [11, 168]. We see composition as the result of two concurrent processes. First, the adherence to certain principles and, second, the application of formal elements.

Formal elements among others include lines, shapes, textures and colors of depicted objects and surface areas. Formal elements are either purposely embedded into the image or they become apparent at a later time. For example see Figure 10.2 which shows a frame from an archive film, where a group of children is marching through high grass. Many beholders perceive a line, formed by the children's heads, although there is no line-shaped real word object in the image. The line develops in the beholder's mind.

Principles of composition include hard to grasp concepts like the dominant idea of the image as well as more tangible concepts like the gradation of lighting, the balance of the depicted elements, and the use of space. Leonardo da Vinci's *The Last Supper* is a textbook example for the principle of space, see Figure 10.3. Da Vinci depicts Jesus and the apostles in a large hall at the table eating supper. The image is a snapshot in time at the moment when Jesus says he knows one of them was going to betray

Figure 10.2: A keyframe from our database, where a group of children marches through high grass. The beholder perceives a line formed by the children's heads, although there is no line-shaped real world object in the image.

him. The artist organizes the apostles (twelve figures) into four groups of three. He positions two groups to the left and two groups to the right of the central figure (Jesus) thereby balancing the number of depicted figures relative to the center of the image. A notable compositional aspect of the image is the use of space. Da Vinci embeds the figures at the table in a large three dimensional hall to create the illusion of depth. The hall's spatial extent is generated through the use of central perspective. The coffered ceiling as well as the tapestries (dark squares at the walls) emphasize the perception of space by introducing (perspective) lines and texture. Da Vinci positions the central figure of the image at the vanishing point of the scene where all perspective lines meet. This positioning draws the beholders look to the semantic and syntactic center of the image. Additionally, da Vinci uses light, employing the middle of the three windows in the hall's back wall to create a halo for the central figure. Light is also used in the depiction of the semantically important figure of Judas (the apostle who betrays Jesus). Da Vinci draws Judas to be in the shade, this way the apostle is darker than the others.

The two examples in this section should hint at the often times very deliberate processes involved in the composition of an image and give the reader an intuitive understanding of what visual composition refers to. A complete discussion of all principles of visual composition is out of scope of this work. We refer the interested reader to available literature, e.g. [11, 35, 168].

Figure 10.3: Leonardo da Vinci's *The Last Supper*. An example for the composition principles of space, balance and gradation of lighting [223].

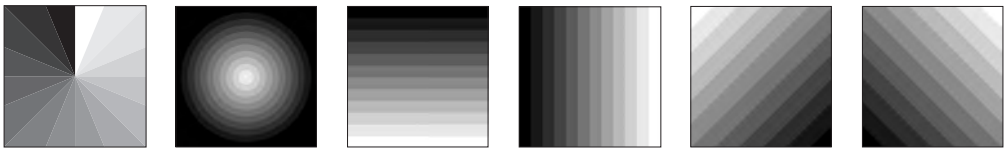
10.3 Evaluated Techniques

10.3.1 Content-based Features

The formal elements and principles of composition can be divided into two groups, the tangible and the intangible ones. We focus on the *tangible* elements and principles. We expect that they can be captured with content-based features and thus are relevant for access to visual databases.

First, we select edge histogram, region shape, and homogeneous texture which are defined in the MPEG-7 standard for multimedia content description [101].

The MPEG-7 edge histogram (EH) summarizes the spatial distribution of edges. The edge histogram captures the general distribution of objects inside the image and may serve as an indicator for balance. The MPEG-7 region shape feature (RS) describes the image's content in terms of coefficients of the Angular Radial Transform (ART) which are invariant towards rotation and robust to scaling [179]. Region shape is linked



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

Figure 10.4: Masks defining the regions that are employed for the description of color and intensity distribution in the KANSEI color and intensity feature. Note that the shading only illustrates the spatial arrangement of the regions.

to shapes, in terms of formal elements of composition. The MPEG-7 homogeneous texture (HT) feature captures the energy and energy deviation of 30 Gabor wavelet frequency channels. The name of homogeneous texture implies its relation to the formal elements of composition, it captures texture information.

Second, we employ the so called KANSEI features by Kobayashi et al. [111]. They propose the joint application of both a shape feature (KANSEI shape) and a color feature. KANSEI shape (KS) is influenced by several formal elements and principles of composition. It reflects gradation in lighting, balance and shape at the same time. The color feature is based on four composition templates. Each composition template defines twelve regions with a specific spatial arrangement, namely radiation-like, circular, horizontal, and vertical (see Figures 10.4(a) to 10.4(d)). The input image is divided into regions according to the composition template. For each region the average color is computed. These averages are the feature components.

Adaptations to the color feature become necessary because we employ frames from black and white films in this user study. First, we reduce the computation of the average color to one color channel, equaling the computation of the average intensity. Second, we discard the radiation-like mask 10.4(a) proposed in [111] in favor of two diagonal ones shown in Figures 10.4(e) and 10.4(f). This modification is based on recommendations of film experts. We name the modified feature KANSEI intensity.

In addition to the *single* content-based features (see Table 10.1), we evaluate three feature *combinations* (summarized in Table 10.2) and a random feature. We obtain the feature combinations through concatenation of the single features' components. The random feature (RM) has 5 components with uniformly distributed pseudo-random values. The random feature defines a lower-bound of retrieval performance which we

Name	Dim.	Type	Abbr.
MPEG-7 edge histogram	80	local	EH
MPEG-7 homogeneous texture	62	global	HT
MPEG-7 region shape	35	global	RS
KANSEI intensity	60	local	KI
KANSEI shape	64	local	KS
Random	5	-	RM

Table 10.1: Features used in the experiments with their dimension, their spatial layout, and their abbreviations used in this chapter.

Combination	Features	Abbr.
KANSEI features	<KI,KS>	KSI
MPEG-7 features	<EH,HT,RS>	MP7
KANSEI and MPEG-7	<EH,HT,RS,KI,KS>	ALL

Table 10.2: Feature combinations employed in the experiments and their abbreviations used in this chapter.

use to compare the other features with. All features should perform better. This is especially true for feature combinations. Feature combinations could improve retrieval results because they capture more formal elements and principles of visual composition than single features.

10.3.2 Proximity Measures

We consider a feature to be capable of representing visual compositions if the users assess the retrieval results obtained with this feature to be relevant. We acquire retrieval results through similarity retrieval using Salton's vector space model [185]. In order to preserve a certain objectivity we employ one similarity measure and one distance measure. We employ Cosine similarity and the Euclidean distance because they are two well-understood representatives of the respective groups of proximity measures [58].

10.3.3 Statistical Methods

We employ factorial analysis of variance [64] to identify significant differences in the means of the relevance judgments to test the hypothesis and to answer the research

questions. Factorial analysis of variance (ANOVA) is a standard method employed in user studies. Significance tests with ANOVA allow for more objective statements than descriptive methods commonly used in information retrieval. ANOVA enables the evaluation of statistical properties of the investigated factors. In our user study the factors are the content-based features, the proximity measures, the composition templates, and the subjects' field of expertise.

Independently of the relevance judgments, we analyze the expressiveness and data quality of the content-based features. For this purpose we introduce the Weighted Average Loading Indicator (WALDI), a measure for the expressiveness of a feature based on Principal Component Analysis (PCA). In the following, we first present the computation of the PCA since it is the foundation for the novel measure and then derive the Weighted Average Loading Indicator. The PCA is a linear transform that takes a set of possible correlated variables (the feature components in our case) as input and transforms it into a set of decorrelated variables (the principal components) as output [161]. The principal components represent a basis that gives us a common coordinate frame for the comparison of different features' information content. By definition the first principal component describes the direction in which the data have highest variability, the second principal component is orthogonal to the first direction and has the second highest variability, etc.

We arrive at the principal components as follows. The data set is given as a matrix X of n d-dimensional vectors $\mathbf{x}_i \in \mathbb{R}^d$, $i = 1, \ldots, n$ with $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$. Consequently, X is a $n \times d$ matrix with n columns and d rows where the columns represent the feature vectors in our case. First, the mean vector $\mathbf{m} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$ with $\mathbf{m} \in \mathbb{R}^d$ is computed and subtracted from each vector $\mathbf{x}_i$ in X, resulting in a matrix $\overline{X}$ with zero mean. Next, the covariance matrix Σ_X is computed as $\Sigma_X = \frac{1}{n-1}\overline{X}\,\overline{X}^\top$. The covariance matrix is a symmetric, positive definite, $d \times d$ matrix, whose diagonal terms are the variances of the feature components and whose off-diagonal terms are the covariances between the feature components. For decorrelated (independent) variables the covariance matrix (Λ) has non-zero values only in the diagonal terms:

$$\Lambda = \begin{pmatrix} \sigma_1^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_d^2 \end{pmatrix}$$

where d is the number of feature components and σ_i^2 with $i = 1, \ldots, d$ are the variances of the feature components. In order to decorrelate the feature components we need to transform the covariance matrix Σ_X into the form of Λ. Therefore, we need to identify the matrix Γ that diagonalizes Σ_X such that

$$\Lambda = \Gamma^\top \Sigma_X \Gamma. \tag{10.1}$$

Equation (10.1) has a solution where Λ is a diagonal matrix of the ordered eigenvalues of Σ_X and Γ is an orthonormal matrix of the corresponding eigenvectors (principal components) of Σ_X [33]. We obtain Λ and Γ by finding the eigenvalues and eigenvectors of Σ_X [33]. After we order the eigenvalues, Λ and Γ are of the form:

$$\Lambda = \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_d \end{pmatrix}, \Gamma = \begin{pmatrix} e_{11} & \ldots & e_{1d} \\ \vdots & \ddots & \vdots \\ e_{d1} & \ldots & e_{dd} \end{pmatrix},$$

where λ_i are the ordered eigenvalues ($\lambda_1 > \lambda_2 > \ldots > \lambda_d$) of Σ_X, and e_{ji} the j-th component of the i-th eigenvector ($i, j = 1, \ldots, d$). From a statistical point of view, the λ_i are the variances of the transformed feature components and the corresponding eigenvectors $\mathbf{e}_i$ point in the direction of this variability. From the ordering of the λ_i it follows, that the first principal component points in the direction of largest variance, the second principal component points in the direction of second largest variance and so forth. In statistics, Γ is also known as the *factor loading matrix*. The factor loading matrix represents the original features' *loading* (influence) on the principal components. The loading's codomain is $[-1, 1]$, high absolute loadings indicate high influence and vice versa. One way to summarize the amount of variance captured by a feature is the Weighted Average Loading Indicator (WALDI). We compute the WALDI by weighting the eigenvector components' absolute values with the corresponding amount of explained variance and taking the sum over all principal components. We weight the vector components' absolute values, because Γ is orthonormal, i.e. all eigenvectors are of unit length and give only the direction of the variance. Using the eigenvalues λ_i and eigenvector components e_{ji} from above, the WALDI for the j-th feature component $j = 1, \ldots, d$ can be expressed as:

$$WALDI_j = \sum_{i=1}^{d} \hat{\sigma}_i \cdot |e_{ji}|, \text{ with } \hat{\sigma}_i = \frac{\lambda_i \cdot 100}{\sum_{c=1}^{d} \lambda_c}. \tag{10.2}$$

The WALDI summarizes the feature components' influence on the variability in the data. Feature components that describe much of the variability in the data obtain high scores while components that describe little variability obtain small WALDI scores. The WALDI may be employed in unsupervised feature selection for selecting expressive features, i.e. features that represent large amounts of variance.

10.4 User Study

We conduct a user study to evaluate the applicability of low-level features for the retrieval of visual compositions in a real world scenario. We select 30 users for the study, 15 film experts (either film archivists or film scientists) as the test group and 15 computer scientists as a reference group. The reference group consists of computer scientists, because of two reasons. The first reason is that computer scientists frequently (mostly due to availability) serve as subjects in user studies concerned with information retrieval. The second reason is that the inclusion of computer scientists allows for a comparison of the two involved mindsets, on one side computer scientists as the creators of retrieval systems and novices regarding visual composition and on the other side film experts as specialists for visual composition and the real users of such a retrieval system.

We implement a system that takes user-defined sketches of visual compositions and example images as input and retrieves images similar to the sketch based on the features and proximity measures from Section 10.3. The system is able to build arbitrary feature combinations and to pair them with different proximity measures. Furthermore, the system allows the user to assess each retrieval result (see Figure 10.5).

The user study is performed with two sets of queries. The first set contains four predefined (common) query sketches which represent compositions typically sought after by film experts. These query sketches (see Figures 10.1(a), 10.1(e), 10.8(a), and 10.8(e)) were suggested by film experts prior to the study and later generated using a graphics tablet and a pressure sensitive brush. The common query sketches enable an objective comparison of two different user groups. The second set of query sketches is defined by the users themselves during the study. This set of query sketches enables the evaluation of the users' subjective satisfaction. The users first assess the retrieval performance regarding the four common query sketches and then draw and assess four individual query sketches.

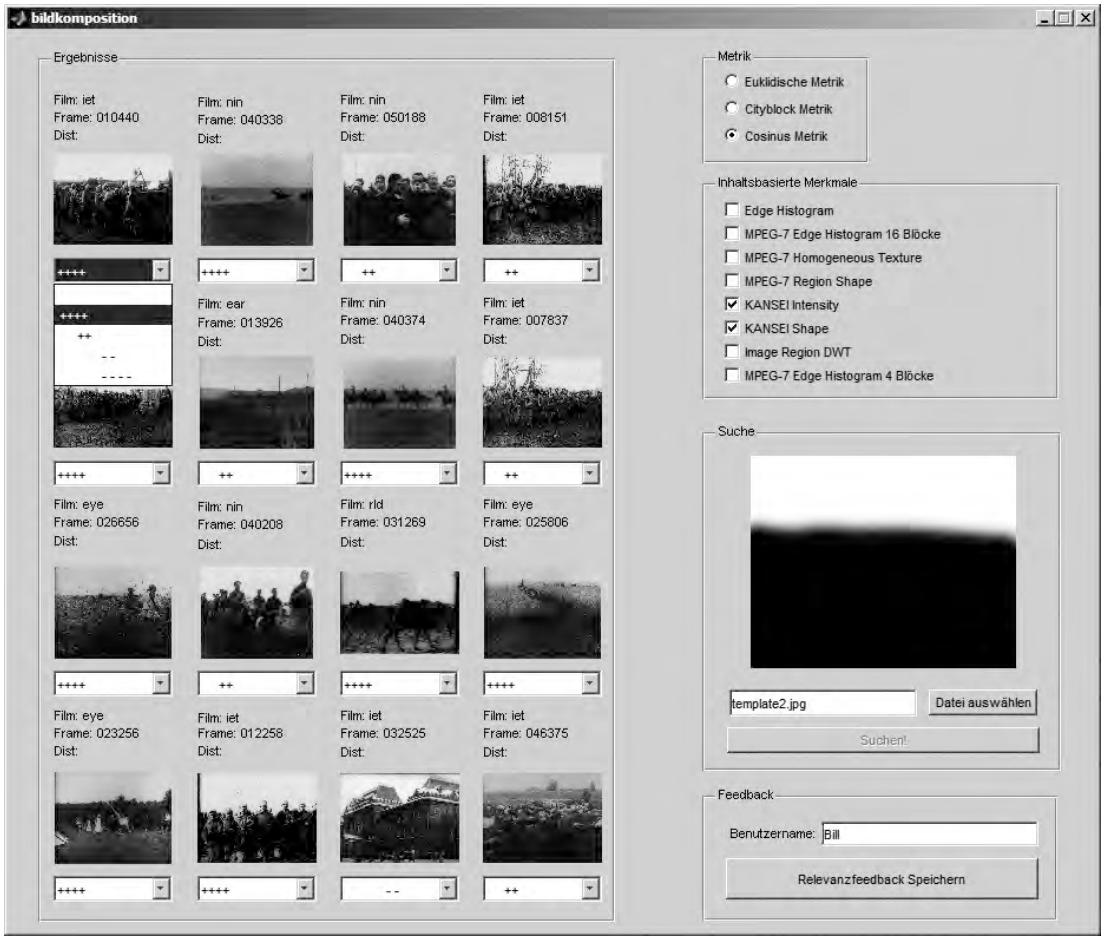
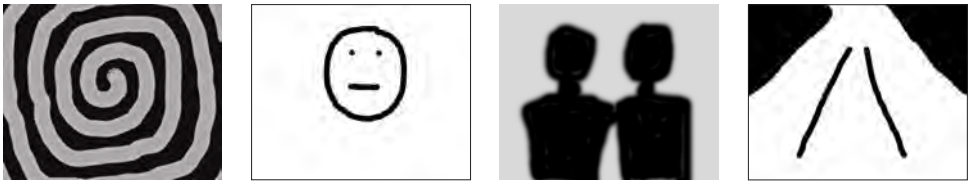


Figure 10.5: The GUI of the retrieval system used in the experiments. The results are presented left to the query sketch. For each result the user can assess its relevance by choosing an entry in the corresponding drop-down menu.

We observe that the individual query sketches (see Figure 10.6) differ from the pre-defined ones in abstractness and the semantic content. Some individual query sketches are entirely abstract, e.g. a spiral, while others are much more semantic, e.g. a schematic face. The performance of queries based on the semantics of sketches will probably suffer from the system's inability to process the semantics presented in the query. In the case of the abstract query images the retrieval performance depends on the frequency of such images in the data set. Note that the system supports user-generated query sketches as well as the use of existing images from known films, the web, etc. For the user study we employ sketches to reduce bias. For example, if existing



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

Figure 10.6: Individual query sketches generated by the users in the study.

images are used for the study, film experts could expect specific frames to be returned regardless of whether these frames are part of the data set or not.

Retrieval is performed on a data set that contains 6690 keyframes from six black and white archive films. The films are formalistic films which make frequent use of visual compositions. We select keyframes from all shots (including the ones without a distinguishable composition) in order to enable an objective evaluation of the employed techniques creating a real world scenario.

We implement a system that takes user-defined sketches of visual compositions as input and retrieves images similar to the sketch based on the features and proximity measures from Section 10.3. For each query sketch, we perform retrieval with the six content-based features listed in Table 10.1 and with the three feature combinations listed in Table 10.2. Each feature and feature combination is paired with both proximity measures (L2-norm and the Cosine similarity). This results in (6 *single features* + 3 *feature combinations*) $*$ 2 *metrics* $=$ 18 different system configurations that are evaluated in the study. Each of the 18 result sets consists of the 16 best matches found in the data set and is assessed separately by each participant. We do not evaluate all possible system configurations to limit the duration of the study for each participant to an acceptable extent. Users spend 90 minutes to four hours to complete all assessments.

Prior to the assessment, we instructed the users to rate the visual similarity of the retrieved matches. All users were informed about the origin of the employed keyframes. Users not familiar with the term *visual composition* were briefed that the term refers to the spatial placement of visual elements inside an image.

	EH	HT	RS	KI	KS
WALDI	35%	25%	21%	50%	100%

Table 10.3: The information content represented by each feature measured with the WALDI technique relative to the best-scoring feature KANSEI shape.

10.5 Experimental Results

10.5.1 Data Quality of Features

An "ideal" feature has decorrelated components and a high score in regard of information content. Feature combinations should exhibit similar properties. Additionally, any two features in a combination should have low inter-feature correlations.

High information content is a necessary but not sufficient property of a good content-based feature. We analyze the features' expressiveness for the image data employed in this investigation. The analysis results are summarized in Table 10.3. We observe that KANSEI shape scores highest followed by KANSEI intensity. This means, they explain large amounts of variance contained in the feature data. The MPEG-7 features consistently have lower scores than the KANSEI features. Their expressiveness is limited in the context of the underlying image data.

In addition to the information content, we investigate intra-feature and the inter-feature correlations. Intra-feature correlations refer to the redundancies between the components of one single feature, while inter-feature correlations refer to the redundancies between components of two or more features. We compute Pearson's correlation coefficient between any two feature components and take its absolute value in order to obtain the correlation matrix depicted in Figure 10.7.

Ideally, the entire matrix would be dark (correlation of zero) except for the main diagonal which should be white (correlation of one). This would indicate that every feature component (and thus every feature) captures specific information that is not captured by any of the other components (and features).

On the intra-feature level, we observe strong correlations inside homogeneous texture and KANSEI intensity. The correlations in homogeneous texture indicate that the energy and energy deviation of the captured frequency channels describe essentially the same information in the image data employed in this user study. The components of KANSEI shape and edge histogram are moderately correlated. Both features base

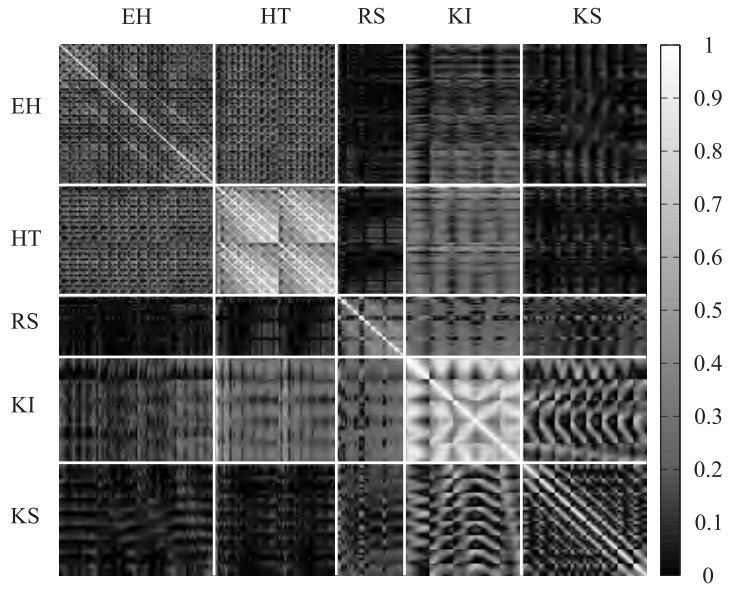


Figure 10.7: The correlation matrix between all feature components. High values (light) indicate high correlations, low values (dark) indicate low correlations. The white lines mark the boundaries between features.

on neighboring image blocks which tend to have correlated content. region shape has the lowest correlations due to the independent basis functions of the Angular Radial Transform.

On the inter-feature level, KANSEI intensity is moderately correlated with all other features. The highest correlation is observed between KANSEI intensity and KANSEI shape. Region shape has low correlations with other features, especially with the two MPEG-7 features. Homogeneous texture correlates with some components of edge histogram. These correlations are expected since edges in an image introduce particular frequencies in the image's frequency domain representation.

10.5.2 Results of the User Study

Hypothesis 1: Low-level features are able to represent visual compositions. We test this hypothesis by evaluating the Prec@16 obtained using the content-based features. Prec@16 is the proportion of relevant retrieval results in the result set of size 16. We choose Prec@16 in order to evaluate the complete result set our system retrieves. See Figure 10.8 for examples of composition sketches and relevant retrieval results. An evaluation of recall is not reasonable because there is no way to create a

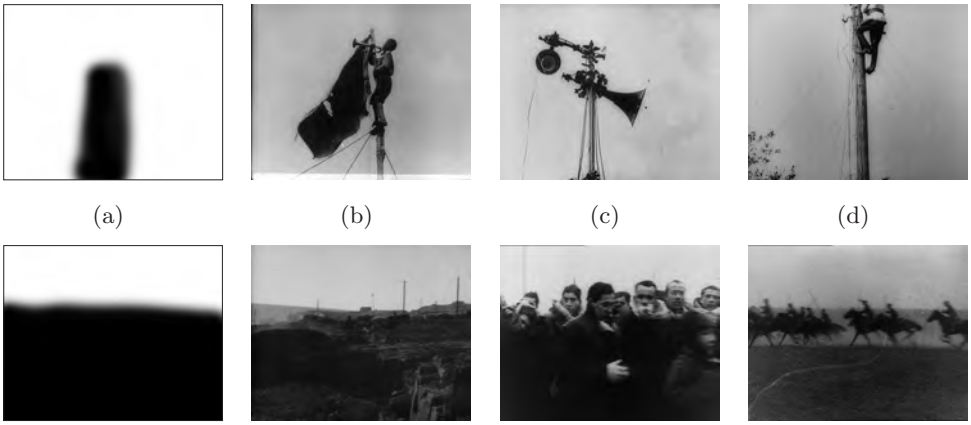


Figure 10.8: Two of the pre-defined query sketches – 10.8(a), 10.8(e) – each with three relevant retrieval results.

	RM	EH	HT	RS	KI	KS	MP7	KSI	ALL
μ	0.07	0.22	0.30	0.34	0.42	0.54	0.37	0.49	0.53
σ	0.06	0.14	0.18	0.20	0.22	0.11	0.19	0.24	0.24

Table 10.4: Mean and standard deviation of Prec@16 for all features and feature combinations.

universally valid ground truth for the keyframes in the data set. A unique assignment of keyframes to composition types is not possible, since this assignment depends on the beholder's subjective assessment.

Table 10.4 lists the mean and standard deviation of Prec@16 for all features and combinations in the study. Note that a Prec@16 value of 1.00 can only be achieved if there are at least 16 relevant examples in the data set which is not the case for all tested sketches. Consequently, we are interested in the relative performance *differences* rather than in absolute precision values.

In order to falsify Hypothesis 1, there should be no significant differences in the Prec@16 values between the random feature and the other content-based features in the study. From the Prec@16 values, we observe that all single features and feature combinations outperform the random feature significantly. The worst-performing real world feature (edge histogram) yields an average Prec@16 of 0.22 while the random feature yields an average Prec@16 of 0.07. This means that Hypothesis 1 is not falsified.

From both the Prec@16 values and offline interviews we conclude that in our experiments the low-level features have the ability to capture aspects of visual composition relevant to the subjects.

RQ 1: Which content-based features perform best? Edge histogram is outperformed by all other single features. The ANOVA confirms (using a level of significance of 5%) this and the following observations. The 0.04 difference between the mean Prec@16 of homogeneous texture and region shape is not significant. The performance differences of KANSEI intensity and the other single features are significant. This makes KANSEI intensity the second best single feature. The best performing single feature is KANSEI shape. KANSEI shape's performance supports the results of the statistical analysis based on WALDI. KANSEI shape captures the variance in the data that is important for retrieval of visual compositions.

In the evaluation of feature combinations, both KSI and ALL outperform MP7. The performance difference between KSI and ALL is not statistically significant and, thus, there are two "best" feature combinations.

The performance differences between the single features and the combinations do not justify statements regarding a clear performance winner. It is nevertheless interesting that KANSEI shape alone yields slightly higher precision than KSI and ALL. However, ANOVA reveals that there is no significant difference between KANSEI shape and ALL. This means that a single feature achieves comparable performance to the feature combinations at lower computational costs.

RQ 2: Which proximity measure preforms better? In order to answer the second research question we analyze the performance differences between the two proximity measures. Cosine similarity yields an average Prec@16 of 0.41 with standard deviation 0.23. Euclidean distance yields an average Prec@16 of 0.39 with standard deviation of 0.22. Although, the Cosine similarity seems to be superior over the Euclidean distance, the factorial ANOVA reveals that there is no significant difference in the performance of the two proximity measures.

RQ 3: Do film experts judge the same retrieval results differently than computer scientists? We investigate the influence of the field of expertise by analyzing

the differences in retrieval performance judgments between computer scientists and film experts. We ask both user groups offline to assess the retrieval system's general ability to represent visual compositions on a five-point scale (deficient - sufficient - satisfactory - good - excellent). Both user groups respond in the range from good to sufficient, with the median for both groups being satisfactory. The statistical analysis of the actual relevance judgments yields an average Prec@16 of 0.38 for computer scientists and of 0.43 for film experts with the same standard deviation of 0.22. These results indicate that film experts asses the relevance differently than the computer scientists. The ANOVA confirms the significance of this difference at a level of significance of 5%. We learn that given identical result sets, film experts rate the relevance of the presented images higher than computer scientists do. This observation is true for all four predefined query sketches employed in this study.

10.6 Summary

Visual composition is an important aspect of accessing visual arts and film. However, little effort has been invested into search and retrieval based on composition so far. We investigate the capability of low-level content-based features for the retrieval of visual compositions in a user study. Our findings suggest that low-level content-based features *are* capable of capturing composition as it is understood by film experts.

Additionally, we learn that film experts assess the relevance of retrieval results to be higher than computer scientists which shows the influence of expertise for composition retrieval. This influence is linked to our finding that film experts, without being aware of it, perceive visual compositions only if there is a strong semantic connection between the query and the result image. Since the proposed technique focuses only on visual similarity film experts are presented with (for them) unexpected results which are semantically unrelated but visually similar. This allows the film experts to analyze visual compositions that they did not perceive before. One long-serving film expert even said: "*The computer sees more than man.*"

Chapter 11

Conclusion

11.1 Summary

This thesis has focused on the retrieval of concepts from archive film material. These films pose novel challenges to automatic analysis due to the presence of sophisticated stylistic aspects and the low quality of the material. The films have not been subject to automatic analysis and retrieval so far. However, film scientists and archivists have been studying these films for decades. During this time, the film experts have developed requirements which are novel in the field of automatic analysis and retrieval. From these requirements we derive syntactical and semantical concepts. In this thesis we develop and present novel methods for the retrieval of the identified syntactical and semantical concepts.

First, we address the detection of less sophisticated concepts with mostly syntactic aspects, such as intertitles and black frames. The corresponding case studies show that even simple tasks become challenging in the context of the investigated film material due to the large number of interfering artifacts.

Films have a hierarchical structure. On a low level, a film consists of shots. The reliable segmentation of a film into shots is the basis for most high-level film analyses. We extend an existing method originally devised for contemporary films to the detection of shot cuts in archive film material by incorporating robust features and introducing a novel fusion scheme that significantly improves the detector's performance. Additionally, we investigate the detection of gradual transitions which are an important stylistic means frequently used in the investigated films. Gradual transition detection in archive

films introduces additional challenges which make the detection more complex than for contemporary material. We perform a first systematic evaluation of gradual transition detection in archive film material. The evaluation shows that gradual transition detection in archive film requires different features and parameter settings and additional verification steps compared to contemporary films.

The next higher level above shots are scenes. We present a novel multimodal framework for scene segmentation. The framework is extensible, requires no a priori information about the films and is applicable to arbitrary film material. We perform a systematic evaluation of the framework's components and parameters. The evaluation shows that scene segmentation of archive films is more demanding than that of contemporary films due to the more sophisticated structure of scenes and the low material quality. Additionally, the evaluation reveals that multimodal processing facilitates scene segmentation in both, archive and contemporary film material.

Similarly to scenes, synchronous montage sequences are semantically related units in a film. We present a novel cross-modal method for the extraction of synchronous montage sequences. For this purpose, we develop a cross-modal correlation measure that simulates human synchrony perception which significantly reduces the number of false positive detections in the experiments. Additionally, we propose a tolerant segmentation scheme that is robust to irregularities and gaps in synchronous montage sequences.

Finally, we investigate motion and visual composition in the archive films. For the retrieval of motion composition we devise a novel trajectory clustering method for highly fragmented motion fields. The method extracts meaningful motion components that allow a compact description of the motion content in a film. We propose an intuitive type of query and robust matching schemes for the retrieval of motion compositions and investigate two retrieval scenarios: the retrieval of shots with user-specified motion compositions and the retrieval of motion continuity between successive shots. In both scenarios we obtain promising results for archive as well as for contemporary film material.

For the retrieval of visual composition we perform a user study to investigate if visual composition as it is understood by film experts can be retrieved automatically. In the user study, we review the applicability of novel and existing low-level content-based features for this retrieval task. We develop a query-by-example system that

takes images with user-specified visual compositions as input and retrieves frames with corresponding visual compositions from a database of archive films. We evaluate the performance of different features, feature combinations, and similarity measures based on the relevance assessments provided by the subjects. The study shows that content-based features have the ability to capture composition as it is understood by film experts. Furthermore, experiments show that the developed retrieval system supports film scientists in gaining novel insights into the historic films.

11.2 Open Topics

We have identified a number of open topics related to the different investigated concepts in this thesis that we plan to address in future research.

Black frames. We focus on the detection of single black frames in Chapter 4. The *repeated* use of black frames in a sequence is an artistic means that is frequently employed in the archive films (see Section 4.1). Similarly to synchronous montage sequences, such sequences have a strong semantic meaning in the films. Automatic retrieval methods for such sequences are needed.

Intertitles. The method for intertitle detection presented in Chapter 4 assumes intertitles to be static. However, there are animated intertitles in archive films, as well. For the detection of such intertitles adequate methods are required. Additionally, the text from the intertitles is a valuable source of context information. Keywords could be extracted from the text automatically to estimate the topics presented in a film. Finally, intertitles indicate topic changes which often coincide with scene boundaries. Consequently, intertitles may be incorporated as an additional clue in scene segmentation.

Scenes. The experiments on scene segmentation in Chapter 7 show that the performance increases with the number of employed content-based features. We expect that scene segmentation further benefits from the integration of more sophisticated features which measure similarity on a higher semantic level. Such features may be obtained for example from face recognizers to group shots with recurring faces and from object detectors to group shots which show similar or identical objects. Additional clues for

scene segmentation may originate from intertitles, synchronous montage sequences, and motion analysis, see the corresponding passages in this section for details.

The scene segmentation framework allows the integration of alternative feature fusion schemes. The investigation and evaluation of different schemes are topics for future research. When for example a large number of features is employed, a fusion scheme based on majority voting would introduce additional robustness.

Synchronous montage sequences. Experiments show that synchronous montage sequences often represent semantically meaningful content that is representative for a film. Consequently, they are well-suited candidates for automatic movie summarization and trailer generation. The integration of synchronous montage sequences would enhance movie summaries and automatically generated trailers.

Synchronous montage sequences are usually subparts of a scene and represent semantically coherent units. The probability that they coincide with a scene boundary is generally low. Consequently, synchronous montage sequences may be employed as additional (and orthogonal) clues in scene segmentation.

Finally, synchronous montage is frequently employed in action scenes in contemporary movies. The proposed method may be extended by features such as loudness, motion, and shot frequency for the detection of action scenes.

Motion composition. An open topic in the context of motion composition is the retrieval of (arbitrary) rhythmic motions. Rhythmic motions may represent typical activities shown in a film such as riding a bike, hammering, and dancing. Different activities may be classified automatically based on the retrieval of rhythmic motion.

Additionally, motion composition can be analyzed at larger scales, across more than one or two successive shots, as well. An open topic is the detection of motion patterns across several shots. An example are several subsequent shots with similar camera motion. Shots in a chasing scene for example may be characterized by a continuous motion direction in several subsequent shots. Such sequences semantically belong together and may be an additional clue for scene segmentation. Additionally, such patterns may be characteristic for different types of scenes, such as action scenes, romance scenes, and dialogs and may be used for scene classification.

Visual composition. Another open topic is the combination of visual composition (which refers to the spatial arrangement of objects) with motion composition (which refers to the arrangement of camera and object motion). Both compositional aspects complement each other and together enable the description (and retrieval) of a much wider spectrum of compositions. The combination of visual composition and motion composition in a single retrieval system further requires the design of a query, that combines both aspects in one simple and intuitive fashion.

Bibliography

[1] A. Adami and D. Barone. A speaker identification system using a model of artificial neural networks for an elevator application. *Information Sciences*, 138(1-4):1–5, October 2001.

[2] B. Adams, C. Dorai, and S. Venkatesh. Toward automatic extraction of expressive elements from motion pictures:tempo. *IEEE Transaction on Multimedia*, 4(4):472–481, 2002.

[3] C. Aggarwal, A. Hinneburg, and D. Keim. On the surprising behavior of distance metrics in high dimensional space. In *Database Theory — ICDT 2001*, volume 1973 of *Lecture Notes in Computer Science*, pages 420–434. Springer Berlin / Heidelberg, 2001.

[4] G. Agostini, M. Longari, and E. Pollastri. Musical instrument timbres classification with spectral features. In *Proceedings of the IEEE Workshop on Multimedia Signal Processing*, pages 97–102, Cannes, France, October 2001. IEEE, IEEE.

[5] G. Agostini, M. Longari, and E. Pollastri. Musical instrument timbres classification with spectral features. *EURASIP Journal on Applied Signal Processing*, 2003(1):5–14, 2003.

[6] G. Ahanger, D. Benson, and T. Little. Video query formulation. *Storage and Retrieval for Image and Video Databases III*, 2420(1):280–291, 1995.

[7] N. Ahmed, T. Natarajan, and K. Rao. Discrete cosine transform. *IEEE Transactions on Computers*, C-23(1):90–93, January 1974.

[8] M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.

[9] ANSI. *Bioacoustical Terminology, ANSI S3.20-1995 (R2003)*. American National Standards Institute, New York, 1995.

[10] E. Ardizzone, M. La Cascia, and D. Molinelli. Motion and color-based video indexing and retrieval. In *Proceedings of the International Conference on Pattern Recognition*, pages 135–139, August 1996.

[11] R. Arnheim. *Kunst und Sehen: Eine Psychologie des schöpferischen Auges*. Walter de Gruyter, 1978.

[12] J. Assfalg, M. Bertini, A. Del Bimbo, W. Nunziati, and P. Pala. Soccer highlights detection and recognition using hmms. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume 1, pages 825–828, 2002.

[13] B. Atal. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America*, 55(6):1304–1312, June 1974.

[14] P. Atrey, M. Hossain, A. El Saddik, and M. Kankanhalli. Multimodal fusion for multimedia analysis: a survey. *Multimedia Systems*, 16(6):345–379, 2010.

[15] P. Atrey, M. Kankanhalli, and R. Jain. Information assimilation framework for event detection in multimedia surveillance systems. *Multimedia Systems*, 12(3):239–253, 2006.

[16] F. Attneave. Dimensions of similarity. *American Journal of Psychology*, 63:516–556, 1950.

[17] Austrian Film Museum. The material is provided with kind permission of the austrian film museum, 2011.

[18] Z. Barzelay and Y. Schechner. Onsets coincidence for cross-modal analysis. *IEEE Transactions on Multimedia*, 12(2):108–120, 2010.

[19] F. Bashir, A. Khokhar, and D. Schonfeld. Real-time motion trajectory-based indexing and retrieval of video sequences. *IEEE Transactions on Multimedia*, 9(1):58–65, January 2007.

[20] F. Beaver. *Dictionary of film terms: the aesthetic companion to film art.* Peter Lang Publishing, 2009.

[21] Y. Bengio, O. Delalleau, and N. Le Roux. The curse of dimensionality for local kernel machines. Technical Report 1258, Department of Computer Science and Operations Research, Université de Montréal, 2005.

[22] J. Bescos, G. Cisneros, J. Martinez, J. Menendez, and J. Cabrera. A unified model for techniques on video-shot transition detection. *IEEE Transactions on Multimedia*, 7(2):293–307, 2005.

[23] S. Birchfield. KLT: an implementation of the Kanade-Lucas-Tomasi feature tracker. http://www.ces. clemson.edu/∼stb/klt, last visited: April 2009.

[24] C. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[25] M. Bober. Mpeg-7 visual shape descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):716–719, 2001.

[26] B. Bogert, M. Healy, and J. Tukey. The quefrency alanysis of time series for echoes: cepstrum, pseudo-autocovariance, cross-cepstrum, and saphe-cracking. In *Proceedings of the Symposium on Time Series Analysis (M. Rosenblatt, Ed.)*, pages 209–243. New York: Wiley, 1963.

[27] D. Bordwell and K. Thompson. *Film art: an introduction.* McGraw-Hill, 8th edition, 2008.

[28] J. Boreczky and L. Rowe. Comparison of video shot boundary detection techniques. *Journal of Electronic Imaging*, 5(2):122–128, 1996.

[29] B. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. *Fifth Annual Workshop on Computational Learning Theory*, pages 144–152, 1992.

[30] P. Bouthemy, C. Hardouin, G. Piriou, and J. Yao. Mixed-state auto-models and motion texture modeling. *Journal of Mathematical Imaging and Vision*, 25(3):387–402, October 2006.

[31] H. Bredin and G. Chollet. Audiovisual speech synchrony measure: application to biometrics. *EURASIP Journal on Applied Signal Processing*, 2007(1):179–179, 2007.

[32] J. Bridle and M. Brown. An experimental automatic word recognition system. JSRU Report No. 1003, Ruislip, England: Joint Speech Research Unit, 1974.

[33] I. Bronstein, K. Semendjajew, G. Musiol, and H. Mühlig. *Taschenbuch der Mathematik*. Harri Deutsch, 5th edition, 2000.

[34] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision*, volume 3024, pages 25–36, 2004.

[35] J. Burnet. *Practical Hints on Light and Shade in Painting*. James Carpenter and Son, 1834.

[36] D. Buzan, S. Sclaroff, and G. Kollios. Extraction and clustering of motion trajectories in video. In *Proceedings of the International Conference on Pattern Recognition.*, pages 521–524, August 2004.

[37] R. Cai, L. Lu, A. Hanjalic, H. Zhang, and L. Cai. A flexible framework for key audio effects detection and auditory context inference. *IEEE Transactions on Speech and Audio Processing*, 14:1026–1039, May 2006.

[38] C. Chan and G. Jones. Affect-based indexing and retrieval of films. In *Proceedings of the annual ACM International Conference on Multimedia*, pages 427–430, Singapore, Singapore, 2005. ACM Press.

[39] H. Chang and S. Lai. Robust camera motion estimation and classification for video analysis. *Visual Communications and Image Processing*, 5308(1):912–923, January 2004.

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

[41] S. Chen, M. Shyu, W. Liao, and C. Zhang. Scene change detection by audio and video clues. In *Proceedings of the IEEE Conference on Multimedia and Expo*, volume 2, pages 365–368, 2002.

[42] Y. Cho, M. Kim, and S. Kim. A spectrally mixed excitation (smx) vocoder with robust parameter determination. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 601–604, May 1998.

[43] W. Chu, W. Cheng, J. Hsu, and J. Wu. Toward semantic indexing and retrieval using hierarchical audio models. *Multimedia Systems*, 10(6):570–583, May 2005.

[44] Z. Chuang and C. Wu. Emotion recognition using acoustic features and textual content. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume 1, pages 53–56, Taipei, Taiwan, June 2004. IEEE, IEEE.

[45] CIE. *Colorimetry*, volume 15. Vienna: Commission Internationale de l'Éclairage, 3rd edition, 2004.

[46] M. Cooper and J. Foote. Scene boundary detection via video self-similarity analysis. In *Proceedings of the International Conference on Image Processing*, volume 3, pages 378–3813, 2001.

[47] M. Cooper and J. Foote. Video segmentation via temporal pattern classification. *IEEE Transactions on Multimedia*, 9(3):610–618, 2007.

[48] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[49] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.

[50] S. Dagtas, W. Al-Khatib, A. Ghafoor, and R. Kashyap. Models for motion-based video indexing and retrieval. *IEEE Transactions on Image Processing*, 9(1):88–101, January 2000.

[51] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366, August 1980.

[52] Y. Deng, B. Manjunath, C. Kenney, M. Moore, and H. Shin. An efficient color representation for image retrieval. *IEEE Transactions on Image Processing*, 10(1):140–147, January 2001.

[53] L. Devillers. Challenges in real-life emotion annotation and machine learning based detection. *Neural Networks*, 18(4):407–422, 2005.

[54] N. Dimitrova and F. Golshani. Motion recovery for video content classification. *ACM Transactions on Information Systems*, 13(4):408–439, October 1995.

[55] A. Divakaran, K. Peker, R. Radhakrishnan, Z. Xiong, and R. Cabasson. Video summarization using mpeg-7 motion activity and audio descriptors. In A. Rosenfeld, D. Doermann, and D. DeMenthon, editors, *Video Mining*, pages 91–121. Kluwer Academic Publishers, 2003.

[56] C. Dorai and S. Venkatesh. Computational media aesthetics: Finding meaning beautiful. *IEEE Multimedia*, 8(4):10–12, 2001.

[57] R. Duda, P. Hart, and D. Stork. *Pattern Classification 2nd edition*. Wiley, 2001.

[58] H. Eidenberger. Evaluation and analysis of similarity measures for content-based visual information retrieval. *Multimedia Systems*, 12(2):71–87, 2006.

[59] S. Eisenstein. *Film Form: Essays in Film Theory*, chapter A Dialectic Approach to Film Form, pages 45–63. Harcourt Brace and Company, 1977.

[60] R. Ewerth, M. Schwalb, P. Tessmann, and B. Freisleben. Segmenting moving objects in mpeg videos in the presence of camera motion. In *Proceedings of the International Conference on Image Analysis and Processing*, pages 819–824, September 2007.

[61] R. Fablet and P. Bouthemy. Motion-based feature extraction and ascendant hierarchical classification for video indexing and retrieval. In *Proceedings of the International Conference on Visual Information Systems*, pages 221–228, June 1999.

[62] J. Fauqueur and N. Boujemaa. Mental image search by boolean composition of region categories. *Multimedia Tools and Applications*, 31(1):95–117, 2006.

[63] L. Fisher. Enthusiasm: From kino-eye to radio-eye. In E. Weis and J. Belton, editors, *Film Sound-Theory and Practice*, pages 247–264. Columbia Univ. Press, 1985.

[64] S. Fisher. *Statistical methods for research workers*. Oliver and Boyd, 1970.

[65] T. Fletcher. Support vector machines explained. available online, http://www.tristanfletcher.co.uk/SVM Explained.pdf, last visited: October 2011, 2009.

[66] 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. *Computer*, 28(9):23–32, 1995.

[67] J. Foote. Visualizing music and audio using self-similarity. In *Proceedings of the 7th ACM International Conference on Multimedia*, pages 77–80. ACM, 1999.

[68] J. Foote. Automatic audio segmentation using a measure of audio novelty. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume 1, pages 452–455, New York, NY, August 2000. IEEE, IEEE.

[69] J. Foote and S. Uchihashi. The beat spectrum: a new approach to rhythm analysis. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 881–884. IEEE, IEEE, 2001.

[70] W. Fujisaki and S. Nishida. Feature-based processing of audio-visual synchrony perception revealed by random pulse trains. *Vision Research*, 47(8):1075–1093, 2007.

[71] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40, January 1975.

[72] A. Fuxjäger. Wenn Filmwissenschaftler versuchen sich Maschinen verständlich zu machen - zur mangelden Operationalisierbarkeit des Begriffs “Einstellung” für die Filmanalyse. *Maske und Kothurn*, 3, 2009.

[73] B. Gajic and K. Paliwal. Robust speech recognition using features based on zero crossings with peak amplitudes. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 64–67, Hong Kong, China, April 2003. IEEE, IEEE.

[74] S. Gauglitz, T. Höllerer, and M. Turk. Evaluation of interest point detectors and feature descriptors for visual tracking. *International Journal of Computer Vision*, pages 1–26, 2011.

[75] M. Gelgon and P. Bouthemy. Determining a structured spatio-temporal representation of video content for efficient visualization and indexing. In *Proceedings of the European Conference on Computer Vision*, pages 595–609, June 1998.

[76] T. Gevers. Robust segmentation and tracking of colored objects in video. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(6):776–781, June 2004.

[77] E. Guaus and E. Batlle. Visualization of metre and other rhythm features. In *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology*, pages 282–285, Darmstadt, Germany, December 2003. IEEE, IEEE.

[78] J. Hadamard. *Sur les problèmes aux dérivées partielles et leur signification physique*. Princeton University Bulletin, 1902.

[79] J. Han and M. Kamber. *Data Mining*. Morgan Kaufmann Publishers (Elsevier), 2006.

[80] S. Handzo. Appendix: A narrative glossary of film sound technology. In E. Weis and J. Belton, editors, *Film Sound-Theory and Practice*, pages 383–426. Columbia Univ. Press, 1985.

[81] A. Hanis and T. Sziranyi. Measuring the motion similarity in video indexing. In *Proceedings of the EURASIP Conference focused on Video/Image Processing and Multimedia Communications*, volume 2, pages 507–512, July 2003.

[82] A. Hanjalic. Shot-boundary detection: unraveled and resolved? *IEEE Transactions on Circuits and Systems for Video Technology*, 12(2):90–105, February 2002.

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

[84] A. Hanjalic, R. Lagendijk, and J. Biemond. Automatically segmenting movies into logical story units. In *Visual Information and Information Systems*, volume 1614 of *Lecture Notes in Computer Science*, pages 654–654. Springer Berlin / Heidelberg, 1999.

[85] R. Haralick and L. Shapiro. *Computer and robot vision.* Number Vol. 2 in Computer and Robot Vision. Addison-Wesley Pub. Co., 1993.

[86] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.

[87] A. Haupmann and M. Witbrock. Story segmentation and detection of commercials in broadcast news video. *IEEE Journal on Advances in Digital Libraries Conference*, 0:168–179, 1998.

[88] J. Hawkins. *Textural properties for pattern recognition.* Academic Press, New York, 1969.

[89] H. Hermansky and N. Morgan. Rasta processing of speech. *IEEE Transactions on Speech and Audio Processing*, 2:578–589, 1994.

[90] J. Herre, E. Allamanche, and O. Hellmuth. Robust matching of audio signals using spectral flatness features. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 127–130, New Paltz, NY, October 2001. IEEE, IEEE.

[91] P. Herrera-Boyer, G. Peeters, and S. Dubnov. Automatic classification of musical instrument sounds. *Journal of New Music Research*, 32(1), 2003.

[92] J. Hershey and J Movellan. Audio-vision: Using audio-visual synchrony to locate sounds. In *Advances in Neural Information Processing Systems*, pages 813–819, 2000.

[93] A. Hervieu, P. Bouthemy, and J-P. Le Cadre. Video event classification and detection using 2d trajectories. In *Proceedings of the International Conference on Computer Vision Theory and Applications*, pages 110–123, January 2008.

[94] W. Hess. *Pitch determination of speech signals : algorithms and devices*. Springer, Berlin, Germany, 1983.

[95] L. Holmstrom, P. Koistinen, J. Laaksonen, and E. Oja. Neural and statistical classifiers-taxonomy and two case studies. *IEEE Transactions on Neural Networks*, 8(1):5–17, January 1997.

[96] B. Horn. Image intensity understanding. AI Memos, 1975.

[97] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:135–203, 1981.

[98] D. Hosmer and S. Lemeshow. *Applied logistic regression*. Wiley, New York, NJ, USA, 2nd edition, 2000.

[99] J. Huang, S. Kumar, M. Mitra, W. Zhu, and R. Zabih. Image indexing using color correlograms. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 762–768, June 1997.

[100] T. Huttunen. Montage culture. University of Helsinki, 2005.

[101] ISO-IEC. *Information Technology - Multimedia Content Description Interface*. 15938. ISO/IEC, Moving Pictures Expert Group, 1st edition, 2002.

[102] A. Jain, R. Duin, and Jianchang M. Statistical pattern recognition: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, January 2000.

[103] N. Jayant and P. Noll. *Digtial Coding of Waveforms Principles and Applications to Speech and Video*. Prentice-Hall signal processing series. Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

[104] H. Jiang, J. Bai, S. Zhang, and B. Xu. Svm-based audio scene classification. In *Proceedings of the IEEE International Conference on Natural Language Processing and Knowledge Engineering*, pages 131–136, Wuhan, China, October 2005. IEEE, IEEE.

[105] Y. Kawai, H. Sumiyoshi, and N. Yagi. Shot boundary detection at TRECVID 2007. In *TREC Video Retrieval Evaluation Online Proceedings*, Gaithersburg, 2007. NIST.

[106] B. Kedem. Spectral analysis and discrimination by zero-crossings. *IEEE Proceedings*, 74:1477–1493, 1986.

[107] M. Khan and W. Al-Khatib. Machine-learning based classification of speech and music. *Multimedia Systems*, 12(1):55–67, August 2006.

[108] M. Khan, W. Al-Khatib, and M. Moinuddin. Automatic classification of speech and music using neural networks. In *MMDB '04: Proceedings of the 2nd ACM international workshop on Multimedia databases*, pages 94–99. ACM Press, 2004.

[109] E. Kidron, Y. Schechner, and M. Elad. Cross-modal localization via sparsity. *IEEE Transactions on Signal Processing*, 55(4):1390–1404, April 2007.

[110] H. Kim, N. Moreau, and T. Sikora. *MPEG-7 audio and beyond*. Wiley, West Sussex, England, 2005.

[111] H. Kobayashi, Y. Okouchi, and S. Ota. Image retrieval system using kansei features. *PRICAI'98: Topics in Artificial Intelligence*, pages 626–635, 1998.

[112] M. Kokare, B. Chatterji, and P. Biswas. Comparison of similarity metrics for texture image retrieval. In *Proceedings of the Conference on Convergent Technologies for Asia-Pacific Region*, volume 2, pages 571–575, 2003.

[113] S. Kopf, T. Haenselmann, D. Farin, and W. Effelsberg. Automatic generation of video summaries for historical films. In *Proceedings of the International Conference on Multimedia and Expo*, volume 3, pages 2067–2070, 2004.

[114] C. Krumhansl. Concerning the applicability of geometric models to similarity data: The interrelationship between similarity and spatial density. *Psychological Review*, 85(5):445–463, 1978.

[115] G. Lance and W. Williams. Mixed data classificatory programs. *Agglomerative Systems Australian Company Journal*, 9:373–380, 1967.

[116] R. Lancini, F. Mapelli, and R. Pezzano. Audio content identification by using perceptual hashing. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume 1, pages 739–742, Taipei, Taiwan, June 2004. IEEE, IEEE.

[117] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, August 2005.

[118] M. Lew. *Principles of visual information retrieval*. Springer, London, Great Britain, January 2001.

[119] B. Li, E. Chang, and Y. Wu. Discovery of a perceptual distance function for measuring image similarity. *Multimedia Systems*, 8(6):512–522, 2003.

[120] T. Li and M. Ogihara. Music genre classification with taxonomy. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 197–200. IEEE, IEEE, March 2005.

[121] T. Li and G. Tzanetakis. Factors in automatic musical genre classification of audio signals. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 143–146, New Paltz, New York, October 2003. IEEE, IEEE.

[122] X. Li, W. Hu, and W. Hu. A coarse-to-fine strategy for vehicle motion trajectory clustering. In *Proceedings of the International Conference on Pattern Recognition*, pages 591–594, August 2006.

[123] R. Lienhart. Reliable transition detection in videos: A survey and practitioner's guide. *International Journal of Image and Graphics*, 1(3):469–486, 2001.

[124] R. Lienhart, C. Kuhmunch, and W. Effelsberg. On the detection and recognition of television commercials. In *International Conference on Multimedia Computing and Systems*, pages 509–516, Los Alamitos, CA, USA, 1997. IEEE Computer Society.

[125] R. Lienhart, S. Pfeiffer, and W. Effelsberg. Video abstracting. *Communications of the ACM*, 40(12):54–62, 1997.

[126] H. Liu, D. Song, S. Rüger, R. Hu, and V. Uren. Comparing dissimilarity measures for content-based image retrieval. In *Information Retrieval Technology*, volume 4993 of *Lecture Notes in Computer Science*, pages 44–50. Springer Berlin / Heidelberg, 2008.

[127] M. Liu and C. Wan. A study on content-based classification and retrieval of audio database. In *Proceedings of the International Symposium on Database Engineering and Applications*, pages 339–345, Grenoble, France, July 2001. IEEE Computer Society.

[128] Z. Liu, D. Gibbon, E. Zavesky, B. Shahraray, and P. Haffner. AT&T research at TRECVID 2006. In *TREC Video Retrieval Evaluation Online Proceedings*, Gaithersburg, 2006. NIST.

[129] Z. Liu, J. Huang, Y. Wang, and T. Chen. Audio feature extraction and analysis for scene classification. In *Proceedings of the IEEE Workshop on Multimedia Signal Processing*, pages 343–348, Princeton, NJ, June 1997. IEEE, IEEE.

[130] Z. Liu, Y. Wang, and T. Chen. Audio feature extraction and analysis for scene segmentation and classification. *The Journal of VLSI Signal Processing*, 20(1-2):61–79, October 1998.

[131] Z. Liu, E. Zavesky, D. Gibbon, B. Shahraray, and P. Haffner. AT&T research at TRECVID 2007. In *TREC Video Retrieval Evaluation Online Proceedings*, Gaithersburg, 2007. NIST.

[132] A. Llagostera Casanovas, G. Monaci, P. Vandergheynst, and R. Gribonval. Blind audiovisual source separation based on sparse redundant representations. *IEEE Transactions on Multimedia*, 12(5):358–371, August 2010.

[133] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[134] L. Lu, H. Zhang, and S. Li. Content-based audio classification and segmentation by using support vector machines. *Multimedia Systems*, 8(6):482–492, April 2003.

[135] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, pages 121–130, 1981.

[136] Y.-F. Ma, L. Lu, H.-J. Zhang, and M. Li. A user attention model for video summarization. In *Proceedings of the tenth ACM International Conference on Multimedia*, pages 533–542, New York, NY, USA, 2002. ACM.

[137] D. Makris and T. Ellis. Learning semantic scene models from observing activity in visual surveillance. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 35(3):397–408, June 2005.

[138] B. Manjunath, J. Ohm, V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):703–715, June 2001.

[139] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004.

[140] E. Mathias and A. Conci. Comparing the influence of color spaces and metrics in content-based image retrieval. In *Proceedings of the International Symposium on Computer Graphics, Image Processing, and Vision*, pages 371–378, October 1998.

[141] R. Meddis and L. O'Mard. A unitary model of pitch perception. *The Journal of the Acoustical Society of America*, 102(3):1811–1820, September 1997.

[142] B. Mehtre, M. Kankanhalli, and W. Lee. Shape measures for content based image retrieval: A comparison. *Information Processing & Management*, 33(3):319–337, 1997.

[143] J. Mercer. Functions of positive and negative type and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society*, A 209(441–458):415–446, 1909.

[144] Merriam-Webster Incorporated. Merriam-webster dictionary. http://www.merriam-webster.com, last visited: August 2011.

[145] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1615–1630, 2005.

[146] D. Mitrović, S. Hartlieb, M. Zeppelzauer, and M. Zaharieva. Scene segmentation in artistic archive documentaries. *HCI in Work and Learning, Life and Leisure*, pages 400–410, 2010.

[147] D. Mitrović, M. Zeppelzauer, and C. Breiteneder. Features for content-based audio retrieval. *Advances in Computers*, 78:71–150, 2010.

[148] F. Mokhtarian and A. Mackworth. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):789–805, 1992.

[149] G. Monaci and P. Vandergheynst. Audiovisual gestalts. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition Workshop*, page 200, 2006.

[150] S. Moncrieff, C. Dorai, and S. Venkatesh. Affect computing in film through sound energy dynamics. In *ACM Multimedia Conference*, pages 525–527, 2001.

[151] B. Moore. *An Introduction to the Psychology of Hearing*. Academic Press, Amsterdam, The Netherlands, 5th edition, 2004.

[152] F. Mörchen, A. Ultsch, M. Thies, and I. Löhken. Modeling timbre distance with temporal statistics from polyphonic music. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):81–90, January 2006.

[153] F. Mörchen, A. Ultsch, M. Thies, I. Löhken, M. Nöcker, C. Stamm, N. Efthymiou, and M. Kümmerer. Musicminer: Visualizing timbre distances of music as topographical maps. Technical Report, 2005.

[154] R. Narasimha, A. Savakis, R. Rao, and R. De Queiroz. Key frame extraction using mpeg-7 motion descriptors. In *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers, 2003*, volume 2, pages 1575–1579. IEEE, 2003.

[155] A. Nefian, L. Liang, X. Pi, X. Liu, and K. Murphy. Dynamic bayesian networks for audio-visual speech recognition. *EURASIP Journal on Applied Signal Processing*, 2002(1):1274–1288, 2002.

[156] R. Nelson and R. Polana. Qualitative recognition of motion using temporal texture. *CVGIP: Image Understanding*, 56(1):78–89, July 1992.

[157] C. Ngo, T. Pong, H. Zhang, and R. Chin. Motion characterization by temporal slices analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 768–773, 2000.

[158] A. Noll. Short-time spectrum and "cepstrum" techniques for vocal-pitch detection. *The Journal of the Acoustical Society of America*, 36(2), 1964.

[159] E. Pampalk, A. Rauber, and D. Merkl. Content-based organization and visualization of music archives. In *Proceedings of the tenth ACM international conference on Multimedia*, pages 570–579. ACM Press, 2002.

[160] C. Panagiotakis and G. Tziritas. A speech/music discriminator based on rms and zero-crossings. *IEEE Transactions on Multimedia*, 7(1):155–166, February 2005.

[161] K. Pearson. On lines and planes of closest fit to a system of points in space. *Pilosophy Magazine*, 2:559–572, 1901.

[162] G. Peeters. A large set of audio features for sound description (similarity and classification) in the cuidado project. Technical Report, 2004.

[163] C. Perng, H. Wang, S. Zhang, and S. Parker. Landmarks: A new model for similarity-based pattern querying in time series databases. In *Conference on Data Engineering*, page 33–42, 2000.

[164] D. Petrelli and D. Auld. An examination of automatic video retrieval technology on access to the contents of an historical video archive. *Program: electronic library and information systems*, 42(2):115–136, 2008.

[165] S. Pfeiffer. The importance of perceptive adaptation of sound features for audio content processing. In *Proceedings SPIE Conferences, Electronic Imaging 1999, Storage and Retrieval for Image and Video Databases VII*, pages 328–337, San Jose, California, January 1999.

[166] S. Pfeiffer, R. Lienhart, and W. Effelsberg. Scene determination based on video and audio features. *Multimedia Tools and Applications*, 15(1):59–81, September 2001.

[167] F. Pitie, A. Kokaram, and R. Dahyot. N-dimensional probability density function transfer and its application to colour transfer. In *International Conference on Computer Vision*, volume 2, pages 1434–1439, 2005.

[168] H. Poore. *Pictorial Composition and the Critical Judgement of Pictures*. Baker & Taylor. Reprint by University of Michigan Library, 1903.

[169] W. Pratt. *Digital image processing: PIKS Scientific inside*. Wiley-Interscience publication. Wiley-Interscience, 2007.

[170] V. Rabaud and S. Belongie. Counting crowded moving objects. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 705–711, June 2006.

[171] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–285, 1989.

[172] L. Rabiner and B. Juang. *Fundamentals of speech recognition*. Prentice Hall Inc., Englewood Cliffs, New Jersey, 1993.

[173] L. Rabiner and R. Schafer. *Digital Processing of Speech Signals*. Prentice Hall Inc., Englewood Cliffs, New Jersey, 1978.

[174] A. Ramalingam and S. Krishnan. Gaussian mixture modeling using short time fourier transform features for audio fingerprinting. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 1146–1149, Amsterdam, The Netherlands, July 2005. IEEE, IEEE.

[175] Z. Rasheed and M. Shah. Detection and representation of scenes in videos. *IEEE Transactions on Multimedia*, 7(6):1097–1105, December 2005.

[176] S. Ravindran, K. Schlemmer, and D. Anderson. A physiologically inspired method for audio classification. *EURASIP Journal on Applied Signal Processing*, 2005(9):1374–1381, 2005.

[177] J. Rennie. Derivation of the f-measure. *In other words*, pages 1–4, 2004.

[178] S. Rüger. *Multimedia information retrieval, Synthesis Lectures on Information Concepts, Retrieval and Services.* Morgan & Claypool Publishers, 2010.

[179] J. Ricard, D. Coeurjolly, and A. Baskurt. Generalization of angular radial transform. In *International Conference on Image Processing*, pages 2211–2214, 2004.

[180] E. Rosch. Cognitive reference points. *Cognitive Psychology*, 7(4):532–547, 1975.

[181] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[182] A. Rosenfeld and E. Troy. Visual texture analysis. Technical report, TR-70-116, Maryland University, College Park (USA). Computer Science Center, 1970.

[183] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. Segmenting, modeling, and matching video clips containing multiple moving objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.*, volume 2, pages 914–921, 2004.

[184] G. Salton and M. McGill. *Introduction to modern information retrieval.* New York [etc.] : McGraw-Hill, 1983.

[185] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[186] S. Santini and R. Jain. Similarity measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):871–883, September 1999.

[187] J. Saunders. Real-time discrimination of broadcast speech/music. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 993–996, Atlanta, GA, May 1996. IEEE, IEEE.

[188] P. Schallauer, W. Bailer, R. Morzinger, H. Furntratt, and G. Thallinger. Automatic quality analysis for film and video restoration. In *IEEE International Conference on Image Processing 2007. ICIP 2007.*, volume 4, pages 9–12, September 2007.

[189] E. Scheirer and M. Slaney. Construction and evaluation of a robust multifeature speech/music discriminator. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1331–1334, Munich, Germany, April 1997.

[190] I. Schmitt. *Ähnlichkeitssuche in Multimedia-Datenbanken*. Oldenbourg Verlagsgruppe, 2005.

[191] N. Sebe, Q. Tian, E. Loupias, M. Lew, and T. Huang. Evaluation of salient point techniques. *Image and Vision Computing*, 21(13-14):1087–1095, 2003.

[192] I. Sethi and N. Patel. A statistical approach to scene change detection. In *Proceedings of SPIE*, volume 2420, pages 329–338, 1995.

[193] R. Shepard. Toward a universal law of generalization for psychological science. *Science*, 237(4820):1317–1323, 1987.

[194] J. Shi and C. Tomasi. Good features to track. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 593–600, June 1994.

[195] S. Sinha, J. Frahm, M. Pollefeys, and Y. Genc. Gpu-based video feature tracking and matching. In *Proceedings of the Workshop on Edge Computing Using New Commodity Architectures.*, May 2006.

[196] M. Slaney and M. Covell. Facesync: A linear operator for measuring synchronization of video facial images and audio tracks. In *Advances in Neural Information Processing Systems*, pages 814–820, 2000.

[197] A. Smeaton, P. Over, and A. Doherty. Video shot boundary detection: Seven years of TRECVid activity. *Computer Vision and Image Understanding*, 114(4):411–418, 2010.

[198] A. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[199] 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*, 22(12):1349–1380, December 2000.

[200] C. Snoek, M. Worring, O. de Rooij, K van de Sande, R. Yan, and A. Hauptmann. Videolympics: Real-time evaluation of multimedia retrieval systems. *IEEE Multimedia*, pages 86–91, 2008.

[201] H. Srinivasan and M. Kankanhalli. Harmonicity and dynamics-based features for audio. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 321–324, Montreal, Canada, May 2004. IEEE, IEEE.

[202] S. Srinivasan, D. Petkovic, and D. Ponceleon. Towards robust features for classifying audio in the cuevideo system. In *Proceedings of the 7th ACM international conference on Multimedia (Part 1)*, pages 393–400. ACM Press, 1999.

[203] S. Stevens, J. Volkmann, and E. Newman. A scale for the measurement of the psychological magnitude pitch. *Journal of the Acoustical Society of America*, 8(3):185–190, January 1937.

[204] S.S. Stevens. On the psychophysical law. *Psychological Review*, 64(3):153–181, May 1957.

[205] S. Sukittanon, L. Atlas, and W. Pitton. Modulation-scale analysis for content identification. *IEEE Transactions on Signal Processing*, 52(10):3023–3035, 2004.

[206] H. Sundaram and S. Chang. Video scene segmentation using video and audio features. In *IEEE International Conference on Multimedia and Expo, 2000.*, pages 1145–1148, Piscataway, NY, USA, 2000. IEEE.

[207] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[208] R. Szeliski. *Computer Vision: Algorithms and Applications*. Texts in Computer Science. Springer, 2010.

[209] H. Tamura, S. Mori, and T. Yamawaki. Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics*, 8(6):460–473, June 1978.

[210] A. Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM, Society for Industrial and Applied Mathematics, Philadelphia, Pa., 2005.

[211] C. Taskiran and E. Delp. Video scene change detection using the generalized sequence trace. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, volume 5, pages 2961–2964, 1998.

[212] W. Tavanapong and J. Zhou. Shot clustering techniques for story browsing. *IEEE Transactions on Multimedia*, 6(4):517–527, August 2004.

[213] H. Terasawa, M. Slaney, and J. Berger. Perceptual distance in timbre space. In *Proceedings of Eleventh Meeting of the International Conference on Auditory Display*, pages 61–68, Limerick, Ireland, July 2005.

[214] G. Thalin. Deshaker - video stabilizer. http://guthspot.se/video/deshaker.htm, last visited: August 2011.

[215] T. Tode and B. Wurm, editors. *Dziga Vertov: The Vertov Collection at the Austrian Film Museum*. Austrian Film Museum/SYNEMA, 2006.

[216] B. Truong, S. Venkatesh, and C. Dorai. Scene extraction in motion pictures. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(1):5–15, January 2003.

[217] B. Truong, S. Venkatesh, and C. Dorai. Extraction of film takes for cinematic analysis. *Multimedia Tools and Applications*, 26(3):277–298, 2005.

[218] T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: a survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, 2008.

[219] A. Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, 1977.

[220] A. Tversky and I. Gati. Similarity, separability, and the triangle inequality. *Psychological Review*, 89(2):123–154, 1982.

[221] G. Tzanetakis. *Manipulation, analysis and retrieval systems for audio signals*. PhD. Thesis. Computer Science Department, Princeton University, 2002.

[222] G. Tzanetakis. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, July 2002. 2002.

[223] Unknown photographer. The last supper. http://www.friendsofart.net, last visited: October 2011.

[224] O. Urhan, K. Gullu, and S. Erturk. Shot-cut detection for b&w archive films using best-fitting kernel. *International Journal of Electronics and Communications*, 61(7):463–468, 2007.

[225] Oguzan Urhan, Kemal M. Gullu, and Sarp Erturk. Modified phase-correlation based robust hard-cut detection with application to archive film. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(6):753–770, 2006.

[226] C. van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 1979.

[227] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

[228] V. Vapnik and A. Lerner. Pattern recognition using generalized portrait method. In *Neural Information Processing Systems*, 1963.

[229] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms, 2008.

[230] T. Veit, F. Cao, and P. Bouthemy. Space-time a contrario clustering for detecting coherent motions. In *Proceedings of the IEEE International Conference on Robotics and Automation.*, pages 33–39, April 2007.

[231] A. Wang, A. Divakaran, A. Vetro, S. Chang, and H. Sun. Survey of compressed-domain features used in audio-visual indexing and analysis. *Journal of Visual Communication and Image Representation*, 14(2):150–183, June 2003.

[232] G. Wang, D. Xiao, and J. Gu. Review on vehicle detection based on video for traffic surveillance. In *Proceedings of the IEEE International Conference on Automation and Logistics*, pages 2961–2966. IEEE, 2008.

[233] H. Wang and H. Lin. A spectral clustering approach to motion segmentation based on motion trajectory. In *Proceedings of the International Conference on Multimedia and Expo.*, pages 793–796, July 2003.

[234] X. Wang, K. Tieu, and E. Grimson. Learning semantic scene models by trajectory analysis. In *Proceedings of the European Conference on Computer Vision*, May 2006.

[235] X. Wang, S. Wang, H. Chen, and M. Gabbouj. A Shot Clustering Based Algorithm for Scene Segmentation. In *Proceedings of the 2007 International Conference on Computational Intelligence and Security Workshops*, pages 252–259, Washington, DC, USA, 2007. IEEE Computer Society.

[236] Y. Wang, Z. Liu, and J. Huang. Multimedia content analysis-using both audio and visual clues. *IEEE Signal Processing Magazine*, 17(6):12–36, November 2000.

[237] P. Ward. *Picture composition for film and television.* Focal Press, 2003.

[238] T. Wold, D. Blum, and J. Wheaton. Content-based classification, search, and retrieval of audio. *IEEE Multimedia*, 3(3):27–36, 1996.

[239] C. Xu, N. Maddage, and X. Shao. Automatic music classification and summarization. *IEEE Transactions on Speech and Audio Processing*, 13(3):441–450, May 2005.

[240] Minerva Yeung, Boon-Lock Yeo, and Bede Liu. Segmentation of video by clustering and graph analysis. *Computer Vision and Image Understanding*, 71(1):94–109, 1998.

[241] J. Yuan, H. Wang, L. Xiao, W. Zheng, J. Li, F. Lin, and B. Zhang. A formal study of shot boundary detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(2):168–186, 2007.

[242] J. Yuan, H. Wang, L. Xiao, W. Zheng, J. Li, F. Lin, and B. Zhang. A formal study of shot boundary detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(2):168–186, 2007.

[243] J. Yuan, W. J. Zheng, L. Chen, et al. Tsinghua university at trecvid 2004: Shot boundary detection and high-level feature extraction. In *NIST workshop of TRECVID*, Gaithersburg, 2004. NIST.

[244] Jinhui Yuan, Jianmin Li, Fuzong Lin, and Bo Zhang. A unified shot boundary detection framework based on graph partition model. In *Proceedings of the 13th annual ACM international conference on Multimedia*, MULTIMEDIA '05, pages 539–542, New York, NY, USA, 2005. ACM.

[245] R. Zabih, J. Miller, and K. Mai. A feature-based algorithm for detecting and classifying scene breaks. In *Proceedings of the 3rd ACM International Conference on Multimedia*, pages 189–200, New York, NY, USA, 1995. ACM.

[246] M. Zaharieva, M. Zeppelzauer, C. Breiteneder, and D. Mitrović. Camera take reconstruction. In *Proceedings of the IEEE Multimedia Modeling Conference*, pages 379–388, 2010.

[247] M. Zaharieva, M. Zeppelzauer, D. Mitrović, and C. Breiteneder. Archive film comparison. *International Journal of Multimedia Data Engineering and Management*, 1:41–56, 2010.

[248] D. Zhang and G. Lu. Evaluation of mpeg-7 shape descriptors against other shape descriptors. *Multimedia Systems*, 9:15–30, 2003.

[249] H. Zhang, A. Kankanhalli, and S. Smoliar. Automatic partitioning of full-motion video. *Multimedia Systems*, 1(1):10–28, 1993.

[250] T. Zhang and C. Kuo. *Content-Based Audio Classifcation and Retrieval for Audiovisual Data Parsing*. Kluwer Academic Publishers, Boston, Massachusetts, 2001.

[251] S. Zhu and Y. Liu. Video scene segmentation and semantic representation using a novel scheme. *Multimedia Tools and Applications*, 42:183–205, 2009.

[252] E. Zwicker. Subdivision of the audible frequency range into critical bands (frequenzgruppen). *The Journal of the Acoustical Society of America*, 33:248, 1961.

[253] E. Zwicker and H. Fastl. *Psychoacoustics: Facts and Models*. Springer, Berlin, Heidelberg, Germany, 2nd edition, 1999.

List of Figures

LIST OF FIGURES

LIST OF FIGURES

List of Tables