Manfred del Fabro, DI

NON-SEQUENTIAL DECOMPOSITION, COMPOSITION AND PRESENTATION OF MULTIMEDIA CONTENT

DISSERTATION

zur Erlangung des akademischen Grades
Doktor
der Technischen Wissenschaften

Alpen-Adria Universitaet Klagenfurt
Faculty of Technical Siences

1. Begutachter:	Prof. Dr. Laszlo Böszörmenyi
Institut:	Alpen-Adria Universitaet Klagenfurt
2. Begutachter:	Prof. Dr. Alan Hanjalic
Institut:	Delft University of Technology

December 2011

Ehrenwörtliche Erklärung

Ich erkläre ehrenwörtlich, dass ich die vorliegende wissenschaftliche Arbeit selbstständig angefertigt und die mit ihr unmittelbar verbundenen Tätigkeiten selbst erbracht habe. Ich erkläre weiters, dass ich keine anderen als die angegebenen Hilfsmittel benutzt habe. Alle aus gedruckten, ungedruckten oder dem Internet im Wortlaut oder im wesentlichen Inhalt übernommenen Formulierungen und Konzepte sind gemäß den Regeln für wissenschaftliche Arbeiten zitiert und durch Fußnoten bzw. durch andere genaue Quellenangaben gekennzeichnet.

Die während des Arbeitsvorganges gewährte Unterstützung einschließlich signifikanter Betreuungshinweise ist vollständig angegeben.

Die wissenschaftliche Arbeit ist noch keiner anderen Prüfungsbehörde vorgelegt worden. Diese Arbeit wurde in gedruckter und elektronischer Form abgegeben. Ich bestätige, dass der Inhalt der digitalen Version vollständig mit dem der gedruckten Version übereinstimmt.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Declaration of honour

I hereby confirm on my honour that I personally prepared the present academic work and carried out myself the activities directly involved with it. I also confirm that I have used no resources other than those declared. All formulations and concepts adopted literally or in their essential content from printed, unprinted or Internet sources have been cited according to the rules for academic work and identified by means of footnotes or other precise indications of source.

The support provided during the work, including significant assistance from my supervisor has been indicated in full.

The academic work has not been submitted to any other examination authority. The work is submitted in printed and electronic form. I confirm that the content of the digital version is completely identical to that of the printed version.

I am aware that a false declaration will have legal consequences.

Unterschrift/Signature:

_____

Klagenfurt, 12. Dezember 2011

Dedicated to all people
who supported me
in my education,
especially to my parents.

Contents

List of Tables

List of Figures

Acknowledgements

I am deeply thankful to my advisor Prof. Laszlo Böszörmenyi for his support of my thesis. I appreciate the numerous discussions with him, his openness and his criticism, which always lead to an improvement of my work.

I would also like to thank Prof. Alan Hanjalic for his input and for supervising my dissertation.

I thank all my colleagues, with whom I have worked together in the last three years, for their help and collaboration.

Special thank goes to Martina Steinbacher for being the heart and soul of our institute.

I would also like to thank the Lakeside Labs research lab for giving me the possibility to work on my PhD in the context of the SOMA project.

Last, but not least, I am deeply grateful to all people that supported me during my education, especially to my parents.

Abstract

This thesis discusses three major issues that arise in the context of non-sequential usage of multimedia content, i.e. a usage, where users only access content that is interesting for them. These issues are (1) semantically meaningful segmentation of videos, (2) composition of new video streams with content from different sources and (3) non-sequential presentation of multimedia content.

A semantically meaningful segmentation of videos can be achieved by partitioning a video into scenes. This thesis gives a comprehensive survey of scene segmentation approaches, which were published in the last decade. The presented approaches are categorized based on the underlying mechanisms used for the segmentation. The characteristics that are common for each category as well as the strengths and weaknesses of the presented algorithms are stated. Additionally, an own scene segmentation approach for sports videos with special properties is introduced. Scenes are extracted based on recurring patterns in the motion information of a video stream.

Furthermore, different approaches in the context of real-life events are presented for the composition of new video streams based on content from multiple sources. Community-contributed photos and videos are used to generate video summaries of social events. The evaluation shows that by using content provided by a crowd of people a new and richer view of an event can be created. This thesis introduces a new concept for this emerging view, which is called "The Vision of Crowds".

The presentation of such newly, composed video streams is described with a simple but powerful formalism. It provides a great flexibility in defining the temporal and spatial arrangement of content. Additionally, a video browsing application for the hierarchical, non-sequential exploration of video content is introduced. It is able to interpret the formal description of compositions and can be adapted for different purposes with plug-ins.

CHAPTER 1

Introduction

This chapter motivates the topic of this thesis and describes its structure. Furthermore, the contributions made within this thesis are briefly overviewed.

1.1 Motivation

In the last decade a transition occurred in the way how people are dealing with multimedia content. The rise of multimedia sharing platforms on the Internet led to a significant change. In former times, only a few central authorities in the shape of TV and radio broadcasters controlled what content was offered to the viewers. The only decision that people could make was to watch or not to watch the offered program. Later, when more channels were available people were at least able to choose between the programs of different channels, but real flexibility was still missing. The schedules were defined by the central authorities and if two channels showed interesting content at the same time, people had to choose which one to watch and which one to miss.

Today, besides professional content providers, amateur and hobby producers of multimedia content entered the scene. Contents are produced anytime, everywhere and immediately shared on the Web. Everyone can be producer and consumer, provider and client - all at the same time. Additionally, people are in a position to decide what they want to watch, when they want, where they want and sometimes even in the quality they want. These circumstances lead to a non-sequential usage of multimedia content. That means that in many cases users do not watch videos or photo collections a whole anymore, but only those parts that are interesting for them.

The whole freedom and flexibility that people can enjoy nowadays introduces many additional problems and complexities for current distributed multimedia systems. In this thesis I am dealing with some of the questions that arise in that context:

- How can we extract or index the most interesting parts of videos?
- What information about a real-life event can be obtained from a crowd of people (e.g. in social network)?
- Is a richer view of a real-life event emerging from the different views (in terms of photos and videos) of different participants?
- Is it possible to tell a story or to create a summary with content from multiple users?
- Subjective opinions about the quality of a summary can be divergent to a great extend? How can the quality of automatically generated stories be evaluated in an objective way?
- How should a story consisting of several photos and videos be presented?

A tremendous amount of videos is produced and published every day. It is impossible to watch all these videos. Even if the amount of videos regarding a certain topic can be reduced by querying the titles or the tags, it remains a non trivial task to select appropriate videos and to identify interesting segments within them. The question

which parts are of interest cannot be answered in general. It depends on the requirements and the intentions of a user. A study concerning the generation of abstracts from 10 scientific documents showed that among six human judges on average only 8 % of the abstracts were overlapping [77]. Therefore, the task is not to identify the most interesting segments, but to index videos in an optimal way to make different parts easily accessible. Video segmentation on the shot level works already well, but shot boundaries do not provide a satisfactory index of videos, because especially long videos may consist of too many shots. Indexing videos on a semantic level, which means identifying scenes, seems to be more appropriate. Typically, videos consist of much less scenes than shots. As scenes contain semantically coherent content, they are more likely to be consumed together.

Indexing videos on a semantic level enables applications that present the content of videos in a clearer and more comprehensive way. In distributed scenarios popular scenes can be extracted and significant delivery time and bandwidth can be saved by distributing only them. The identification of semantic scenes is a non-trivial task. An actual implementation depends on the type of video that has to be segmented. Different types have different characteristics, e.g. the structure of a movie is different from the structure of a news video or the structure of a surveillance video.

Scenes of interest may occur in several videos, not only a single video. In my opinion, one of the current challenges in this field is finding interesting scenes in multiple videos. Different videos may show a situation of interest from different perspectives and may reveal different information of a scenery. If we go one step further, we do not even look at a single scene, but at whole real-life events. By using the term *real-life events* I consider happenings that occur in our daily life, e.g. social events like pop concerts or sports events, but also sudden incidents, like accidents or disasters.

When people visit social events they take their mobile phones or digital cameras with them and they document their personal experience. But people do not only take photos and videos for their personal memories. Today, experiences are shared with

the whole world by uploading content on the Web. This trend can also be observed for "spontaneous" real-life events. If incidents happen in our daily life more and more people use their mobile phones to report about them on social platforms.

Social media platforms for different purposes can already be found on the Internet. Figure 1.1 lists a tremendous amount of platforms and categorizes them according to their purpose[1]. Such self-organizing human communities have a big potential that is not utilized sufficiently yet. I am of the opinion that people who witnessed an event are a good source of information to report about that event. Spectators are spread all over the area where an event takes place. They can preserve all happenings from different viewpoints. Especially, if an event is distributed over a large area and not only limited to a concert hall or a sports stadium, this circumstance can become a great advantage. The crowd of visitors is everywhere at the same time. Theoretically, the crowd is in a position to capture everything of importance. In practice this assumption does not always hold, because not everyone is actually taking photos or videos, but the amount of user-generated content is growing and growing.

People can inform themselves about real-life events by looking at the content shared by participants. They get a view of an event by looking through the eyes of people that have been there. In fact, they look at the photos and videos of them. We introduce a new notion for this view that emerges from the different views of different people: "The Vision of Crowds". It is a paraphrase of the well-known notion of the Wisdom of Crowds [101]. The Vision of Crowds is a similar approach, however, the emphasis is not on new knowledge, but rather on a new view.

1.2 Contributions

This thesis investigates techniques for video segmentation, video summarization and video presentation. Within that context several contributions are made.

[1]Digital September/October 2011: `http://www.digital-zeitschrift.de/media/files/digital_zeitschrift_2011_09.pdf`

Figure 1.1: Popular social media platforms on the Internet

1.2.1 Overview and Classification of Video Scene Detection Approaches

A lot of video scene detection approaches have been published in the last decade. Most of them are targeted at movie segmentation, but also news videos, sitcoms or home videos are presented as use cases. It became hard to preserve an overview of all the presented approaches and their pros and cons. Therefore, one contribution of this thesis is a comprehensive survey of scene segmentation approaches. Two different classification schemes are used to group all presented works. First, the algorithms are classified according to the underlying mechanisms used for the segmentation. Second, several use cases for scene segmentation approaches are defined and an investigation is performed for which use cases the presented algorithms are suited for. In the end, problems that are still unsolved in this field and future challenges are identified.

1.2.2 Video Segmentation Based on Recurring Motion Patterns

Scene segmentation approaches rely on low-level features as base for the segmentation. Low-level features are features that can be extracted from the digital representation of videos and which do not directly have a high-level semantic meaning, e.g. color or motion. Regarding the used low-level features most scene segmentation approaches rely on color features and some additionally use audio features. Only a few of them incorporate the motion information of videos to identify action scenes. This leads to the question whether it is possible to use the motion information for other segmentation tasks beyond finding action scenes.

For certain types of videos, scenes correspond to a repetition of a pattern of motion directions of people and objects shown. Especially in the sports domain this can often be observed. In many sports competitions athletes are competing one after another, camera positions are fixed and camera pans and zooms follow similar patterns. One

contribution of this thesis is to take advantage of these characteristics and to identify scenes based on recurring motion patterns in sports videos.

1.2.3 The Vision of Crowds

In the focus of interest of this thesis is community-contributed multimedia content, especially in the context of real-life events. Community-contributed contents are photos, videos and their corresponding metadata that are shared by people on social media sharing platforms on the Internet. One major contribution of this thesis is an investigation how community-contributed content can be used in different scenarios to compose summaries of the events the content originates from. In the following I use the term *event summary* for such summaries that should represent the most important situations that happened during an event in our real life.

First, a live summarization approach is presented. In the context of my thesis a social event that took place at Klagenfurt University was chosen to conduct a case study. During this event visitors were encouraged to report about interesting situations by uploading photos and videos to our system. An automatic director component was developed that selects photos and videos shared by visitors and composes summaries which are immediately shown to all visitors of the event. These live summaries help visitors to keep up to date by showing hot spots of an event while it takes place.

An interactive graphical user interface for the manual or (semi-)automatic creation of event summaries has been implemented in addition to the live summary use case. It provides a useful tool to get an overview of a social event after it took place. This GUI can be used by people that missed a social event, but afterwards want to know what happened there. Presentations can be defined fully manually, but it incorporates also the automatic director component for the automatic generation of presentations, which can be modified after they have been composed.

For the first two use cases I only relied on our own system as a source of information and as data repository. The gain of the data collected during the case study is rather limited. We did not get a large amount of content. Therefore, the event

summarization approach was extended to work with content from the social media sharing platforms Flickr[2] and YouTube[3]. They are very popular and widely used. As a lot of people share their content on these platforms they enable me to create event summaries of very large events that attracted the attention of millions of people around the world. These summaries show the big potential of the Vision of Crowds. An evaluation of the collected data and the quality of the generated summaries is shown. When discussing the quality of summaries, I am especially interested in the coverage of important situations and not in the technical quality of the content. In addition, several investigations are performed how good community-contributed content is suited for reporting about real-life events and which caveats currently exist.

1.2.4 Non-Sequential Presentation of Multimedia Content

The summaries that are composed in the context of my work are not only a sequential alignment of photos and videos. One contribution of this thesis is a new and innovative presentation concept for mesh-ups consisting of photos and videos. A compact, but flexible formalism is used to describe the temporal and the spatial alignment of content on the screen.

Furthermore, a video browser for the non-sequential exploration of videos has been developed. It is able to interpret the above mentioned formal description of multimedia presentations and can be used to show photos and videos in parallel. In addition, a parallel and a tree-like hierarchical video browsing concept have been implemented. In hierarchical video browsing a video is divided into segments, which may show semantically coherent content. Each segment can be recursively divided into further sub-segments until the bottom level is reached, where a video segment is divided into its single frames.

[2]Flickr: `http://www.flickr.com`
[3]YouTube: `http://www.youtube.com`

1.3 Structure

This thesis is structured as follows. Chapter 2 describes the context of this thesis, i.e. the research fields that are covered by it. The research done during my thesis was a part of the Self-Organizing Multimedia Architecture (SOMA) project. The basic ideas and concepts behind this project are presented and the role of my contributions for SOMA is explained. Furthermore, the notion of "The Vision of Crowds" is introduced. In Chapter 3 a comprehensive survey of video scene segmentation approaches published in the last decade is presented. The algorithms are categorized in two different ways: (1) based on the underlying mechanisms for the scene segmentation and (2) related to possible use cases for the specific approaches. My own scene segmentation approach that detects scenes based on recurring motion patterns is included. Chapter 4 introduces three different approaches for the creation of multimedia presentations with content (photos and videos) from different sources in the context of real-life events. The three approaches are (1) automatic live event summaries, (2) interactive event summaries and (3) event summarization using community-contributed content. In Chapter 5 innovative concepts for the presentation of multimedia content are introduced and Chapter 6 concludes this thesis and outlines open research questions for future work. In the appendix a tabular overview of scene segmentation approaches is given.

CHAPTER 2

Context

The research work done during this dissertation contributed to the Self-Organizing Multimedia Architecture (SOMA) project [7]. In this chapter the SOMA project is introduced and all research areas are presented, which are covered within my thesis.

2.1 Self-Organizing Multimedia Architecture

In the last decade, significant changes occurred in multimedia content consumption (i.e. multimedia search, retrieval and delivery). On the one hand, the requirements of users are growing steadily regarding the semantic precision of answers to their queries. People are used to get very good answers when they use text-based document retrieval systems, like search engines on the Internet. Therefore, it seems that they expect a similar performance of multimedia retrieval systems. On the other hand the quality of the content has to fit their equipment and connection. At the moment, content distribution networks are the most common solution to realize Video-on-Demand (VoD) systems. Fixed configurations are defined in advance to distribute the content

in a client/server manner. These predefined configurations pay attention to different bandwidths or device properties.

Such predefined profiles and the distribution of content from several central instances are not sufficiently flexible. In the SOMA project we assume that in future, flexible and - at least partially - self-organizing facilities will play an important role in distributed multimedia information systems. A lot of dynamics can be observed in the behavior how people organize themselves on the Internet. Social networks and social media sharing platforms are more and more used as source of information. If people participate in such networks, they are able to control which content they consume and which not. They are not bound to central authorities anymore that offer content in a *top-down* manner, without hardly any possible intervention of the viewer. This new flexibility of the users demands for new technologies that support them with such new application patterns, but current systems lag behind the expectations.

In future, distributed multimedia systems should provide support for user communities to organize themselves. The underlying mechanisms, such as search and distribution, must also pay attention to flexible, self-organizing users. Therefore, in the SOMA system the intentions of participating users should be identified automatically in order to provide different services for different intentions. The collective wisdom of the participants is used to enhance the semantic value of metadata.

2.1.1 Unit concept

The research on self-organizing distributed systems is still in an early stage, especially with respect to multimedia information systems. It makes a great difference whether a user is interested in a whole one-hour video or only two short subsequences of it. Delivery networks should take into account such circumstances, but by now videos have the rigidness of being sequential.

In the SOMA project we propose a new concept that breaks with the sequential view of videos in favor of a more flexible view, in which videos are regarded to be

composed of a set of units [94]. A unit is supposed to be short, semantically meaningful and has a variable size and duration. Each resulting video unit gets a unique resource identifier in our network. The following definition gives a formal description of a unit.

Definition 1 (Unit) *A unit is defined by the following rules:*

1. *A unit u represents a photo p or a part of a video v, where $u \equiv p \vee u \in v$.*
2. *Let d_u be the duration of unit u and d_v the duration of video v, $u \in v \Rightarrow d_u << d_v$.*
3. *$\forall u_i, u_j$ where $i \neq j$, then $d_{u_i} \leq d_{u_j} \vee d_{u_i} > d_{u_j}$.*
4. *If $u \equiv p$, then $d_u = D$, where D is constant.*

Every video stream uploaded to our system is divided into logical and physical units of meaningful size and content. This video segmentation process is called *decomposition*. Considering size and content at the same time is of course a challenging issue. A good tradeoff has to be found. On the one hand, it should be possible to deliver units fast through the network. On the other hand, a unit should contain semantically meaningful content in order to fulfill the users' requests with as few units as possible and without presenting scenes that do not satisfy the users' information need. An ideal unit conveys video segments that are typically consumed together.

Different levels can be considered for the segmentation of videos. Figure 2.1 shows an illustration of a hierarchical video representation. For example, a video can be divided into several logical levels like key frames, shots, scenes or even groups of scenes. But also other logical hierarchies are possible. Beside several logical segmentations, at least one physical segmentation has to be performed if the content is delivered through a network.

In addition to hierarchical segmentation, we distinguish between full and partial decomposition. Full decomposition partitions a video in a way that the original video

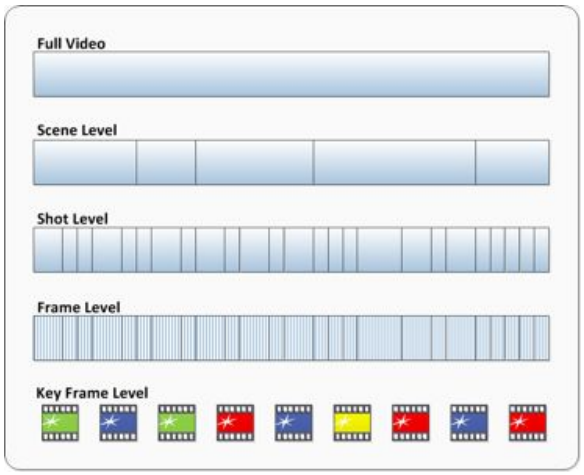

Figure 2.1: Levels of a hierarchical video representation

can be reproduced again with all units. In a partial decomposition only certain parts are extracted from a video and the rest is disregarded. The original video cannot be reproduced. This is common for surveillance scenarios, where parts of the video can be left out where nothing happens. The following definition describes the decomposition in a formal way.

Definition 2 (Decomposition) *A decomposition is described by following characteristics:*

1. *Let v be a video, then $u_1 \cup u_2 \cup \ldots \cup u_n$ is a decomposition, where $\forall u_i \in v \wedge u_i \neq u_j : u_i \cap u_j = \emptyset$.*

2. *$\forall u_i \in v$ where $u_1 \cup u_2 \cup \ldots \cup u_n \equiv v$ is called a full decomposition.*

3. *$\forall u_i \in v$ where $u_1 \cup u_2 \cup \ldots \cup u_n \subset v$ is called a partial decomposition.*

4. *If $u_1 \cup u_2 \cup \ldots \cup u_n$ is decomposition of v, then $a_1 \cup a_2 \cup \ldots \cup a_n$ where $\forall a_i \in u_i$ is called a hierarchical decomposition.*

The initial efforts in video segmentation aimed at indexing videos for non-sequential video access or for summarization tasks. Only little work has been done to examine

video segmentation issues in the context of video delivery so far. If consumers of distributed video platforms are not interested in the full content of a video, but only in certain parts of it, it is unnecessary to deliver data that is not going to be used.

2.1.2 A New Understanding of Video Streams

In SOMA users become from passive consumers of multimedia to active composers. New videos can be composed by assembling different units, which may originate from different videos, to a new video stream. In our architecture this process is called *composition*.

We do not distribute and show the available videos as a whole, but only the parts that might be important or interesting for certain users or user groups. This aggregation of video units from different sources leads to a new understanding of video streams. A formal description of the composition is given by the following definition.

Definition 3 (Composition) *A composition is defined by the following properties:*

1. *Let v_x and v_y be videos, $u_1 \cup u_2 \cup \ldots \cup u_n$ is a composition, where $\forall u_i, u_j$: $(u_i \in v_x \wedge u_j \in v_x) \vee (u_i \in v_x \wedge u_j \in v_y)$.*

2. *Let t be the timestamp of a unit, $u_1 \cup u_2 \cup \ldots \cup u_n$ is a composition, where $u_i \in v, u_j \in v : t_{u_i} \leq t_{u_j} \vee t_{u_i} > t_{u_j}$.*

3. *An original video v is a composition $u_1 \cup u_2 \cup \ldots \cup u_n \equiv v$, where $\forall u_i, u_j \wedge i < j : t_{u_i} < t_{u_j}$.*

The task of composing a new multimedia stream needs not necessarily be done manually by the users themselves. It can be delegated to software agents that compose new streams based on predefined profiles, which express similar interests or topics.

2.1.3 Architecture of SOMA

We defined a layered architecture for the SOMA system. It consists of three major layers: (1) the *Sensor Layer*, (2) the *Distribution Layer* and (3) the *User Layer*.

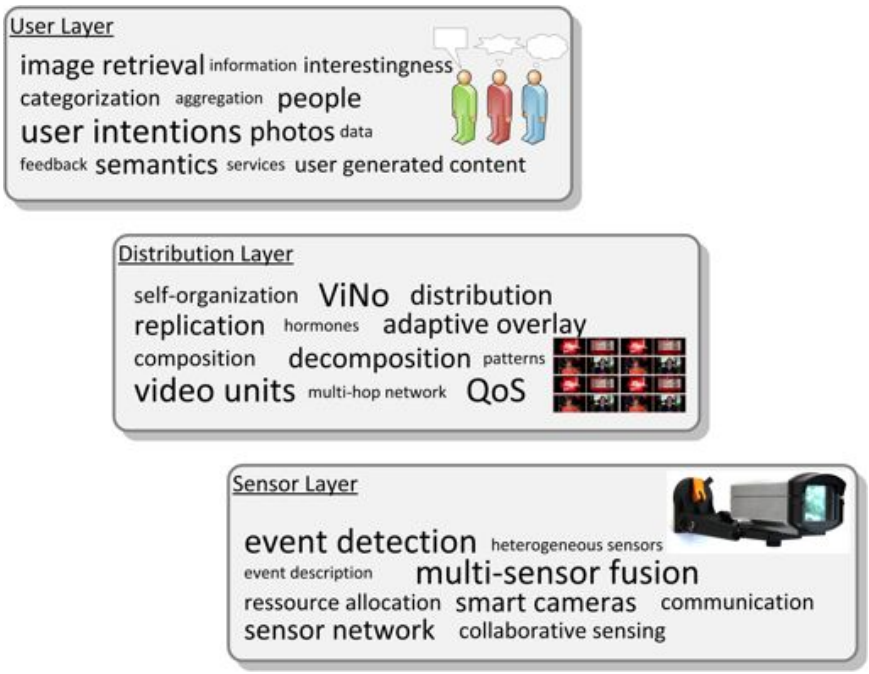

Figure 2.2: The three SOMA layers described with the most important key-words [7]

Figure 2.2 illustrates these three layers with tag clouds, which characterize each layer with several key-words.

Sensor Layer

On the Sensor Layer smart cameras are used (i.e. cameras with own processing unit and memory) to detect semantically meaningful events, e.g., the gathering of a large group of people or the drive by of a police car instead of just *a lot of movement* or *a sudden change in color*. The data of the cameras is fused with data of other sensors. As a result alerts can be generated. Information about detected events, video units showing these events and the generated alerts are passed to the Distribution Layer. The Sensor Layer cannot decide upon the relevance of an event. The semantics of events are verified on the User Layer. More details about different aspects of the Sensor Layer are given in [72][74][73].

Distribution Layer

The main goals of the Distribution Layer are the generation and efficient distribution of units in the SOMA network and the composition of new videos that come up to the intentions and needs of the users. The illustration in Figure 2.3 shows decomposition and composition in the context of the User- and the Distribution Layer. Both layers can be seen as autonomous, self-organizing areas. At the User Layer, people organize themselves in communities to tag, annotate, rate, and share content. At the Distribution Layer, video units are distributed and replicated only based on local decisions on proxies. As consumers, users are able to compose individual video streams or to watch automatically created ones consisting of units from different sources. The feedback and the estimated intentions of the users are used to improve the semantic quality of compositions and the replication of units on the network. The work presented in Chapter 3 and Chapter 4 is a contribution to the decomposition- and composition strategies of the SOMA system.

The placement of replicas is performed in a self-organizing manner. It is assumed that in creating compositions some units will become very popular, others much less. Popular units are replicated on several network nodes and moved towards interested clients. The resource management can thus be optimized for the delivery of popular units, with significant resource savings resulting from not having to deliver units, which are not of interest.

The self-organizing resource management of the SOMA system is presented in [95] [96] [97]. For both, the description of the data transport as well as for the temporal and spatial description of compositions an own formalism called *Video Notation (ViNo)* is used [94]. ViNo is a multipurpose multimedia language, which allows us to express even complex configurations in a compact way.

User Layer

On the User Layer the users' information needs should be fulfilled. People's abilities of self-organization (cp. the Wisdom of Crowds [101]) are utilized to achieve this goal.

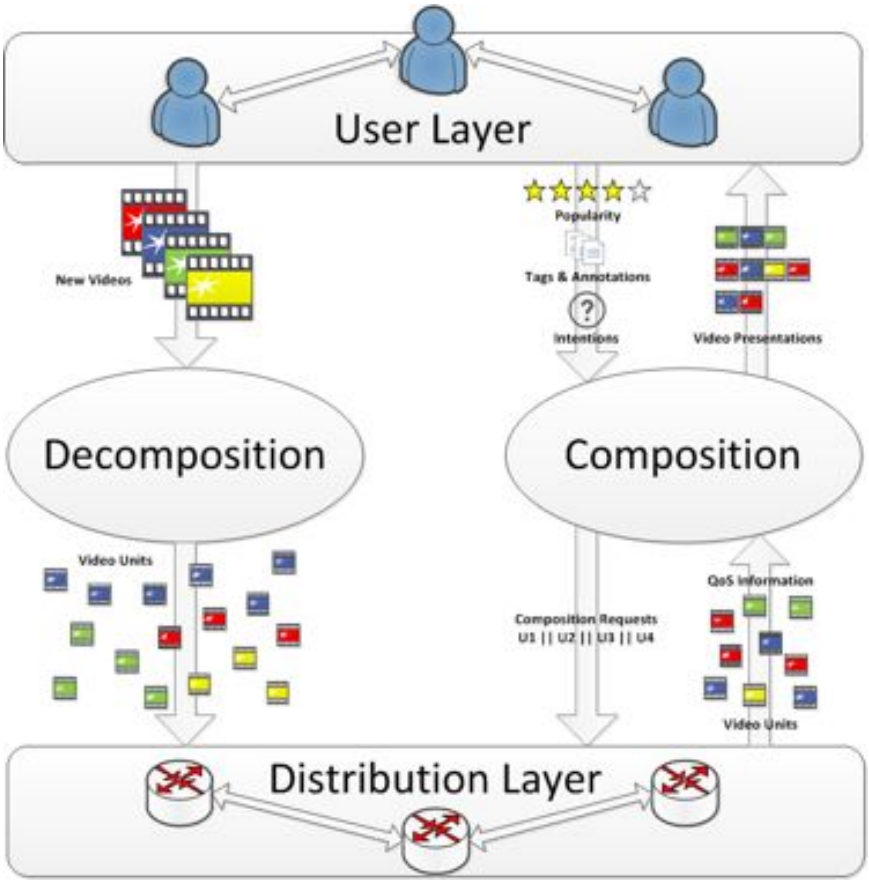

Figure 2.3: Decomposition and composition in the context of the SOMA layers [7]

New data units are produced and uploaded to the system or compositions of existing units are consumed. Besides explicit user feedback, the User Layer derives implicit feedback based on usage patterns and collaboration statistics. The self-organization of the users is supported and enhanced by emergent patterns in consumption and collaboration. The results from the user feedback (explicit as well as implicit) are used to enrich the data units and compositions by conceptual links, popularity approximations and metadata. Hints about the observed usage patterns and emergent semantics are provided to the lower levels of the SOMA system. Details about the research work regarding user intentions are given in [43][44][56][98]. Our research results regarding the presentation of individually composed multimedia streams are topic of this thesis and are explained in detail in Chapter 5.

2.1.4 Self-Organization

The global behavior of a self-organizing system is defined by local decisions of the system's entities. This concept is called *emergent behavior*. Basic elements for self-organizing distributed multimedia information systems are defined in [7].

In SOMA two levels of self-organization are present: (1) intra-layer and (2) inter-layer self-organization. Each layer organizes its components and resources to a certain extent within the borders of the layer in a self-organizing way (intra-layer). The Sensor Layer organizes smart cameras and sensors; the Distribution Layer manages storage capacities, processing power and bandwidth; and the User Layer handles the presentation of the content, the aggregation of usage patterns and the user feedback based on the self-organizing behavior of the users.

Intra-layer self-organization alone is not sufficient. Therefore, self-organization principles are also applied across different layers. For example, a smart camera (or even a set of cameras) might not be able detect a certain semantically relevant event without support from the upper layers. On the other hand, knowledge from the lower layers may impose reasonable limits at the User Layer and thus transform unrealizable requests into realistic ones. For example, if we know that a user browses pictures of

cars with no explicit, exact goal and it would take too long to download some of the requested pictures, then it seems to be a better idea to present other car pictures, still fitting the user's intention. Instead of insisting and forcing the user to wait for the download to be complete, pictures, which may potentially be equivalent with regard to the user's intention, are shown in place of the missing ones.

The SOMA system is targeted at use cases with special characteristics. These characteristics and two concrete examples of challenging usage scenarios are presented in [7]. In such scenarios production, search, access, delivery, processing and presentation of multimedia data must become much more flexible than today. In many cases these issues must be handled spontaneously. While spontaneity is of high value in everyday life, it is extremely hard to realize it in technology, which led to a series of ideas on self-organizing multimedia systems. Some of the ideas are still the subject of ongoing research.

2.2 Research Areas

The research done in the context of this thesis is related to three major research areas in multimedia. In this section an overview of these areas is provided.

2.2.1 Video Segmentation

In the last decade many different approaches for video segmentation have been proposed. The first attempts aimed at automatically finding shot boundaries within a video. Shot boundaries can be defined as the physical boundaries where camera changes happen. Many shot detection algorithms have been proposed [105, 121]. The best ones achieve a high accuracy and thus this task can be regarded as essentially solved. The problem is that on the one hand even short videos can consist of a large number of shots and on the other hand, some kinds of videos consist of a single shot (e.g. in surveillance). Therefore, it is insufficient to index a video only on the shot level. It is more likely that people do not search for a single shot, but for semantically

meaningful scenes, which may consist of several shots. Distributed multimedia systems can benefit from a semantically meaningful segmentation, because the amount of unnecessarily transmitted data can be reduced. In contrast to shots, which have a clear definition, it is much harder to define what a semantically meaningful scene is.

Several definitions how a scene is characterized have been proposed. *Hanjalic et al.*[32] describe a scene as a series of temporally contiguous shots, which is characterized by overlapping links that connect shots with similar visual content. *Sundaram et al.*[100] define a scene as a contiguous segment of visual data with a long term consistency of chromaticity, lighting and ambient sound. They use the notion of a computable scene, because all these properties can be easily determined using low-level audio and video features. *Rui et al.*[78] and *Cour et al.*[16] define a scene as a sequence of semantically related and temporally adjacent shots depicting a high-level concept or story. This description pays attention to scene definitions in film literature. In French classical theater a scene corresponds to the arrival and departure of characters [62] and the Film Encyclopedia [41] defines a scene as a section of a motion picture with unified time and space. *Truong et al.*[104] and *Tavanapong et al.*[102] distinguish between different types of scenes: (1) Serial scenes like the ones just mentioned. (2) Parallel scenes where interwoven parallel actions belong to one scene, e.g. actors in different places, flashbacks or montage techniques used by directors. (3) Traveling scenes where actors move through different places within the scene.

A lot of different approaches for the segmentation of videos into scenes have been presented in the last decade. A major contribution of my thesis is a comprehensive survey of scene segmentation approaches. It is presented in Chapter 3. In addition to the survey, I have developed and implemented an algorithm, which detects scenes based on the identification of recurring motion patterns within a video stream.

Another approach for finding recurring visual sequences in live TV broadcasts is presented in [22]. The authors use an edge feature for comparing video clips of a 24 hour live TV broadcast. The purpose is to find and document recurring commercials

throughout the TV program of one day. In contrast to that approach, my algorithm tries to find neither commercials nor identical scenes, but scenes with similar motion.

In [47] a shot detection algorithm is presented that calculates an activity level for each frame and segments the video stream into shots that contain high activity, because it is assumed that these parts are the most interesting ones within a video. As already mentioned, scenes created in such a way are rather short. With my approach longer scenes, which consist of shots that semantically belong together, are identified.

My scene detection algorithm is based on previous work of *Schoeffmann et al.* [83] on video exploration. With this interactive Video Explorer users can easily identify different motion sequences and search for similar subsequences. The motion information is extracted from compressed H.264/AVC videos and mapped to the HSV color space in order to visualize motion direction and intensity. A screenshot of the Video Explorer can be seen in figure 2.4. Below the video playback area an interactive navigation index is shown that visualizes the motion information. With the help of this index users can quickly and easily recognize semantics like fast or slow motion, the direction of the motion and camera zooms or camera pans. By selecting a motion sequence it can be searched for similar sequences throughout the whole video stream.

The limitations of the Video Explorer are that users can only search for small sequences and that recurring sequences have first to be identified by the user. Therefore, the new algorithm presented in section 3.5 can be seen as an extension for it, towards automatic, semantic segmentation.

2.2.2 Video Summarization and Real-Life Events in Multimedia

The summarization of multimedia content is target of many research projects. Most of them focus on video abstraction and video summarization. Two extensive reviews of key frame extraction and video summarization approaches are given in [63, 105]. The presented algorithms summarize single videos with selected key frames or with a short summary video.

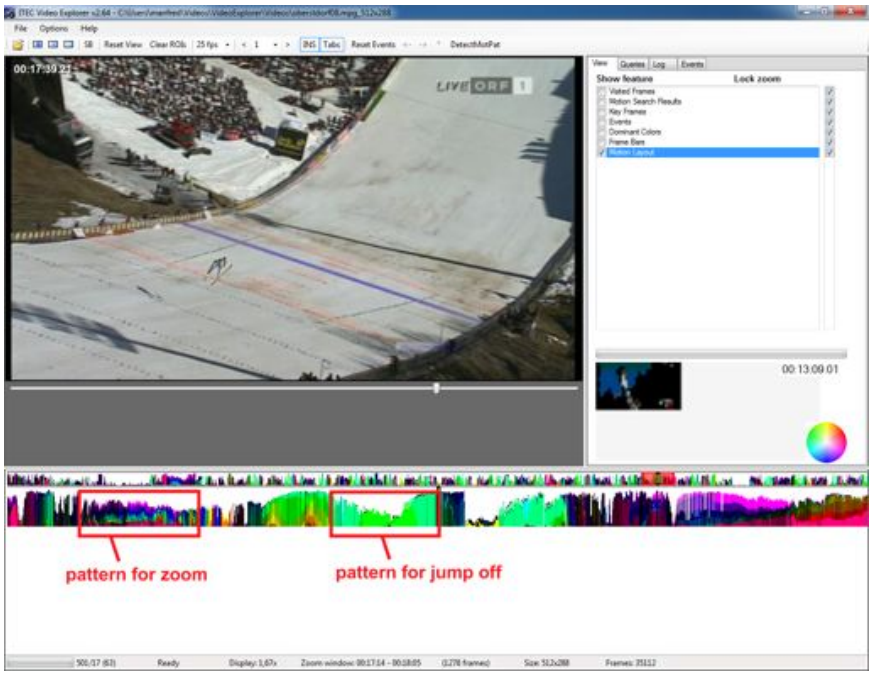

Figure 2.4: The Video Explorer showing the motion based navigation index [83]

The focus of my work is not on summarizing single videos, but on summarizing the happenings of real-life events with content from multiple social media platforms. Therefore, I would like to point out some interesting research works concerning real-life events in multimedia.

One approach, which uses multiple videos as input for a summarization algorithm, is introduced in [40]. Videos of a whole basketball season in the USA and the corresponding metadata are used to create summary videos under different aspects, like summaries of the whole championship, of only one team or even of a single player.

Not only the summarization of videos has been extensively studied. The summarization of image collections has also been a target of research activities [92]. This work defines three aspects of an effective summary and formalizes models to optimize them: (1) quality of the content, (2) diversity of the content and (3) coverage of the whole collection.

During the last few years more and more research activities were focusing on real-life events in the context of multimedia data. In [116] a common event model for multimedia applications is proposed. Eight basic aspects are defined to describe an event, but also the relationship of an event to other events. These aspects include the time and the location, the involved entities as well as the type and the structure of the event.

An event-based clustering algorithm is proposed in [59]. A layered clustering algorithm produces different clusters of videos, where each cluster represents one event. The authors state the problem of incomplete or misleading metadata. Therefore, they perform a data compensation based on the textual descriptions of the content to gather missing dates or GPS coordinates.

In [52] information from the web-based event directories Last.fm, Upcoming and Eventful is used to get metadata about an event, like the title or the geo information. With the help of this additional information the authors try to gather as much photos and videos as possible from Flickr and YouTube. Some interesting findings about the uploading tendency and the number of available photos and videos per event are given. An ontology is used to identify connections between events, media and people participating in events.

A visual-based method for retrieving events in photo collections of community-contributed contents is introduced in [103]. Based on a query image an image collection is searched for similar photo records that may be of the same event. A geo-temporal consistency score is computed for each photo record to estimate whether it shows the same event or not.

In [109] an automatic remixing approach for community-contributed content from music concerts is presented. Users can record and upload videos during live events. Afterwards, the shared content is synchronized based on the creation timestamps and a master audio track is extracted from the single audio tracks of the synchronized videos. In the end, video remixes of a concert are automatically created based on automatically detected regions of interest.

The organization of tagged photo collections based on landmark and event detection is presented in [71]. Photos are arranged on their spatial closeness and their relatedness to events. An online travel application for place exploration uses the presented concepts for the arrangement and presentation of the content.

In [28] a joint content-event model is proposed that allows an event-based indexing of videos instead of a concept-based one. It consists of a content part that models videos in terms of scenes and shots and an event model that defines different events and how they may be related to each other. A referencing mechanism based on a trained classifier is used to assign events to the content

The *Multimediaeval*[1] workshop also hosts a *Social Event Detection* benchmark. The task is to find all content related to a certain event in a large repository of community-contributed multimedia data.

A work that is not event-centric but that shows the power of utilizing community-contributed content is presented in [91]. Images of online photo collections are used to generate 3D views of famous places in the world where a lot of photos are taken. The introduced application allows an exploration of places based on the content of people that have really been there.

2.2.3 Video Browsing

Video browsing is an appealing approach to find out whether a video or some parts of it are of interest and where the most interesting segments are located within a video. Many efforts have already been made in this field and several tools were introduced in the last decade. While some of them try to improve navigation with extended timeline sliders (e.g. [23][36]), others show content abstractions that can help users to more quickly locate desired segments [6][88]. Some other tools facilitate browsing by an index of extracted key frames, typically at different levels of granularity (e.g. [9]), or by providing smart fast-forwarding features (e.g. [14]).

[1]Multimediaeval Benchmark: `www.multimediaeval.org`

Furthermore, many investigations are performed to find out how to organize big video archives in a hierarchical way to make the contents better accessible. However, it exist only few video browsing tools that support hierarchical navigation in videos. Using hierarchical browsing mechanisms, video content can be displayed in a well-arranged way, helping users to get a better overview of the relations between certain video segments and to find searched segments faster.

A key frame based video browser has been introduced in [31]. Videos can be browsed on three levels (scenes, shots and key frames). A hierarchical video representation based on a tree structure has been presented in [38]. The browsing hierarchy is illustrated by static key frames that can be clicked on to navigate through different levels. A more dynamic video browsing tool is introduced in [27]. On each level of the browsing hierarchy a rapid serial visual presentation (RSVP) carousel interface [117] can be used to scroll through the key frames of that level. A cone-tree-like representation of key frames has been presented in [60]. This approach provides already a 3-dimensional interface for the navigation through a hierarchy of key frames.

A comprehensive review on video browsing applications can be found in [86]. In this thesis a video browsing tool is presented in section 5.2.1, which also provides a tree-like browsing mode, but this tool displays video segments that can be played in parallel instead of key frames only.

CHAPTER 3

Decomposition of Multimedia Content

One fundamental assumption of my thesis is a shift in the usage of video content. In many cases people do not want to get those parts of videos that are not interesting for them. Motion pictures are usually watched as a whole from the beginning to the end. On social media sharing platforms on the other hand, people only share important parts of videos, e.g. certain reports from a news broadcast, parts of a TV program or selected clips taken with mobile phone cameras. This usage scenario applies even more to professional usage scenarios. In traffic surveillance nobody wants to watch all the content produced 24/7, but only scenes where something important happens, like a traffic jam or an accident.

The focus of this chapter is on how to identify semantically meaningful scenes in videos effectively and efficiently. A lot of work has already been done in this field in the last decade. For researchers and engineers starting in the field of video scene segmentation it became hard to overview the variety of presented approaches. No other survey of scene segmentation approaches has been presented recently. To the best of my knowledge the last one was presented approximately ten years ago [108].

Therefore, I performed a new, comprehensive survey of existing approaches. It provides a categorization of algorithms according to the underlying mechanisms used for the segmentation. Additionally, six possible application scenarios and algorithms that can be applied to them are introduced. I consider even new domains, which the authors of some algorithms originally did not take into account. Furthermore, current challenges for scene segmentation research are formulated.

At the end, an own video scene segmentation algorithm is introduced, which has been implemented and evaluated in the context of this thesis. It is solely based on the motion information of videos and identifies scenes based on the most frequent motion pattern within a video. It is especially targeted at the sports domain and special characteristics that many sports videos show. As existing scene segmentation approaches do not consider these characteristics I created an algorithm that is better able to cope with them.

Throughout this chapter I refer to the terms and concepts for the decomposition of multimedia content, which are introduced in section 2.1.1.

3.1 Video Scene Segmentation: State-of-the-Art

Basically, the scene segmentation approaches presented so far can be divided into seven categories: visual-based, audio-based, metadata-based, graph-based, statistics-based, film-editing-rule-based, and hybrid approaches. These categories are not all on the same abstraction level. Three different levels can be identified. Figure 3.1 shows how the categories are related to different levels. Low-level approaches are focused on visual, audio or textual features of videos. On the second level the solutions are based on a certain algorithmic principle, like graph algorithms or statistical methods. High-level approaches are building a conceptional model and try to segment videos based on that. I decided to use these different abstraction levels for my categorization to remain compatible to the original focus of the presented works.

Figure 3.1: Abstraction levels for scene segmentation approaches and the six classes used for the categorization

3.1.1 How to Use This Survey for Further Research?

With this survey I do not only want to give a broad overview of the approaches published in the last decade. I also want to provide a guide that should help researchers and developers as starting point how to tackle specific problems.

The categorization according to the methods used for the scene segmentation (visual-based, audio-based, metadata-based, ...) in section 3.2 helps to choose an appropriate strategy with respect to characteristics of the videos, the available toolkits or libraries and the knowledge of the researchers involved. Visual analysis based on color features is e.g. not suitable for black-and-white-movies. On the other side, researchers that have a broad knowledge of visual image retrieval are more likely to use a visual or a hybrid approach instead of an audio-based one. Audio-based approaches can obviously only be used, if an audio stream is available. It is hard to compare the results of the different algorithms, as different scene definitions and also different evaluation strategies are used. While some approaches evaluate with exact scene boundaries, *Hanjalic et al.*[32] regards a detected scene boundary to be correct, if it is within three leading or following shots of adjacent scenes.

The categorization according to use cases (movies, news, sports videos, ...) in section 3.3 helps to find a suitable scene segmentation approach according to the scene characteristics of the videos that should be segmented.

At this point, I define two practical examples for scene segmentation problems and later I am going to refer to them when describing the different scene segmentation

solutions. The first example is a full decomposition approach for the segmentation of movies. The required task is to identify all scenes within a given movie. I refer to it as *Example 1*. It is assumed that not enough example movies are available to be able to create a discriminative test set. Metadata are not available either.

The second example is a partial decomposition problem in the medical domain. A doctor is capturing videos of his arthroscopic surgeries. Afterwards specific scenes must be extracted to make a report and a tool that supports this process (semi-) automatically would be helpful. The videos of the surgeries have special characteristics. They usually consist of one single shot, have no audio stream and no metadata are available. In the following parts of this chapter I refer to this example as *Example 2*.

3.2 Survey of Scene Segmentation Approaches

A survey of segmentation methods was already presented by *Vendrig et al.* [108] in 2002. They focus only on visual segmentation methods for movies and TV series. Scene segmentation algorithms are analyzed according to the comparison method and the temporal distance function. A classification framework has been defined to categorize them into four classes. The advantages and disadvantages of the different classes are discussed.

This survey presents also algorithms using other modalities than visual features alone and additionally new approaches are introduced that have been published since then. The ordering of papers is as follows. In each category the papers are sorted chronologically by the year of publication. But if a work uses the same principles as a paper published before, it is presented immediately after the first paper that introduced that idea, regardless of the publication date. The characteristics that are common for each category as well as the strengths and weaknesses of the presented algorithms are stated.

3.2.1 Feature Level

Scene Segmentation Based On Visual Features

Video scene segmentation approaches that only rely on visual features and do not use graph algorithms, statistical methods or film grammar are summarized in this category. Visual features are color, motion and texture.

One of the first video scene detection approaches has been introduced by *Rui et al.*[78]. For all shots of a video the first and the last frame are selected as key frames. Similar shots are clustered based on color histograms and an activity measure. A time-adaptive grouping algorithm ensures that two shots do not belong to the same group if they are too far apart. Overlapping groups of shots are merged for the scene construction.

A single pass algorithm has been presented by *Hanjalic et al.*[32]. It works in the compressed domain of MPEG videos. For every shot several key frames are extracted. A block-based measure in the LUV color space is used to calculate the similarity of shots. An overlapping links method is used to detect scene boundaries. Shots that are visually similar are connected with links. If different groups of shots have overlapping links, all affected shots are merged to one scene. The example in Figure 3.2 shows overlapping links between three groups of shots. The circles represent shots or respectively key frames of them. The first group of shots is connected with a dotted line, the second one with a solid line and the third one with a dashed line. At the end of shot k_3 no more overlaps are recognized, therefore, this shot marks a scene boundary, called "logical story unit" (LSU) in the example.

Kwon et al.[46] try to improve this algorithm. Motion features are used in addition to color features for the shot similarity function. An improved overlapping links method has been developed that needs fewer shot comparisons to identify the overlapping links. A post processing step is performed at the end. Neighboring scenes with a duration under a certain threshold are merged to one scene.

Wang et al.[113] also use color and motion features for measuring the similarity of shots. Their overlapping links method for scene detection uses forward and backward

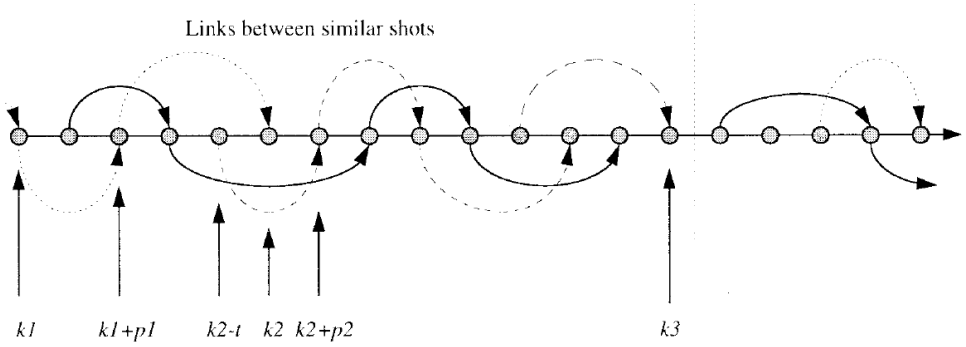

Figure 3.2: Overlapping links example [32]

search. To avoid scenes with less than three shots they introduce a post-processing step where such scenes are merged with the previous or the following scene according to visual similarity. The same method is also used by *Mitrovic et al.*[61] for the segmentation of artistic archive documentaries. Instead of color histograms and motion features they use block-based intensity histograms, the edge change ratio and SIFT key points for their similarity function.

Similar to overlapping links based methods are algorithms that use sliding windows for the scene segmentation. *Zhao et al.*[127] compare shots based on visual features and their temporal distance. A sliding window that covers a fixed number of shots is used to merge similar and temporal close shots into one scene. *Cheng et al.*[115] also use a sliding window for scene detection. The similarity of shots is calculated using HSV color histograms.

Lin et al.[50] propose a scene segmentation approach based on force competition. For each shot a dominant color histogram and a spatial structure histogram are calculated. Two different forces have been defined for the scene detection: a splitting force and a merging force. The splitting force compares each shot with its three ancestors and successors and indicates the difference to the previous shots. The merging force indicates the coherence of one shot with its three following shots. An ideal scene boundary is detected, if the splitting force reaches its maximum and the merging force reaches its minimum. In practice this is not always the case, thus

two additional rules have been defined for the scene boundary detection: (1) if the splitting force reaches its maximum, the merging force must be under a threshold and (2) if the merging force reaches its minimum the splitting force must be above a threshold to identify a scene boundary.

A scene detection method for Hollywood movies has been introduced by *Rasheed et al.*[76]. First, a HSV color histogram is calculated for all shots. Based on this information the authors introduce a new measure for the detection of scene boundaries called "Backward Shot Coherence". Each shot is not only compared with one other shot, but with the last N shots. N has to be defined in advance. If this measure is below a threshold, the corresponding shot is regarded to be at a scene boundary. Observations showed that this approach leads to over-segmentation, especially for action scenes. Therefore, a second step is performed. In addition to color, the shot length and the motion content are estimated for all shots. Short adjacent shots with high motion are merged to one scene.

Odobez et al.[69] cluster shots based on RGB color histograms and temporal closeness using a spectral clustering method. This approach is targeted at home videos and considers special characteristics of such videos. Home videos typically do not follow a storyline and they consist only of few shots, but certain rules of attention focusing can be used for the identification of scenes. Spectral clustering for scene detection is also used by *Chasanis et al.*[10]. Shots are grouped based on visual similarity (HSV color histograms). Each shot gets a cluster label assigned and in the sequence of labels patterns are identified. Scene changes are considered to occur at those points where the pattern changes. For the pattern recognition the Needleman-Wunsch algorithm [64] is applied, which is commonly used in bioinformatics to align protein or nucleotide sequences.

A motion-based video representation for scene change detection has been introduced by *Ngo et al.*[65]. Spatio-temporal slices are computed that express the motion within a video. Figure 3.3 shows an example of such a slice. An estimation is performed whether multiple motion (camera + object) or whether only camera motion

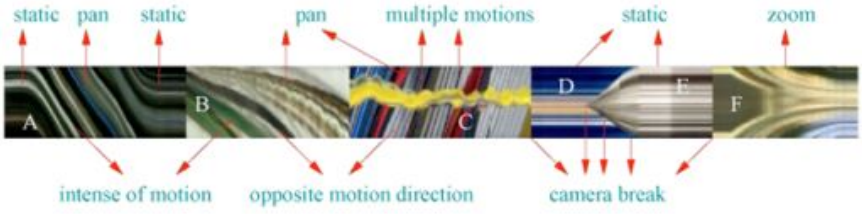

Figure 3.3: Patterns in a spatio-temporal slice [65]

is present in a shot. Key frames are extracted for shots that contain only camera motion. For shots that contain both motion types a background image is extracted. It is assumed that the dominant motion layer corresponds to the background region of a shot. Color histograms are computed for the key frames and the extracted background images. Similar shots are clustered using histogram intersection and a sliding window is used to perform a time constrained grouping of shots to scenes.

Another motion-based approach is presented in section 3.5 of this thesis. In contrast to the previous solution this algorithm solely relies on motion features. Instead of performing a shot detection the video is partitioned into segments of similar and coherent motion. At the end the most frequent pattern of motion segments is identified and scenes are extracted that correspond to that pattern. Therefore, it is a partial decomposition approach. It can be used for videos where the same motion patterns can be recognized again and again. For example, in the sports domain repeating scenes such as the start or the arrival of athletes can be identified with high accuracy. One problem of this algorithm is that it slightly over-segments videos.

Scene segmentation approaches that rely only on visual features are often tuned for a special purpose or a special domain. They often do not achieve good results when they are applied to a test set which is different from the one used by the authors. In general, it is very difficult to link low-level visual features to high-level semantics. Therefore, many authors are focusing on a limited domain to achieve a high accuracy. It is possible to apply visual-feature-based approaches to both examples from the introduction. For Example 2 only those algorithms can be used that do not rely on shot detection first. Or an alternative strategy for this first segmentation step must

Feature	**Discrimination of**
Zero-crossing Rate	Speech/Music
Short-Time Energy	Speech/Music
Spectrum Flux	Speech/Non-Speech or Music/Environment Sound
Band Periodicity	Music/Environment Sound
Noise Frame Ratio	Music/Environment Sound
Linear Spectral Pairs	(Noisy) Speech/Music or Speaker Identification

Table 3.1: Audio feature comparison [54]

be found, because endoscopic videos consist of a single shot. An overview of the presented solutions is given in Table A.1. The column "Domain" lists the domains that have been used by the original authors for their evaluations.

Audio-Based Scene Segmentation

In this subsection scene segmentation approaches are summarized that only rely on analyzing the audio stream(s). Not all of them aim at finding scenes in videos. Two approaches only focus on auditory scene segmentation, however, these ideas are very interesting. In [54] an analysis of audio features is presented. I summarize their findings in Table 3.1. In general, audio features are used to classify audio segments into speech, non-speech, music or environmental sound. Moreover, it is possible to identify speaker changes.

An audio-based segmentation approach that identifies semantically meaningful scenes has been proposed by *Lu et al.*[53]. They present a solution inspired by common video segmentation algorithms. Based on low-level features an audio stream is divided into audio segments and for each segment key audio elements are identified. A semantic affinity measure has been defined to calculate the affinity between all pairs of audio elements. Two audio elements have a high affinity, if they usually occur together and if only a short time interval is between them. Elements that have a high affinity are grouped in a scene.

A framework for the segmentation of racquet sports video has been introduced by *Liu et al.*[51]. Based on six audio features audio segments are categorized into four classes: ball impact, cheer, silence and speech. The classification is done using a Support Vector Machine (SVM) with a small training set. With the help of the identified audio classes rally scenes are identified. The discrimination of video segments into rally and non-rally segments can be used for a partial decomposition of video content. Although a SVM is used, this algorithm is not classified as statistics-based approach. The SVM is only used for classification of audio segments, but not for the segmentation process.

"Joke-o-mat" – another domain specific approach for the segmentation of sitcoms has been presented by *Friedland et al.*[25]. The audio track is segmented into chunks of fixed size and each chunk is classified. The classifier has been trained to distinguish different audio classes like actors, laughs, music and other noise. A rule-based system has been developed that transforms the detected segments into scenes that reflect narrative themes. Such themes can be dialog elements or punchlines. This system achieved already great success. Joke-o-mat was the winner of the ACM Multimedia Grand Challenge 2009 [1].

Niu et al.[68] introduced semantic audio textures, which are semantically consistent chunks of audio data. First, Gaussian Mixture Models (GMM) are trained to identify basic audio classes. Then these basic segments are merged according to predefined audio textures and with the help of genre-specific heuristics, e.g. for commercials and sitcoms.

The advantage of audio features is that they are less expensive to compute than visual features. The problem is that it is not a trivial task to detect semantic classes in an audio stream and to identify a scene structure with those classes. The accuracy of video segmentation approaches that rely solely on audio features is lower than that of algorithms based on visual features. Therefore, I conclude that visual features should additionally be used where they are available to improve the results. In Example 1, for instance, both data are available and thus a hybrid approach using video and audio

should be preferred for the movie segmentation. Audio-based approaches cannot be used in Example 2, because no audio stream is available in endoscopic videos. Table A.2 shows an overview of audio-based segmentation approaches.

Metadata-Based Scene Segmentation

This category lists approaches that rely only on the metadata of videos for the segmentation process. Although for certain videos a lot of metadata is available, especially for professionally produced ones, I only found one approach that is solely based on metadata. In all other cases metadata is used in addition to other features.

Cour et al.[16] introduced an approach for recovering the scene structure in movies and TV series. The screenplay and closed captions are used to parse a movie into a hierarchy of shots and scenes. Both sources of information are available for the majority of movies and TV series produced nowadays. The screenplay narrates the actions and the scenery and provides a transcript of the dialogs, while the closed captions provide timestamps of the dialogs. The screenplay is aligned to these timestamps and then it is parsed into elements of the type narration, dialog or scene-transition using a simple grammar. If metadata had been available in the two examples, it would have been a good choice to use it for the scene segmentation. But in none of them metadata are available and this approach cannot be used. Table A.3 summarizes the characteristics of this approach.

Using metadata leads to a high accuracy in scene segmentation as it is fast and easy to process this information. Where it is available it should be used. Of course, it can be combined with other segmentation methods, but it should always be examined if such a combination boosts the results. If metadata alone leads to a high accuracy and it is not possible to achieve major improvements with additional methods, it is better not to use them, because they may need much more resources in contrast to metadata-based algorithms alone. In TRECVID 2010 [93] we observed such a situation during the evaluation of the Known-Item-Search (KIS) task. Our automatic

retrieval approach only relying on metadata reached the third place. Extending it by visual analysis led actually rather to a slight decrease of accuracy.

3.2.2 Algorithmic Level

Graph-Based Scene Segmentation

Since the early days of video scene segmentation graph-based approaches have consistently been used in this field of research. All presented algorithms have in common that videos are first divided into shots and then these shots are clustered and arranged in a graph representation. Figure 3.4 shows an example of a graph-based approach. Nodes represent shots or clusters of shots and edges indicate high similarity or temporal closeness between the connected nodes. By applying graph segmentation algorithms, the constructed graphs are divided into subgraphs, each representing a scene. The solutions presented in this section differ only in the way how the similarity is calculated or how the graph is partitioned.

One of the very first algorithms that were generally proposed for video scene segmentation was introduced by *Yeung et al.*[120]. After the shot detection, color and luminance information are used to perform a time constrained clustering of the shots. If the temporal distance between two shots is too large, the shots do not belong to the same cluster, even if they were visually similar. Then a scene transition graph is built from the results of the clustering. Each node represents a cluster and edges indicate the story flow from one cluster to the next. Finally, the scene segmentation is done by detecting cut edges in the scene transition graph. A cut edge is an edge that separates two story units. The graph is partitioned into disjoint subgraphs by removing all cut edges. The results show that this algorithm suffers from over-segmentation. It produces too small story units. In many cases more than one story unit belongs to the same scene.

Sidiropoulos et al.[90] propose two improved versions of the original scene transition graph approach [120]. One approach improves the results with a speaker-based post-processing step. If one speaker can be identified in two connected nodes, they

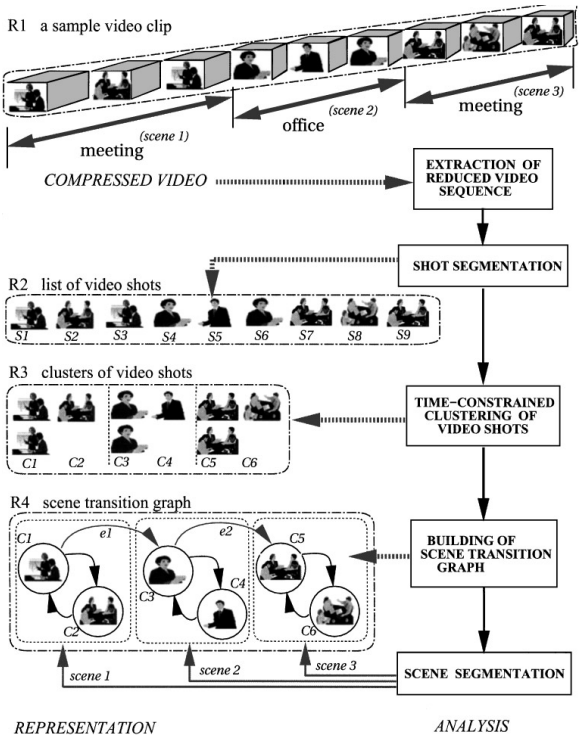

Figure 3.4: Illustration of the graph-based method [120]

are merged. The second approach builds an audio-visual scene transition graph. In addition to the visual graph, an audio-based graph is constructed. The audio stream is segmented according to speaker changes and background conditions. At the end the results of both graphs are merged to detect scenes.

Another graph-based approach has been introduced by *Ngo et al.*[66], [65]. This algorithm also clusters the shots based on visual similarity and temporal distance. First, shots are detected and represented with a graph where each node represents a shot. The edges indicate visual similarity. The Normalized Cut (NCut) algorithm [89] is used to recursively partition the graph into subgraphs. The NCut minimization problem is transformed into solving a standard Eigensystem. This leads to a set of disjoint clusters of shots. A temporal graph is built where each node represents a cluster and edges indicate transitions from one cluster to another. For example, if shot S_i is in cluster C_j and shot S_{i+1} is in cluster C_k, there is a transition between C_j and C_k. After the temporal graph has been created, the shortest path in the sequence from the cluster with the first shot to the last cluster with the last shot of the video is identified. Two assumptions are made to identify a scene. (1) Each scene contains at least one cluster that is located on the shortest path from the first to the last shot. (2) Two different scenes are only connected by one edge. Therefore, an edge between two clusters C_1 and C_2 is removed from the graph, if no path exists that traverses from C_1 to C_2 and vice versa. The algorithm has been tested with cartoons, commercials and home videos.

Lu et al.[55] use a similar approach for their video summarization algorithm. For the partitioning of shots they also use the NCuts algorithm presented in [66]. For the scene segmentation they do not use graphs, but so called "shot strings". A shot string defines a pattern how different shots are grouped in a scene and whether they are repetitive or not. Scenes are detected by identifying such patterns in a sequence of shot clusters.

Rasheed et al.[75] introduce their Shot Similarity Graph. Edges indicate the likelihood that connected shots belong to one scene. This likelihood is expressed by a

weighted shot similarity function. The similarity is calculated by comparing HSV color histograms and the motion content of two shots. The additional weight is a decreasing function of the temporal distance between two shots. It is influenced by a constant temporal decreasing factor. With the help of this weight it should be avoided grouping not temporally close shots to one scene. Once the graph has been created and all weights are assigned, a recursive bipartitioning of the Shot Similarity Graph is done using NCuts. In contrast to the former approaches, NCuts are not used for the clustering of the shots, but for identifying scenes in the Shot Similarity Graph. The evaluation with two different ground truths (DVD chapters and a manually obtained ground truth) shows that the proposed algorithm results in strong over-segmentation of the videos.

This algorithm has also been investigated by *Zhao et al.*[128]. They found that the temporal decreasing factor should not depend on a predefined constant factor, but it should depend on the number of shots in a video. Therefore, the weighted shot similarity function has been adapted to pay attention to this fact. Compared to the method in [75] the improved version of the algorithm achieves a significantly higher precision, thus it produces less over-segmentation.

Video scene detection using dominant sets has been introduced by *Sakarya et al.*[79]. They propose a tree-based peeling strategy using dominant sets for scene segmentation. The idea of this approach is to perform a step-wise partitioning of the shot graph. In each step only two sets are created: the dominant and the rest. Figure 3.5 illustrates this concept. The shots in the dominant set have a high similarity to each other and a high dissimilarity to all other shots. The similarity measure has been taken from [75]. The dominant set forms a scene. Then the algorithm is recursively applied to the remaining segments of the video. As a result, one scene is identified in each iteration.

Sakarya et al.[80] also present a two-level graph-based segmentation approach. In the first step the shot boundaries are detected and for all pairs of shots a similarity

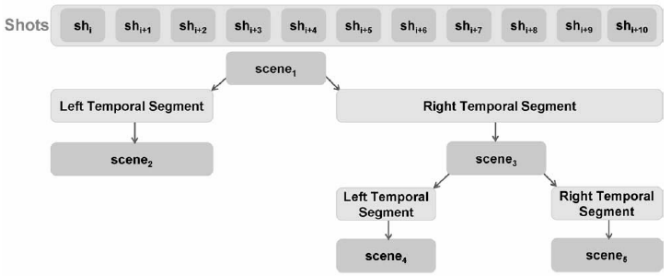

Figure 3.5: Dominant sets method [79]

matrix is calculated. Color histograms, motion information, shot duration and temporal closeness are used for the similarity measurement. A graph is built according to the matrix. Nodes represent shots and the edges indicate the similarity between shots. By using the Normalized Cuts criterion the graph is partitioned into clusters of similar shots. Shots that belong to the same cluster are merged. In the second step the algorithm calculates a similarity matrix for the clusters created in the first step. Normalized Cuts are used to partition the resulting graph again. The partitions identified in the second phase represent video scenes.

Another graph representation based on a shot similarity matrix has been proposed by *Zhang et al.*[126]. The similarity is calculated based on HSV color histograms. The graph is partitioned into scenes using a spectral clustering method.

RoleNet, a movie segmentation algorithm that takes social relationships in motion pictures into account, has been introduced by *Weng et al.*[114]. A face recognition algorithm is used to identify different characters in a movie and leading roles are detected. If two characters occur in the same shot, a social relationship between them is considered. A graph representation is used to model all social relationships in a motion picture. An illustration is shown in Figure 3.6. Each node represents one actor and an edge between two actors indicates that both occur in one scene. Weighted edges are used to express in how many scenes two actors occur together. In human perception story boundaries are often derived from the interactions between

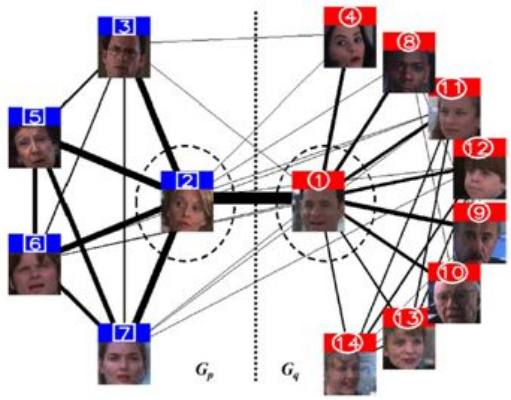

Figure 3.6: Example of RoleNet with 2 main characters [114]

characters. This algorithm pays attention to this fact and segments videos according to characters in a common context.

An approach using multiple graph representations for scene segmentation is presented in *Sakarya et al.*[81]. While other solutions combine multiple features to create a single graph, this algorithm creates an own similarity graph for each feature. In each graph the shots are clustered into scene candidates. At the end the detected scene boundaries of all graphs are merged in only two clusters: (1) scene boundary and (2) no scene boundary. Two different clustering algorithms have been examined: k-means and dominant sets. Both clustering algorithms identify too many scene boundaries. Therefore, a simple elimination step is inserted. If multiple adjacent shots are considered to be scene boundaries, only the middle shot of that sequence is marked as scene boundary. All shots before the middle shot are added to the previous scene, all following shots to the next scene. Experiments show that both clustering approaches lead to similar results.

Video scene segmentation using graph-based solutions works best for restricted environments. Especially for videos with always repeating types of scenes, like news broadcasts or talk shows. The accuracy for motion pictures is lower. Such videos have dynamic environments and directors rely on different camera techniques and effects to trigger certain emotions of the audience. It is more difficult to create a scene graph

for them. Nevertheless, many authors tested their algorithms with motion pictures. As a result, the evaluations generally show over-segmentation. Especially dynamic scenes, like action scenes, are over-segmented. Improvements have been proposed, but over-segmentation still remains a problem. Therefore, I do not recommend it for the movie segmentation in Example 1. The endoscopic videos in Example 2 only consist of a single shot. All presented graph-based approaches rely on shot detection first, thus they are not applicable to the problem in Example 2. An overview of the presented graph-based solutions is given in Table A.4.

Statistics-Based Scene Segmentation

This category consists of video scene segmentation approaches that try to express the boundary detection problem using statistical models. An optimal solution is approximated by maximizing the probability of the estimated scene boundaries to be correct.

Video scene segmentation using Hidden Markov Models (HMM) is presented by *Huang et al.*[34]. Three different approaches have been investigated. For all experiments they relied on RGB color histograms, 14 audio features and 16 motion features. For the training as well as for the evaluation five content-classes were used: commercial, live basketball game, live football game, news and weather. From each class over 10 minutes of video were used as training data. First, two different two-pass algorithms were evaluated. In the first pass both solutions segment a video into shots and calculate the likelihood for each shot to be a scene boundary. In the second pass the overall likelihood for the entire sequence of shots is optimized using two different approaches: a class transition penalty and a maximum segment constraint. The class transition penalty approach assigns to each detected boundary a penalty value, thus weak boundaries are eliminated. The maximum number of scenes has to be defined in advance and the algorithm eliminates scene boundaries to reach this constraint. The evaluation showed that it is difficult to define an accurate penalty value or maximum segment constraint in advance. The third algorithm that has been evaluated is a

one-pass algorithm. It tries to identify scene changes on the shot level. The optimal state sequence and scene class determination is performed only in a single step.

Xie et al.[118] use HMM for the segmentation of soccer videos. They extract the dominant color ratio and motion intensity from the compressed domain of MPEG videos. A single-pass algorithm is used to classify soccer videos into play and break segments. A manually labeled training set is used to train the HMM.

Another HMM-based scene segmentation and classification approach was proposed by *Yasaroglu et al.*[119]. They use face detection, audio classification (speech, music and silence), location change analysis and motion as features for their scene detection approach.

Vinciarelli et al.[110] use social network analysis in conjunction with HMM for broadcast news story segmentation. The algorithm does not take the content of videos into account, but social relationships between the persons that are involved in the news. The audio stream is analyzed to identify and cluster speaker segments. Affiliation networks are used to assign speakers to events. Finally, Hidden Markov Models are used to map social relationships into stories.

An evaluation using different learning algorithms for story boundary segmentation was introduced by *Hsu et al.*[33]. They analyzed the performance of a Maximum Entropy approach, Boosting algorithms and Support Vector Machines (SVM). For the training and the evaluation multi-dimensional feature vectors (35 or 195 dimensions) were extracted from the videos using multiple features. The evaluation showed that SVM outperformed the other two approaches.

Goela et al.[29] also rely on SVM for scene change detection. They use 12-dimensional feature vectors. Audio features are predominantly used (beside video features), as they are not computationally expensive. Each second of audio is classified into one of the four classes music, speech, laughter or silence. A binary SVM classifier is used to classify the video shots and audio segments into scene and non-scene boundaries.

Video scene segmentation using Markov Chain Monte Carlo (MCMC) was proposed by *Zhai et al.*[124]. Scene boundaries are initialized at random locations and updated using diffusion and jumps. Updating the boundaries of adjacent scenes is called diffusion. Merging two scenes or splitting an existing scene is a jump. MCMC is an iterative approach. Therefore, diffusions and jumps are applied to the video in each iteration in order to maximize the probability for the estimated scene changes.

Gu et al.[30] model scene segmentation as energy minimization problem. Shots are extracted from a video based on color features. For each shot its content and context energy are calculated. The content energy represents the energy of the shot itself. The context energy indicates the influence of neighboring shots. The scene segmentation is performed finding a global minimum in the content energy function using the Expectation Maximization algorithm. To avoid over-segmentation a probabilistic voting algorithm decides whether an identified boundary is really a scene transition or not.

Statistics-based approaches can be very powerful and achieve a high accuracy. A lot of data is needed in advance to build the statistical models or for the creation of training sets. For the movie segmentation in Example 1 it was defined that not enough training samples are available. Therefore, statistics-based methods should not be used in that example. If the training data is not carefully selected, the algorithms may not achieve accurate results. If multiple features are used, it must be carefully evaluated how these features can be best combined. In Example 2 a lot of videos are already available. A subset of them can be manually labeled and used as training set for statistics-based approaches. They are a good choice for single-shot videos if a good training set is available. An overview of the presented statistical methods is given in Table A.5.

3.2.3 Conceptional Level

Film-Editing-Rule-Based Scene Segmentation

In professional movie production directors typically rely on certain rules when they create video scenes. These rules are often referred to as "film grammar" or "film-editing rules". Several video scene segmentation approaches that have been proposed in the last decade consider these rules. The following rules are often referred to [102], [24], [104], [12]:

- **180 degree rule:** an imaginary line is used to position the cameras. Figure 3.7 illustrates this concept. All cameras are only located on one side of the line, capturing the scene always from the same side. The context in the background of the scene gets preserved.
- **Action matching rule:** the motion direction should be the same in two consecutive shots that record the continuous motion of an actor.
- **Film rhythm rule:** the number of shots, the regularity of sounds and the motion within shots depict the rhythm of a scene. Within one scene the rhythm should not change. In most cases a fast rhythm indicates the presence of an action scene.
- **Shot/reverse shot rule:** a scene may consist of alternating shots. A typical example is a dialog between two persons. The camera moves between the two characters while they are talking. But also alternating shots between people and objects of interest are possible.
- **Establishment/breakdown rule:** when a scene is established the location of the scene is introduced and all involved characters, objects and their spatial relations are shown in an overview shot. The breakdown shots are close-ups that go more into detail. They are often described using the shot/reverse shot rule.

Of course, film grammar alone is not sufficient to extract scenes. In addition to these rules low-level features are needed, nevertheless, they are not classified as hybrid approaches. The main idea of them is to segment according to film-editing rules, thus deserving an own category.

Adams et al.[2] extract expressive elements from motion pictures based on film grammar. In particular, they take advantage of the fact that film makers often use a different tempo for adjacent scenes. Especially scenes with high tempo are never aligned together in order not to confuse the audience. Tempo is influenced by the shot length and the motion within a shot. In four rounds edges are located in the tempo function to identify significant large and small pace transitions. Large pace transitions are considered to mark a story boundary, while small pace transitions only indicate an event within a scene. The algorithm does not try to extract semantics, but it considers that segments with different tempo belong to different semantic scenes.

Another approach using film grammar and tempo has been introduced by *Cheng et al.*[15]. They also use motion (MPEG-7 motion activity) and shot duration as parameters for their tempo function. In the first two phases of their algorithm similar shots are clustered and merged to scene units. After phase two an over-segmentation of the content can be observed. Therefore, in a third phase the tempo computation is used to group scene units to scenes. If a large transition in the tempo function of two scene units can be observed, the content of these two units belongs to different scenes and they are not merged. If only a small tempo transition between two scene units can be observed, they are merged if both have quick tempo. Adjacent scene units with slow tempo are not merged at all.

Aner et al.[3] presented an approach for clustering shots based on similar background images. Several key frames are extracted and a mosaic representation for each shot is created that shows the background of the setting. Similar mosaics are clustered. With the help of predefined plots it is possible to extract certain scenes from sitcoms (e.g. scenes that take place in the living room) or sports videos (e.g. penalty shots at basketball games). The definition of plots does not really correspond

to film-editing rules, but certain knowledge how sitcoms or sports videos are composed must be available. Therefore, this algorithm is mentioned in this category. It is a partial decomposition approach, as only scenes are extracted that correspond to plot.

A rule-based scene extraction solution has been presented by *Chen et al.*[12]. They try to identify dialog and action scenes in videos using predefined patterns and rules. A dialog scene consists of a sequence of three types of recurring shots: shots that show actor A, shots that show actor B and shots that show both actors. To discover dialog scenes 18 elementary dialog patterns consisting of different arrangements of these three shot types have been defined. Dialog scenes are identified in the shot sequence of a video using a finite state machine that covers the predefined patterns. The temporal appearance of shot patterns in action scenes is similar to the one in dialog scenes, but action scenes usually consist of shorter shots. Therefore, the shot length is examined to differentiate between dialog and action scenes. As this approach only defines rules for the extraction of dialog and action scenes it can be classified as partial decomposition approach.

Zhou et al.[129] and *Tavanapong et al.*[102] present their "ShotWeave" approach. Only the four corner regions and the background region at the top of key frames are used to compare shots based on color and motion features. Only these regions are compared for the similarity matching. The example in Figure 3.7 shows how the different film-editing rules are considered by this approach. Shot 1 is an establishment shot. It introduces the setting and the characters. Both characters start to move towards each other. Shot 2 shows the left actor, shot 3 shows the right actor (shot/reverse shot). Finally, in shot 5 when both actors meet each other the whole setting can be seen again. As all cameras are placed on one side of the scenery (in front of the 180 degree line), at least parts of a common background (the bar) can be seen in all shots. By examination of the corner and background regions the proposed algorithm tries to find such common objects in all shots of one scene performing up to three sequential steps. First, the background region is examined. If the difference

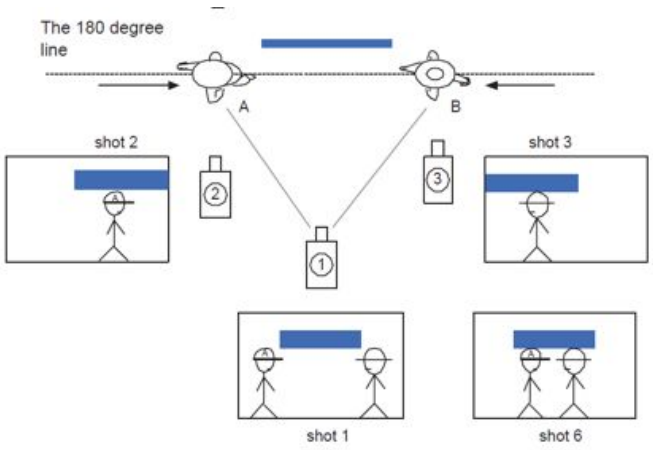

Figure 3.7: ShotWeave example and the 180 degree rule [102]

between the background regions of two shots is within 10 % both are considered to belong to one scene. Otherwise, the upper corners are compared. If the minimum of the two differences is within 10 % the corresponding shots belong to one scene. The upper corner comparison is used to detect the same locale in close-ups. If it shows no similarity, the same is done for the lower corners. This matching is used to identify traveling scenes. If all comparisons show no similarity, the compared shots are not regarded to belong to the same scene. At the end the overlapping links method [32] is used to merge overlapping clusters of shots.

Scene extraction from motion pictures is also performed by *Truong et al.* [104]. HSV color histograms are used to extract shots from a video. Two different approaches are evaluated for the scene detection. The first one identifies significant changes in the color distribution of the shots and marks them as scene boundaries. The second one is a so called neighborhood coherence approach, where each shot is compared with a certain number of preceding and following shots. If the coherence is below a threshold a scene change is identified. Both methods detect the majority of scene boundaries, but there are also a lot of false positives. Therefore, some refinement techniques are applied. Beside experiments with the size of the temporal window in the shot coherence approach, some film-editing rules are considered to

achieve improvements. Fades and dark areas are detected, as these two punctuations often indicate a scene change. In addition, a tempo analysis is performed. Neighboring scenes with high tempo (short duration and high motion) are merged to one scene, as two scenes with high tempo seldom occur in succession. Finally, so called "high impact colors" are detected. These are colors that are often used to cause excitement. If two neighboring scenes share the same high impact color, they are merged. While the original two approaches suffer from over-segmentation, the film-grammar-based improvements lead to better results.

Geng et al.[26] also introduced a partial decomposition approach using film grammar to identify dialog and action scenes. In addition to the alternating shot structure, the audio correlation of shots is examined to identify dialog scenes. For the detection of action scenes motion and audio correlation are used in combination with film-editing rules.

Another rule-based algorithm has been presented by *Chen et al.*[13]. For each shot of a video several key frames are extracted. Overlapping areas in the key frames are detected in order to extract background images from shots. Then a two-pass algorithm is used to identify scene boundaries. In the first pass shots are grouped to candidate scenes based on visual similarity of their background images. In the second pass each candidate scene is compared with its two subsequent scenes to pay attention to film-editing rules. If the candidate scene is similar to one or both subsequent scenes, the similar scenes are merged and the algorithm is again applied to the new scene. This process is repeated until no more scenes can be merged.

Video segmentation algorithms based on film grammar achieve a high accuracy for motion pictures. Directors typically rely on a set of basic rules for the scene construction. Considering these rules helps to identify scenes. Problems occur, if two adjacent scenes are similar and follow the same rules. The algorithms also fail where directors intentionally break the rules to achieve certain effects, like confusion of the audience. The main target of approaches presented in this category is the segmentation of motion pictures. Therefore, they are ideally suited for the problem

in Example 1. On the other side, endoscopic videos do not follow traditional film-editing rules at all, thus the presented solutions cannot be used in Example 2. A new film grammar for endoscopic videos could be, however, an interesting innovative way. Table A.6 shows an overview of the presented papers.

3.2.4 Hybrid Scene Segmentation Approaches

Finally, in this category scene segmentation approaches are introduced that combine methods presented in the previous categories.

Video segmentation based on visual and audio features has been proposed by *Huang et al.*[35]. They assume that a scene change is always associated with a simultaneous change of image, motion and audio characteristics. Twelve audio features are extracted to compute an audio dissimilarity function. Local maximums are searched in this function to detect audio breaks. Color histograms are used to compare the image similarity and a correlation function is used to compute the difference in the motion of successive frames. For each audio break a visual break is searched in the neighborhood. If one is found, it is regarded to be a scene boundary. Visual breaks are either color or motion breaks, or both of them.

Lienhart et al.[49] also presented an approach using audio and visual features. Great changes in the spectral content are registered as audio cuts. Shots are clustered based on a color and a texture feature. Furthermore, a dialog detection is performed using a face detection algorithm and the shot/reverse shot rule.

Only audio and color features are considered by *Sundaram et al.*[99], [100]. With the help of eight audio features audio scene changes are detected. Color coherence is used to detect visual scene changes. Audio and visual breaks that are near to each other are identified with a nearest neighbor algorithm. An example is shown in Figure 3.8. The circles denote audio scene changes and the triangles indicate video scene changes. Only where audio and video scene changes occur within a predefined temporal window are those breaks regarded to be scene boundaries. An improved version of their algorithm also considers silence and the structure of dialog scenes. It

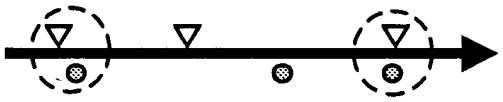

Figure 3.8: Synchronization of audio and video scene boundaries [99]

applies two additional rules [100]: (1) If a silent region intersects with a visual break, it is considered to be a scene change. (2) Strong visual changes alone also mark scene changes.

Chen et al.[12] also use audio and video features for scene detection. Three different values (volume, power and spectrum) are calculated. They are not combined. Whenever one of these features indicates a significant change, a scene boundary is detected. *Nitanda et al.*[67] use a fuzzy c-means clustering algorithm to classify audio segments into five classes (silence, speech, music, speech with music background and speech with noise background). They also identify scene changes by detecting simultaneous audio and shot cuts.

A combination of graph-, statistics- and film-grammar-based segmentation of talk and game shows is proposed by *Javed et al.*[39]. Shots are structured in graph representation based on color similarity and temporal closeness. Considering the special structure of commercials (rapidly changing shots) it is possible to automatically remove them from the video. The graph is used to detect story segments and for each one the likelihood of being a commercial is calculated. The remaining segments are classified into host and guest story segments.

Chaisorn et al.[8] presented an approach for news video segmentation using Hidden Markov Models (HMM), audio and video features (speaker change, audio type, color, motion, shot duration) and metadata (videotext). The shots of a video are classified into 13 categories, like Intro, Speech/Interview, Sports, Weather, Commercial, etc. These categories, location change information and speaker change information are used to perform the scene segmentation using HMM.

News video segmentation is also performed by *Zhai et al.*[125]. A shot connectivity graph is built, where the nodes represent shots and the edges indicate temporal transitions between them. Cycles that are connected at the anchor node in the graph are regarded to be a scene. Anchor shots are detected using color histograms and face matching. In a second phase a refinement is done. Based on color and motion information weather scenes are detected. Additionally, a database with 150 key words from different sports games is used in connection with a speech recognition algorithm to detect sports scenes. As anchor persons often appear more than once in a semantic scene, this algorithm over-segments videos. Therefore, neighboring scenes with visual and textual similarities are merged.

A real-time highlight extraction algorithm for live baseball videos has been introduced by *Ariki et al.*[5]. It is a partial decomposition approach. Pitcher scenes are extracted using visual features and a speech recognition system is used to extract text from the audio stream. As live radio broadcasts contains more speech than TV broadcasts, the radio stream is used. The extracted texts are matched against a baseball text corpus and if certain keywords (e.g. home run) can be identified, the corresponding scene is marked as highlight scene.

Arifin et al.[4] presented a segmentation approach using a pleasure-arousal-dominance (P-A-D) model. They do not try to bridge the semantic gap using a cognitive level approach like the most presented here, but to investigate an affective level solution. The P-A-D model analyzes color, motion and audio features to describe emotions, from unpleasant to pleasant (pleasure), from calm to excited (arousal) and how much attention an emotion gets (dominance). Six emotion categories can be identified using this model: sadness, violence, neutral, fear, happiness and amusement. A hierarchical-coupled dynamic Bayesian network topology is used to classify video segments on an affective level according to the detected emotions. A spectral clustering algorithm groups similar emotional segments. The scene segmentation is

performed using a directed graph. It models the temporal relationships between clusters and is used to identify coherent video scenes. This approach is especially suited for motion pictures, where emotions play an important role.

Zhu et al.[130] present an approach for the segmentation of continuously recorded TV broadcasts. A spatio-temporal clustering algorithm groups similar shots. Color and texture features are used for the similarity measurement. In addition, a template matching is performed to classify detected scenes into conversational, action and suspense scenes. The template matching is based on the average intensity distribution, face detection, activity analysis and audio analysis. Suspense scenes are scenes with low audio energy and low activity.

A role-based movie segmentation approach is presented in *Liang et al.*[48] and *Sang et al.*[82]. A script that contains the scene structure and related character names is aligned to the movie. Within the textual information the algorithm tries to identify characters and builds bag-of-role representations, according to the idea of the Bag-of-Words model.

Joke-o-mat HD, which is an improved version of Joke-o-mat [25], has been presented by *Janin et al.*[37]. In addition to the audio-based segmentation of the first version, this one also relies on metadata like expert annotations, fan-generated scripts and closed captions to improve the accuracy of the scene detection.

Ellouze et al.[24] combine two shot clustering techniques to identify video scenes. A time window of 30 seconds and Kohonen Maps [45] are used to cluster shots based on visual features (color and texture). Kohonen Maps provide the advantage that each shot is always compared to all other shots, in contrast to the pairwise shot comparison of other approaches. Moreover, a fuzzy 2-means clustering algorithm is used to classify shots based on tempo features (motion, audio energy, and shot frequency) into action and non-action content. At the end, the results of both algorithms are merged for the scene extraction.

A segmentation approach that goes beyond the boundaries of scenes has been introduced by *Wang et al.*[111]. Multimodal features like images, audio streams

and text transcripts are used to segment a recorded TV broadcast into diverse TV programs.

The more information is available the better it should be for scene segmentation purposes. Hybrid solutions are best suited to combine the strengths of different approaches and to eliminate the individual weaknesses. The challenge is how to combine different approaches or different features. It is not good to combine different features in advance and to use only a single similarity measure for them. Some features may be more important than other ones and the combination of features destroys the semantic message of one feature [49]. It seems to be a best practice to make a separate segmentation with each feature or method and to combine the different results at the end. This procedure helps to estimate the impact of each single method that contributes to a hybrid approach. As a result, methods that do not provide major improvements or approaches that are computationally expensive can be identified and excluded. Hybrid approaches that rely on metadata or on statistics-based methods are not well suited for the movie segmentation in Example 1, because no metadata and not enough training samples are available. A combination of visual-based and audio-based methods could be a good choice. In Example 2 it can be only relied on visual-features, no other data are available. Therefore, hybrid approaches cannot be used for the endoscopic videos in Example 2. An overview of the hybrid segmentation approaches presented in this section is given in Table A.7.

3.3 Use Cases for Video Scene Segmentation

While most approaches presented in this chapter evaluate their strategies with indexing of movies, I try to identify additional application scenarios for them. The following domains are considered: movies/TV series or sitcoms, news broadcasts, sports videos, single-shot videos and black-and-white videos. An approach may be assigned to more than one of these domains. In addition, it is investigated which scene segmentation approaches can be used in interactive segmentation tools.

3.3.1 Movies/TV Series or Sitcoms

Most existing approaches use motion pictures for the evaluation. The ground truth for the scene structure is determined manually or taken from DVD chapters. In human perception it is rather clear what a semantically meaningful scene of a movie is, but for automatic scene segmentation it is a complex task to identify the scene boundaries. Different types of scenes, like dialog scenes or action scenes, can be identified in movies. These scenes are characterized by different properties, like alternating shots or fast cuts. It is difficult to pay attention to all these different characteristics. Furthermore, movie directors tend to develop their own styles on how to structure scenes. Therefore, algorithms that work well with movies of one director can fail if applied to movies of another director.

In this chapter I presented many approaches that rely on film-editing rules (Section 3.2.3). Especially [2] presented very good results, but also [15], [12] and [104] are good suited for movie segmentation. Algorithms that are solely based on audio features are not very accurate [53]. Approaches that combine audio features with visual features have proven to be more successful and should be preferred. In [35], [49], [99], [100], [12] and [67] good results are achieved with algorithms that combine visual and audio features.

If only visual features should be used for the scene segmentation, the sliding window [78], [127], [115] approach or the overlapping links method [32], [46], [113] achieve good results. In movies a scene often corresponds to a certain location, thus finding similar shots within a certain temporal interval seems to be a good solution for detecting scene boundaries. Furthermore, the backward shot coherence [76] or the pattern matching in [10] lead to promising results using visual features alone.

Very good results for the scene segmentation of motion pictures can be achieved, if metadata like the screenplay or closed captions are available. Approaches using metadata have been presented in [16], [48], [82]. *Arifin et al.*[4] show that their pleasure-arousal-dominance model is an interesting approach for movie segmentation, but improvements regarding the accuracy are desired.

Graph-based algorithms are less suited for the scene segmentation of motion pictures. In most cases these algorithms result in over-segmentation, as the evaluations of the presented algorithms show. RoleNet by Weng et al. [114] and the speaker graphs of Sidiropoulos et al. [90] seem to be the most promising graph-based approaches for movie segmentation.

Statistics-based algorithms (Section 3.2.2) can achieve good results in this domain, but movie scenes are more diverse than scenes of TV shows or news videos and thus the creation of a discriminative training set is a more challenging task.

The task of finding scenes in TV series and sitcoms is quite similar to finding scenes in motion pictures. All of the suggested approaches can also be used for that task. The big difference is that TV series and sitcoms are typically characterized by a fixed group of actors and a limited set of locations where the plot takes place. These characteristics remain the same across all episodes. Video scene segmentation approaches can take advantage of these always repeating characteristics. If one approach delivers good results for one episode it is very likely that it will work well with all episodes.

3.3.2 News Broadcasts

News videos have a clear structure. Reports are mixed with anchor shots and interview scenes. In addition, some special program sections exist, like the weather forecast. Relying on this clear structure it is possible to achieve a very good accuracy for the segmentation.

In general, graph-based approaches (Section 3.2.2) are well suited for the segmentation of news videos. Always recurring scene structures can be well mapped to graph representations. The majority of the presented approaches only evaluated their algorithms with motion pictures, but it should be easy to apply all of them to news videos. Most graph-based algorithms suffer from over-segmentation, but for news videos over-segmentation may be acceptable. For example, consider a news scene consisting of an anchor shot, a report and an interview in the end. If such a scene is

segmented into three parts this result would be still acceptable in my point of view. The algorithms in [66], [65], [126] or [90] achieve very good results.

The only graph-based approach that is not suited for news video segmentation is the one presented by *Weng et al.*[114]. The identification of different roles may not work for news videos. It seems impossible to identify connections between the anchor persons and the persons in the news reports, because anchor persons usually do not occur in shots that belong to news reports.

Using the statistic-based approaches presented in Section 3.2.2 is another possibility for scene detection in news videos. The segmentation of news videos with statistics-based approaches has already successfully been shown in [110] and [33]. The clear structure of news videos that always remains the same enables a precise training of the statistical methods used and leads to very good results.

The classification of *Zhu et al.*[130] may also be used for the segmentation of news videos. Instead of distinguishing between action, dialog and suspense scenes it should be possible to classify a news video into anchor (suspense), interview (dialog) and report (action) scenes.

The Joke-o-mat [25], [37] could be turned into a News-o-mat. For example, the audio stream could be used to identify anchor segments, enabling users to jump from one anchor shot to another.

A different news segmentation approach could be implemented using the pleasure-arousal-dominance model [4]. Instead of identifying typical scene structures it may be used to index a news video into segments that cause different emotions.

Chaisorn et al.[8] present an approach that is specialized in the scene segmentation of news videos. A sophisticated analysis using multiple features and methods is performed that leads to very good results. It is the only approach that uses, besides others, textual information from the videotext for the segmentation.

In news videos a lot of captions are displayed. Using OCR algorithms the text information can be extracted from these captions and used as additional metadata.

Applying automatic speech recognition algorithms to news video segmentation should also lead to good results.

3.3.3 Game or TV Show Videos

Game and TV shows typically have characteristics that do not change during different shows. They are always produced in the same studio, the setting always looks the same, specific jingles are played according to certain events that occur and for the majority of shows even the camera positions and lighting conditions do not change. For the segmentation of Game or TV shows it may be sufficient to perform only a partial decomposition that extracts only certain situations of interest, like games or questions. Scene segmentation approaches that search for common characteristics in videos should be used.

Scenes look different for different types of shows. In a quiz show like "Who wants to be a millionaire?" a scene could be a question from the point the host reads it out until the solution is shown. Other shows consist of longer scenes. In a late show scenes correspond to different guests of the show. Therefore, different scene segmentation approaches are suited for different types of Game or TV shows.

As the approach of *Javed et al.*[39] shows, a segmentation into host and guest scenes can already be achieved with high accuracy. Audio-based approaches [53], [68] can be used to detect different speaker segments, cheering of the audience or the jingles mentioned before in order to perform a scene segmentation. For example, in "Who wants to be a millionaire?" each question is preceded and each answer is followed by a certain jingle. Tools like the Joke-o-mat [25], [37] could be used to navigate from one question to another.

The pleasure-arousal-dominance model [4] should be suited to distinguish between different success of participants in game or quiz shows. Positive emotions indicate scenes where the person wins, while negative emotions show that the person was not successful. For example, a question has not been answered correctly.

TV shows that use fixed camera positions and always the same camera pans and zooms can be segmented with an approach based on motion features [17]. Film-editing-rule based approaches that classify scenes into action or dialog scenes [12], [26] can also be applied to game shows. Dialog scenes correspond to scenes where the host talks to the participants, while action scenes are scenes that show the participants competing in a game.

3.3.4 Sports Videos

Many different athletes are typically involved in a sports video. The question is how a scene actually looks like in a sports video? Finding scenes often corresponds to finding highlights or to finding the segments where a specific athlete is shown. Therefore, finding scenes in sports videos strongly depends on the sport and the defined task. Some sports are characterized by repeating scenes, such as ski jumping, others by long rather boring and short exciting scenes, like soccer.

Ariki et al.[5], *Zhai et al.*[125] and *del Fabro et al.*[17] present already algorithms that identify typical characteristics of certain sports videos to perform a scene segmentation. Especially [5] achieves very good results, but also [125] and [17] are good suited for their specific problems.

The Joke-o-mat [25], [37] could be turned into a Highlight-o-mat that enables users in finding highlight scenes of sports videos. An audio classifier can be used to identify cheering of the spectators or excited speech of the news reporter. In most cases these two characteristics go hand in hand with highlights of a sports event.

Other audio-based approaches are also well suited for detecting highlight scenes. An approach specialized in finding rally scenes in racquet sports videos is presented in [51]. But also the algorithms in [53], [68] can be used to identify scenes based on cheering or excited speech.

The film-editing-rule-based approaches in [2], [15] can also be used for a highlight scene detection. They classify scenes according to a tempo function. High tempo corresponds to action scenes in motion pictures. In sports videos high tempo can

indicate the presence of a highlight. This assumption may not hold for all sports. It must be investigated under which conditions tempo functions may be used.

The identification of arousal and dominance introduced in [4] is another possibility to detect highlights scenes based on the excitement and the emotions contained in videos. For example, goal scenes in soccer games may be found by searching for video segments of athletes that are joyfully celebrating a goal.

The classification of *Zhu et al.*[130] into action, dialog and suspense scenes can also be applied to the sports domain. The detection of action scenes corresponds to the identification of highlights, dialog scenes are interviews with trainers and athletes during or after the competition and suspense scenes are the remaining parts where nothing exciting happens.

Videos of sports events where athletes compete one after another result in repeating scenes with similar characteristics as [17] shows. Such videos are good suited for the use of graph-based (Section 3.2.2) or statistics-based [34], [119] approaches.

3.3.5 Single-Shot Videos

Many scene segmentation approaches first search for shots and then those shot boundaries are identified where scene changes may occur. These algorithms cannot be used for videos that consist only of one single shot, e.g. surveillance videos or videos of arthroscopic surgeries. In such videos scene segmentation algorithms have to search for other reference points to identify scenes. Segmentation algorithms are not expected to perform a full decomposition of the content, but only to identify important scenes. In surveillance scenarios only those scenes must be identified where something happens. In videos of arthroscopic surgeries blurred scenes where nothing can be recognized can be excluded.

Tracking of persons can be done with face recognition algorithms like in [114]. But for a successful use of face detection the quality of the videos must be high. In surveillance scenarios this is often not the case. If a single-shot video contains an

audio stream, it may be possible to make a segmentation based on different audio classes like in [53].

Another possibility for the segmentation of single-shot videos is motion analysis. The identification of recurring patterns in dominant motion histograms [17] or the detection of segments with different tempo [2], [15] are promising approaches for such a strategy. A similar approach is the energy minimization algorithm in [30].

Statistical methods (Section 3.2.2) are also well suited for detecting certain events. The performance of such methods depends again on the quality of the training set, but single-shot videos typically show a narrow domain and thus discriminative trainings sets can be built with reasonable effort.

If only two different types of scenes must be detected in single-shot videos, a trained classifier can be used to distinguish between these two types. Such an approach is presented in [118].

3.3.6 Black-and-White Videos

A lot of scene segmentation approaches rely on color features. For the segmentation of black-and-white movies other features must be investigated. Surprisingly few approaches have tested their algorithms with black-and-white videos so far. The scene structure depends on the type of video (movie, news, etc.). Characteristics of these types of videos are mentioned at the corresponding use cases. In this section it is only pointed out which approaches do not rely on color features and thus can be used for black-and-white videos.

One possibility to segment black-and-white movies is to use audio-based approaches [53], [68], if an audio stream is available. Also the graph-based approach presented in [90] that creates speaker graphs is well-suited in such a case. All other graph-based approaches rely on color features and are thus less suited for black-and-white videos.

Metadata-based approaches like [16], [48], [82] are another solution. The problem is that most black-and-white-movies had been produced a long time ago and no

metadata is available for them or not in an automatically processable form, e.g. handwritten notes.

Film-editing-rule-based approaches that rely on tempo functions [2], [15] are also well-suited for the scene detection in black-and-white movies, because the segmentation is based on motion intensity and shot duration. *Mitrovic et al.*[61] only use visual features (SIFT, edge change ratio and block-based intensity histograms). Their evaluation with artistic archive documentaries shows that this algorithm achieves good results for black-and-white videos.

3.3.7 Interactive Scene Segmentation

All approaches presented in this chapter rely on automatic scene extraction. In this section it is shortly investigated, how some of these algorithms could be applied to interactive scene segmentation tools. It may be possible to incorporate most algorithms presented in this chapter into an interactive tool, but I only focus on some selected approaches. By implementing these algorithms into easy to use graphical tools, their accuracy may be enhanced considerably by a human in the loop. Of course, for interactive applications the algorithms used must not have a too high runtime complexity. Otherwise users would have to wait too long for the results. As, unfortunately, most authors make no comments regarding the complexity of their algorithms, I cannot consider this aspect here in detail.

As a first example, an interactive application could make use of shot strings [55]. Users could define shot strings consisting of example shots in advance and thus they could tune the algorithm according to the characteristics of the video that should be segmented.

The graph-based algorithms in Section 3.2.2 could also be directly mapped to a graphical user interface. By defining simple graph representations with example shots, users could define the structure of scenes they are searching for. RoleNet [114] could also be used for video retrieval tasks. By defining social relationships between actors, users could specify scenes with certain actors they are searching for.

Some approaches search for common background images [3], [13] or similar objects in the background of scenes [129], [102]. Instead of automatically searching for such common background images or objects, users could be enabled to interactively define them. For example, by manually selecting background regions.

The visualization of the dominant motion histograms used by [17] is shown in Figure 3.12. In an interactive application users may be able to select the motion pattern they are searching for interactively. A video exploration tool that provides an interactive motion-based search feature among many other features is presented in [88].

The pleasure-arousal-dominance model by *Arifin et al.*[4] shows only modest results. Maybe an interactive tool can improve the results. For example, users could annotate a few selected parts of a video with predefined emotions in a training phase in order to improve the algorithm. Another approach would be an application that allows users to extract only scenes with certain emotions.

3.4 Open Questions in Video Scene Detection

The comprehensive review in this chapter shows that a lot of different approaches for video scene segmentation have been developed in the last decade. It is difficult to make comparisons regarding the accuracy of the different algorithms. All authors use their own test sets, only few try to use the same videos that others used before. Compared to the first approaches some improvements have been achieved, like the reduction of over-segmentation. After more than 10 years of research in this field it can be stated that the different algorithms with their individual test sets and evaluation methods reach similar, but not comparable results.

A unified test set for video scene segmentation is needed. TRECVID [93] performed several video retrieval tasks during the last decade and a remarkable community of video retrieval experts emerged in that time. It would be very interesting to perform a video scene segmentation task in the context of an ongoing TRECVID

workshop. It would ensure a unified evaluation method for all algorithms. At the moment different solutions do not only use own test sets, but they also have different tolerance levels for correctly detected scenes.

It is not only difficult to compare the accuracy of current approaches, but also to compare the complexity and the performance of different algorithms. Only very few papers give information about the computational complexity of the presented solution. It should always be possible to estimate an algorithm by relating its accuracy to its performance. Especially for situations where a sophisticated analysis is not possible or only limited resources are available, e.g. in live scenarios, fast algorithms are needed. If algorithms can be compared regarding their complexity, new potential for performance improvements may be identified.

Furthermore, appropriate applications for scene segmentation algorithms must be identified. Most of the presented approaches in my survey are evaluated with motion pictures, which typically have a clear scene structure. It is quite easy to obtain a ground truth for them or to take DVD chapters as reference. I am of the opinion that algorithms which segment motion pictures are treating an artificial problem. For all professionally produced motion pictures a lot of metadata is available, including also scene boundaries. For TV recordings metadata is usually not available. In such cases scene segmentation makes sense, but as set-top boxes and hard disk recorders have typically only low computational power, the complexity of the algorithms is again an important aspect. Another possible application field is the growing amount of videos produced by amateurs or hobby producers.

3.5 Video Scene Detection Based on Recurring Motion Patterns

In the sports domain many videos can be found that have been captured from fixed camera positions and that show similar situations throughout the video. The scenes of such videos look identical. The shots have a high visual similarity and in the

audio stream there are typically no significant changes that could indicate a scene boundary. It is difficult to use existing video scene segmentation approaches for such videos, because most of them rely on detecting significant differences in the color distribution or in the audio stream to identify scene changes.

In the context of my thesis I have implemented an own scene segmentation approach based on the motion information of video streams [17]. Not only the motion within one frame is observed, but how the motion is distributed across several frames. If adjacent frames have the same motion direction, these frames form a coherent motion sequence. Recurring motion patterns are identified in the video stream and scenes are extracted, which correspond to these patterns.

For example, think of a ski jumping event where the jumps of all competitors are always shown from the same camera positions and vantage points with similar zoom and pan sequences. An example for a recurring sequence is given in Figure 3.9. It shows an amplified part of the navigation index of Figure 2.4. The upper bar represents 150 seconds of a ski jumping competition, where three athletes are shown. The takeoffs of the three competitors are marked with *(A)*, *(B)* and *(C)*. The lower bar shows a 15 seconds long excerpt of athlete *(B)* in detail. The different motion sequences that occur during his jump can clearly be recognized. They are expressed by different colors. The first greenish V-like sequence describes the motion during the jump, the yellow sequence shows the landing and the two other greenish and the pink sequences occur while the athlete runs down the out run and stops. Comparing sequence *(B)* with the other ones in the upper bar, it can be well recognized that the three scenes consist of similar motion sequences. Regarding the aim of my algorithm, the lower bar shows a motion pattern that the algorithm searches for in the whole video stream.

Human observations with the video exploration tool introduced in [88] have shown that recurring motion patterns often indicate semantic scenes of a video. Thus, only by examination of the low-level visual feature motion, high-level scenes can be found

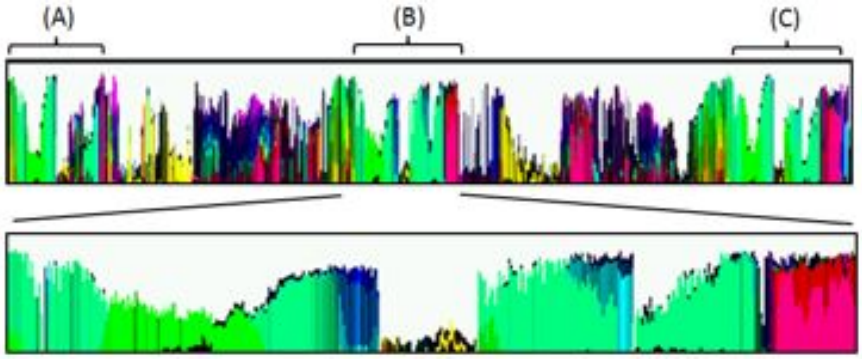

Figure 3.9: Visualization of motion sequences that occur in a ski jumping video [87]

in a video. The algorithm does not gain any semantic information from the motion information, but it is supposed that the results contain semantically meaningful content. The identification of the semantics has to be done by users looking at the results.

As recurring motion patterns in sports videos often correspond to scenes showing different athletes, the proposed solution is good suited to index sports videos based on athletes. If a retrieval system incorporates such an indexing, users could easier identify those parts with athletes they are interested in. People that missed a competition may only be interested in the best athletes or in athletes representing their country. Others may be fans of certain athletes and are only interested in jumps of them. A trainer on the other hand may be interested in scenes of his own athletes, but maybe also in the scenes of the best competitors to be able to make comparisons.

The algorithm is divided into four steps. First the motion information is extracted from the compressed video stream and a motion histogram is calculated for each frame. Then coherent sequences with similar motion are searched. A hierarchical clustering algorithm is used to group similar motion sequences. Based on the resulting clusters recurring motion patterns are identified. It must be noted that videos are not segmented into all their scenes, but only in those scenes that match the identified pattern. In contrast to typical video segmentation approaches it does not result in a single, static segmentation. Different, logical segmentations are extracted for a video,

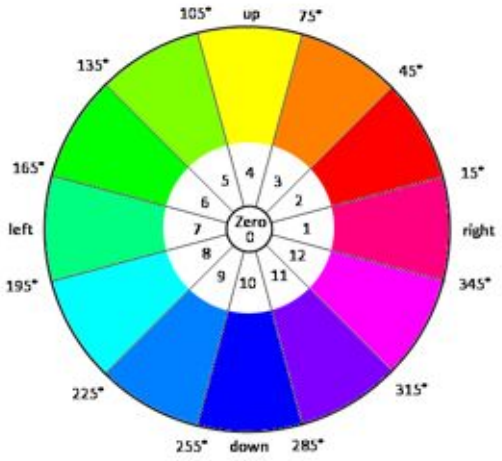

Figure 3.10: Motion vector classification for a motion histogram with 13 bins. Bin 0 expresses the amount of pixels with no motion [87]

according to the hierarchical video representation introduced in Figure 2.1, where an indexing of videos on scene-, shot- and frame-level is suggested.

3.5.1 Motion Classification

The motion histogram for each frame is created using the motion vector information contained in H.264/AVC bit streams. As I-frames do not contain motion vectors, the motion information for I-frames is interpolated from the two adjacent frames. The extraction of the motion information is done in the compressed domain, thus no full decoding of the video stream is needed [87]. The resulting motion histogram consists of 13 bins as illustrated in Figure 3.10. Each bin indicates the percentage of pixels that move in that particular direction. For example, bin 1 shows how many pixels move to the right, which means how many pixels belong to macroblocks with a motion vector angle between 345 and 15 degrees. Bin 0 indicates the amount of pixels that do not move at all.

3.5.2 Sequence Detection

After the motion histogram has been created for each single frame, sequences of frames with the same dominant motion direction are identified. Each frame is compared with its successor in the stream. If both show a similar motion direction, they belong to the same sequence and the second frame is compared with the next one. Two adjacent frames are regarded to have a similar motion if the relative difference in the motion direction between them does not exceed a certain threshold. Empirical investigations have shown that 40 % is a good value for a broad range of videos.

With only one iteration over all frames it is possible to find all connected motion sequences within the video stream. Motion sequences with less than 25 frames are disregarded, because they often occur due to noise in the motion information. This restriction avoids including noisy information in the following steps of the algorithm. The motion sequence detection is somehow similar to shot detection, because rather short video sequences are detected and in most cases the boundaries or motion sequences correspond to shot boundaries. An exception may be zoom and pan sequences, where the dominant motion can change within one shot.

3.5.3 Clustering

In the next step a hierarchical clustering of the detected motion sequences is performed. A flow diagram that illustrates all clustering steps is shown in Figure 3.11. For each motion sequence a key frame is selected. To keep the algorithm simple and fast the center frame of the sequence is used, which is a common approach in video retrieval [105].

At the beginning of the hierarchical clustering each detected motion sequence forms an own cluster. Each key frame of a cluster is compared to the key frames of all other clusters. As distance metric the absolute difference between all motion histogram bins of the key frames is used. Clusters that have a minimal absolute distance that is below 5 % are merged and a new clustering round starts.

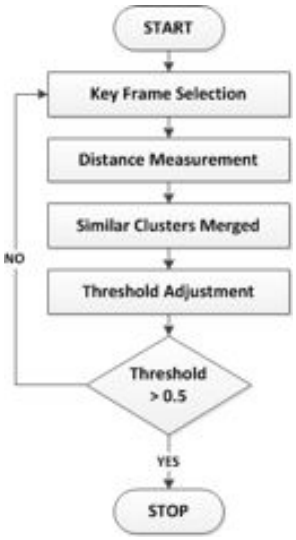

Figure 3.11: Flow chart illustrating the steps of the hierarchical clustering algorithm

Beginning with the second iteration, key frames are identified for clusters that consist of more than one motion sequence. Again the center frame of each sequence is compared to all other center frames of the cluster. The center frame with the minimal distance to all other ones is selected as key frame for the whole cluster and similar clusters are merged again.

If in a round no clusters can be merged, the threshold is increased by another 5 %. This is repeated until the threshold reaches 50 %. Empirical observations showed that merging clusters with a distance higher than 50 % results in blurred clusters that have negative impact on the results.

3.5.4 Identification of Recurring Patterns

The clusters created in the previous step are numbered ascending. Each one of the chronologically ordered shots gets the corresponding cluster number assigned and recurring patterns are identified in this sequence of cluster numbers. A sliding window with an initial size of 4 is used. It is moved over the created cluster sequence to identify the pattern candidates. For each pattern candidate it is estimated how many matches can be found in the whole cluster sequence. If the end of the cluster sequence

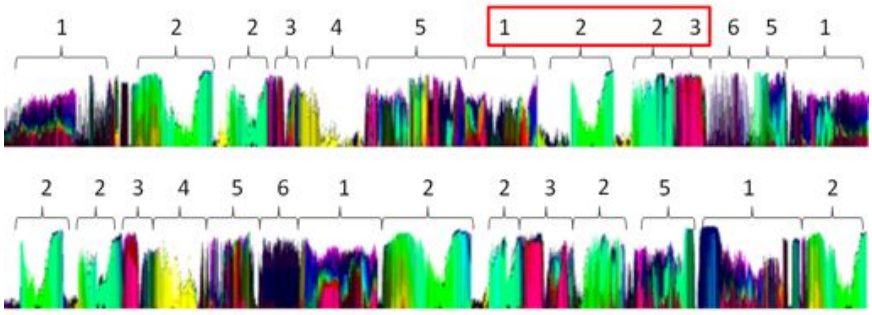

Figure 3.12: Recurring patterns are identified in the sequence of cluster numbers. In this example the pattern 1-2-2-3 is the most frequent one. [17]

is reached, the size of the sliding window is increased by one and the pattern matching starts over at the beginning.

The pattern matching is repeated until the size of the sliding window equals to one third of the number of identified motion sequences. This is done because at least three occurrences of a motion pattern should be embodied in a video stream in order to be able to call it a recurring pattern.

At the end the most frequent pattern is used for the video segmentation. Each video segment that matches that pattern forms a scene. This leads to a partial segmentation of the original video, because all other scenes that do not correspond to the most frequent pattern are ignored. Each identified scene contains all frames from the beginning of the first motion sequence to the end of the last motion sequence that are a part of the pattern. Therefore, frames between the included motion sequences that have been left out due to the restrictions of the sequence detection (only sequences with at least 25 frames are recognized) are also added to the scene.

The basic principle of the pattern matching is illustrated in Figure 3.12. Each detected motion sequence is assigned to one of the clusters, which is expressed by the cluster numbers above of each sequence. In the example shown, the algorithm detects four occurrences of the pattern *1-2-2-3*, which corresponds to exactly one jump scene in the video stream.

The identified scenes represent the top level of a hierarchical video representation. The second level consists of the motion sequences that build the basis for the scenes. The third level contains the frames of all motion sequences and the fourth level consists only of the key frames.

3.6 Evaluation

The presented scene segmentation approach was evaluated with two test videos from the sports domain, in particular with videos of ski jumping competitions.

3.6.1 Test data

Videos of ski jumping competitions are well suited for testing this algorithm, because they contain a lot of short recurring sequences that have been captured from fixed camera positions.

The first video (*oberstdorf*) has an overall length of 23 minutes and 24 seconds and it shows the jumps of 16 competitors. At the beginning of the video stream some commercials and interviews are shown. The second video (*garmisch*) has a duration of 1 hour 39 minutes and 36 seconds. It shows 68 jumps that are distributed across the whole stream. From time to time single jumps are interrupted by commercials, interviews or result tables. Four replay scenes are shown with normal playback speed in the break between the two runs of the competition, thus they are also considered to be found. This ground truth has been manually created for both videos.

3.6.2 Results

The effectiveness of the presented algorithm is estimated with the amount of identified jump scenes. The common evaluation metrics *Recall* and *Precision* are used. R is the set of relevant scenes and P is the set of scenes found.

$$Recall = \frac{|R \cap P|}{|R|}$$

Video	oberstdorf	garmisch
Relevant scenes	16	68
Size result set	20	115
True positive	12	59
Recall	0.75	0.87
Precision	0.6	0.51
Slow motion scenes	2	21
Recall (incl. slow motion)	0.78	0.9
Precision (incl. slow motion)	0.7	0.7

Table 3.2: Evaluation results for the scene detection based on recurring motion patterns

$$Precision = \frac{|R \cap P|}{|P|}$$

The results are shown in Table 3.2. For the first video (*oberstdorf*) 12 from 16 relevant scenes are found. The result set has a size of 20 scenes, this leads to a recall of 0.75 and a precision of 0.6.

For the second video (*garmisch*) the result set consists of 115 scenes, 59 of them show jump scenes. This means a recall of 0.87 and a precision of 0.51.

The low precision values are a consequence of how the results of the algorithm are rated. Only jump scenes that are shown with normal playback speed are counted as relevant results. The result set also contains slow motion replay scenes that show jumps. In the first calculation of recall and precision these playback scenes are not regarded to be relevant results. I observed that in many cases the directors of the two videos used vantage points for the replay scenes that differ from the original scenes. Only in a few cases the same cameras are used. Although slow motion scenes are played with less speed they show similar motion patterns like the original scenes if they were captured from the same camera positions. Slow motion scenes have less motion intensity, but still enough to be within the defined similarity threshold. Furthermore, slow motion scenes are slightly longer than the original scenes. As the clustering algorithm is generally designed to group sequences of different length, it is also able to detect and cluster slow motion scenes correctly.

As a consequence, I investigated the motion patterns of the replay scenes in the result set and noticed that 2 slow motion scenes of the video *oberstdorf* and 21 slow motion scenes of the *garmisch* video show the same motion pattern like the relevant scenes in the result set. If recall and precision are calculated again including the identified replay scenes, the recall slightly increases and a precision of 0.7 is achieved for both videos.

3.6.3 Performance

The most time consuming task is the extraction of the motion information from the video stream. This information is extracted in the compressed domain of H.264/AVC videos. Motion classification is a low-complexity task, as it does not require full decoding of the video. The runtime complexity is dominated by the entropy decoding, which consumes 22 to 42 percent of the full decoding workload [87].

For the three other steps of the algorithm (motion sequence detection, clustering and pattern matching) measurements were performed on a desktop computer with an Intel Core2 Duo CPU with 2.8 GHz and 4 GB RAM. The *oberstdorf* video consists of 35112 frames. 215 motion sequences and 20 scenes are detected in 1230 ms. The *garmisch* video consists of 149415 frames. The algorithm identifies 933 motion sequences and 115 scenes in 48678 ms. In fact, less than one minute for a video that has a length of nearly 1 hour and 40 minutes. The processing time grows significantly for the longer video. This is due to the fact that more motion sequences are found in the longer video and thus the clustering needs more time.

CHAPTER 4

Non-Sequential Composition of Multimedia Content

Nowadays a tremendous amount of community-contributed content is available on social media sharing platforms and it is getting more every minute. The problem is that for users of such platforms it is becoming harder to find the content they are searching for. Simple grids or lists showing the results of a query are not sufficient anymore. New ways for the exploration of the content are needed.

In this thesis I am focusing on real-life events and I am introducing a new idea where content can be explored in connection with the context where it was produced. Instead of searching for particular content, users are able to explore community-contributed content related to a real-life event. I rely on community-contributed content, because photos and videos shared by other people express what those people saw with their own eyes during the capturing process. By assembling content from different people, we also assemble their different views and a new and rich representation can be created. We call this emerging view the *The Vision of Crowds*.

Users that explore such presentations can benefit from a compact representation revealing to them the most important happenings of a real-life event, which helps

them to save time. Furthermore, they may discover content they would not have thought of on their own, because people sharing content captured during an event may also report about unknown or even surprising elements, which in the end creates a richer view. Other approaches presented so far in the context of real-life events in multimedia did not try to create a new view of events based on community-contributed content, but rather to assign content to the events where it was produced. Further details are given in section 2.2.2.

One of the main principles this thesis is based on is a new understanding of video streams. Videos are not distributed and watched as a whole anymore, but only parts regarded to be interesting or important. Therefore, people should be able to assemble – or at least to consume automatically assembled – video streams consisting of selected units from different sources. In this thesis this process is called *composition*. The basic composition concepts are introduced in section 2.1.2.

The composition can be compared with the work of a director who has to distinguish between relevant and not relevant content. In this chapter three special composition types regarding real-life events are presented, which have been developed in the context of this thesis: (1) live event summarization, (2) interactive event summarization, and (3) event summarization using community-contributed content.

4.1 The Vision of Crowds

Twenty years ago people were informed about a social event, such as a royal wedding, through a few, authorized, professional camera teams and journalists of printed press. Nowadays, a vast amount of additional photos, videos and text, the latter mainly in form of metadata of the images, are uploaded to social media sharing platforms, such as Flickr and YouTube. At social events many visitors are producing and sharing photos and videos with their digital cameras or with camera-equipped mobile phones. This user-generated content can serve as additional source of information for the news coverage of social events. Reports may be enhanced by mixing professionally produced

content with the contributions of visitors in order to present a comprehensive and multifaceted view of an event.

Every visitor has its own vision (view) of a social event. Different people may be interested in different activities or they may have different intentions when visiting an event. As visitors are spread across the whole area where an event takes place, photos and videos are captured at different places. Composing the contributions of different visitors leads to presentations that express how a crowd of people is experiencing an event. People looking at such presentations are able to get an impression of an event by watching presentations not only through the eyes of a single director, but through the eyes of many visitors – the Vision of Crowds.

The idea of the Vision of Crowds is a paraphrase of the well-known notion of the Wisdom of Crowds [101]. Decisions that are made by a group of people are often smarter than decisions that are only made by a single person. Many web applications rely on it by gathering information, e. g. for image annotation. However, in the focus of our interest is not how a group of people decides, but how a group of people witnesses a social event. One way to get this information is to rely on crowdsourced data by looking at the photos and videos that individual visitors have captured.

The Vision of Crowds goes farther than crowdsourcing [21], which is an emerging approach to collect and maybe fuse data of most different sources. The focus of the concept of the Vision of Crowds is not collecting or even fusing the data, but rather to use them as a basis for creating semantically valuable summaries of an entire social event.

At many events professional camera teams are on site as well to produce high-quality reports. They typically try to identify the best locations for their cameras in advance to be able to capture the most important situations. In contrast to visitors they are less flexible, because the number of their cameras and their possible positions is limited. The visitors instead are spread all over the area where the event takes place. Therefore, they are in a position to capture situations from all over that area, even situations that take place in different locations at the same time. They may

capture situations that the professional camera teams miss. Moreover, a specific situation can be covered by several amateur clips from different viewpoints, maybe revealing different details. Professional camera teams on the other hand, may have access to locations that are closed for visitors and of course they usually take better images and videos than common visitors. In this thesis I am only concerned with summarization based on community-contributed content. On the long term a hybrid solution assigning special weights to professionally produced content may lead to better results.

In video summarization [63][105] it is common to try to find the most important key frames, shots or scenes in a single video in order to compose a shorter video or a compilation consisting of still images. Video summaries provide a short alternative to the original video. In my work I do not summarize single videos, but rather whole social events. Photos and videos showing interesting situations are identified in order to be able to compose semantically meaningful *event summaries*. The aim is to show different situations and activities, which happened during an event.

For this study only social events related to entertainment were used. However, the presented approach is also applicable to other events, such as a traffic jam on a high-way [106], seen by a number of drivers on the road, or a certain medical event, identified by a group of medical doctors in an arthroscopic surgery video [58].

4.2 Basic Composition Ideas

The basic ideas for the composition of video streams consisting of multimedia units from different sources are introduced first.

4.2.1 Sequential and Parallel Composition

The composition itself can be considered as a spatial and temporal aggregation of video units. Basically, two different types of compositions can be defined: *sequential composition* and *parallel composition*.

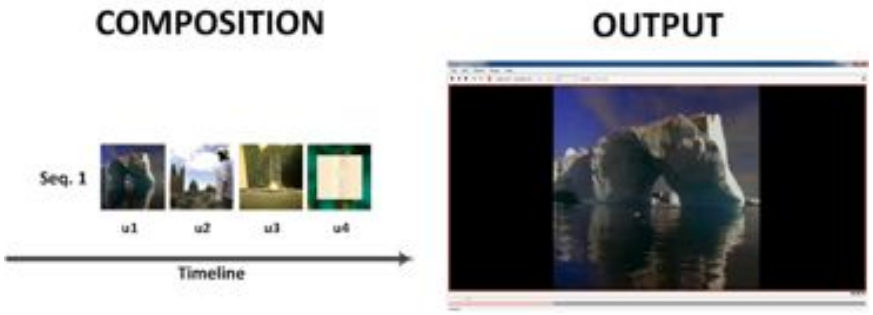

Figure 4.1: Schematic illustration of a sequential composition consisting of 4 units and the corresponding presentation in the video browser

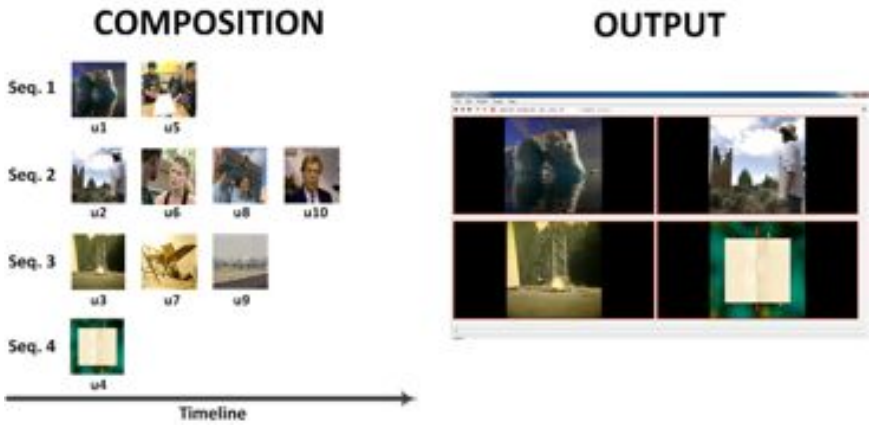

Figure 4.2: Schematic illustration of a parallel composition consisting of 10 units and the corresponding presentation in the video browser

In a sequential composition (illustrated in Figure 4.1) units are aligned sequentially one after another. The playback of a unit starts as soon as the preceding unit has finished. A traditional video stream, which is watched from the beginning to the end, is defined as a sequential composition of video units (e.g. frames) under certain QoS constraints (e.g. 25 fps). When using parallel compositions (shown in Figure 4.2), a certain number of units are shown to the user at the same time. There is no correlation between parallel video streams. Each stream is played independently from the others.

4.2.2 Composition in a Distributed Self-Organizing Multimedia System

The quality of a composition depends especially on two criteria: the availability of video units and the information about the content of a unit. The more information about a unit is available, the more accurate it can be estimated how well that unit fits into a composition or not. A lot of information about the content can be gathered from users or user communities. People are organizing themselves to tag, annotate, rate or exchange content.

The composition takes advantage of these self-organizing activities and utilizes as much human knowledge as possible for selecting appropriate units for a composition. The more human knowledge is available the easier it is to bridge the semantic gap. Beside the content and the visual quality of the units, the context information is very important to satisfy the individual intentions of the users. For example, textual descriptions help to identify units that meet the search requests of users. Context information, like the location where and the date when the content has been captured, as well as usage statistics and ratings can be used to distinguish interesting from non interesting units. Furthermore, if the intention of a user is known, units can be selected especially to the needs of that user and the content can be arranged in a way that well supports the user's task. In fact, all available information about the content and its context is used at the composition. Research on user intentions goes beyond the scope of this thesis. Interesting work has been done in the field of user intentions in the recent years [43][44][56][98].

In a self-organizing distributed multimedia system no global knowledge exists which units are available in the network at a certain point in time. Each node only has a local view of the available content. Only units that are available at a certain node can be used for the composition there. If selected units are not locally available, the QoS characteristics of the network must be considered in the unit selection process. Compositions that demand for high quality of the content and low delays cannot be

assembled with units that provide only low quality or cannot be delivered without high delays.

For the composition itself it is not necessary to know where the requested units are located. The self-organizing network must ensure that a good replication of units throughout the network is performed and that all parts of a requested composition are delivered to the right place, independent of the location of the units. For that reason, I do not consider any delivery or QoS issues in my thesis. I postulate that all units known by a client can be delivered in the desired quality and on time. How these issues are accomplished is out of the scope of my thesis. An interesting solution is presented in [95][96][97].

4.2.3 Manual vs. Automatic Composition

Users are not only passive consumers anymore. They become to active composers of multimedia content. This does not mean that an interactive human user always has to take the burden of manually defining compositions: predefined composition patterns can serve for different types of users and user intentions. The essential point is that the presented concept basically supports free and flexible interactive compositions and compositions, which can be formulated fully automatically.

In real usage scenarios it is rather common to offer a GUI, which helps users in composing video streams. But also a combination of automatic and interactive composition is possible. This means, a composition is suggested based on a predefined presentation profile, which may be modified by the user before it is shown on the screen.

For situations where presentations are shown to a bigger audience the intention of each single user cannot be identified and considered. Therefore, presentation profiles must be defined for the whole audience in advance. New video streams are automatically composed for the whole audience, without any human interruption. This automatic composition based on predefined profiles is referred to as *automatic director*.

4.3 Live Event Summarization

In a first experiment I investigated the concept of the Vision of Crowds in a live scenario [19]. The question was whether it is possible to identify the hot spots of an event solely based on community-contributed content while that event takes place. The assumption is that people tend to move to locations where something interesting is happening. If many people are in one place, it is also more likely that a lot of content will be produced there.

In November 2010 an event called *The Long Night of Research* took place at the Klagenfurt University. At 104 exhibition stands, spread across the whole campus, different research projects were presented to the public. More than 5000 people visited the university where they could attend presentations, watch demonstrations or participate in experiments. I was concerned with the question: *How to keep the visitors up-to-date?* This question was answered by taking advantage of the Vision of Crowds. All visitors were encouraged to capture photos and videos and to upload them to our system. Additionally, 10 students were engaged to act as particularly diligent visitors taking photos and videos of the event. The collected content was used to inform all visitors about ongoing activities at the event. Figure 4.3 shows the architecture of the system.

Visitors with mobile phones were able to upload their photos and videos as attachment to an email. In the subject line they could annotate the content with tags. For people that used digital cameras a number of upload stations equipped with card readers were available all over the campus area. All data uploaded to our system was stored on a file server.

The shared content was presented to all visitors in two ways. Several web kiosks, also spread across the campus, could be used to browse the content using an interface similar to the Flickr website. Additionally, we installed big video screens in places where a lot of people were expected. My video browser [20] was used to show automatically composed summaries of the community-contributed content on these

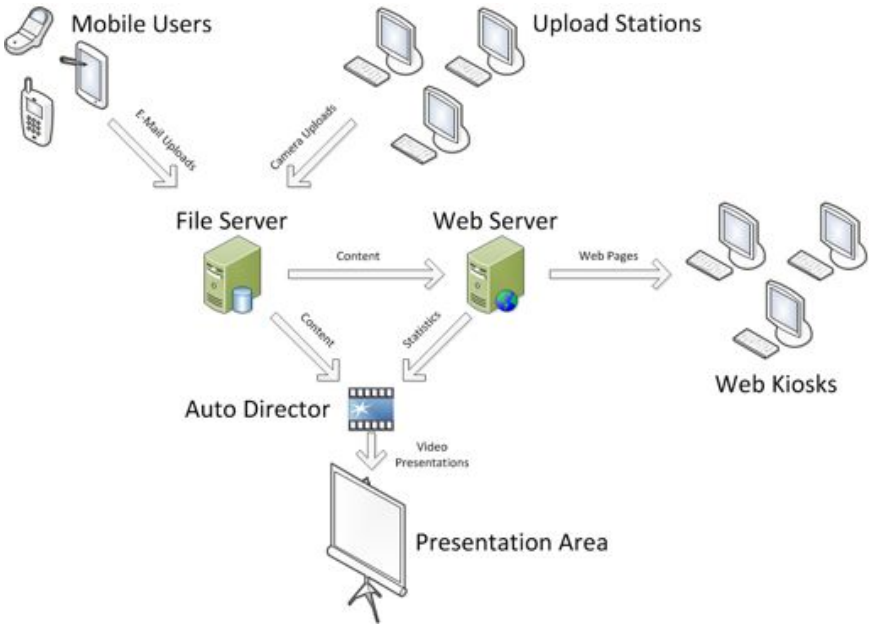

Figure 4.3: Architecture of the system used at the case study [19]

screens. Further details about the presentation are given in Chapter 5. Summaries that consisted of up to four parallel video streams were shown to the visitors.

The concept of the automatic director, which has been introduced in section 4.2.3, was implemented for this case study to automatically decide which content to show next whenever a presentation finished. The automatic director can be configured with predefined presentation profiles. Each profile consists of a combination of tags and the number of parallel streams to be shown. Based on several factors one profile is selected and a presentation is composed, which consists of photos and videos that were annotated with some of the tags defined in the profile. Listing 4.1 shows the pseudo code of the automatic director algorithm. First, usage statistics are collected from the web kiosks (number of views, the most often searched key words and user ratings) and the upload stations (most used tags). The most recent uploads are investigated to select a profile for the next presentation. The automatic director considers two factors for the content selection: (1) the most popular content in our web kiosks with respect to the selected profile and (2) the most recent uploads that match that

profile. Because of this combination, we are able to inform visitors about hot spots by showing current content and best-of content at the same time.

Listing 4.1: Main steps of the Auto Director

```
DO : statistics = getStatsFromWebKiosk()
     uploads = getRecentUploads()
     profile = selectProfile(statistics, uploads)
     presentation = compose(profile, statistics, uploads)
     showPresentation(presentation)
WHILE auto_director_alive
```

The decision which profile to use at a certain point in time is based on usage statistics and the most recent content uploads. The pseudo code in Listing 4.2 outlines this process. The most recent uploads of the last 30 minutes are investigated to identify one or more candidate profiles for the next presentation based on the number of uploads. If only one candidate profile is identified, it used for the next presentation. If more candidates are found, we use a history of all presentations already shown and the usage statistics from the kiosks to decide which profile to select. The history counts how often a profile has already been shown to avoid presenting only very popular projects all the time. With the help of it we are able to consider also less popular projects every now and then in order to provide a better overview of the event. If no candidate profile can be identified based on the uploads of the last 30 minutes, a predefined best-of profile is chosen to give a general overview of the event.

We made an empirical observation of the summaries shown by the Auto Director. The visitors behaved in a self-organizing way and reported mainly about places and situations, which are worth to be seen. As a result, the Auto Director composed presentations of these places, which again attracted further people to go there. A novel, very interesting, semantically rich, multi faceted view emerged.

Listing 4.2: Profile selection method

```
global history

function selectProfile(statistics, uploads)
  candidates = getCandidatesFromUploads(uploads)
  if count(candidates) == 1
    profile = getFirstElement(candidates)
  else if count(candidates) > 1
    candidates = considerHistory(candidates)
    profile = selectFromCandidates(candidates, statistics)
  else
    profile = getBestOfProfile()
  endif
  addToHistory(profile)
  return profile
end function
```

4.4 Interactive Event Summarization

In interactive image and video search the results only correspond to the query with a certain accuracy. The higher the matching accuracy for one photo or video is the more likely it is that it is shown under the top-ranked search results. If the accuracy for the searched item is clearly higher than for other ones, it is sufficient to use such an approach for presenting the results. In real-world scenarios often this is not the case. Very often there are different photos or videos in the result set that match the query with nearly the same accuracy. In such a case it might be hard to find the searched content efficiently. Users may have to look at many photos and videos of the result set one after another to get the essential information out of them. With the help of my approach I would like to support the users in handling large result sets of

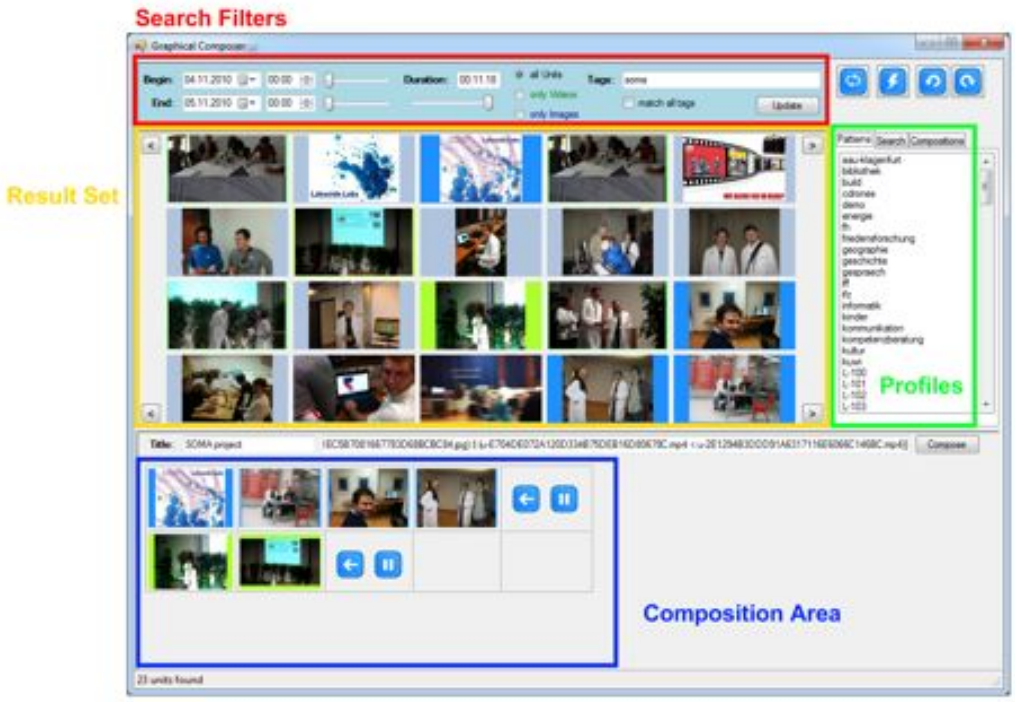

Figure 4.4: GUI for the composition of event summaries [19]

multimedia data. Several units may be examined in parallel. Therefore, I developed a GUI that intuitively allows composing individual multimedia presentations.

The live summarization of events raises the question how to inform people about a social event, which they might not have visited, after it took place? What could be better than having a look through the eyes of people that witnessed that event? During the presented case study 1444 photos and videos were uploaded to our system. As a consequence I have implemented a plug-in [19] for my video browser [20] that enables users to interactively compose event summaries. In Figure 4.4 a screenshot of the composition interface is shown.

By default, users can browse the whole content that has been uploaded. As different people are likely to have different interests, search filters can be used to narrow down the amount of available content (*Search Filters*). It is possible to define the time period when certain photos or videos were uploaded, the maximum duration

and whether only videos or only photos should be shown. Furthermore, it is possible to indicate tags that the searched photos and videos must be annotated with.

At the center of the window, the search results (*Result Set*) are shown in a grid view. The background color indicates whether a thumbnail represents a photo (blue) or a video (green). At the bottom (*Composition Area*) it is possible to manually compose event summaries by simply dragging and dropping thumbnails on the blue symbols. Dropping a thumbnail on the ← symbol adds the photo or video to the corresponding sequence, dropping it on the || symbol adds a new parallel stream to the presentation. The example shows a composition of two parallel streams. The first one consists of a sequence of four photos, while in parallel a sequence of two videos is going to be shown.

In addition to the manual composition it is possible to use the same profiles that are used by the Auto Director. In the right part of the window a list with all profiles is shown (*Profiles*). A double click on the name of a profile results in an automatic composition of an event summary. Only photos and videos are included that meet the defined search filters, like tags or the time span. Finally, by clicking on the compose button, the presentation is shown in the video browser.

4.5 Event Summarization Using Community-Contributed Content

If we query social media sharing platforms such as Flickr or YouTube to get informed about a certain social event, like the royal wedding of William and Kate in April 2011, we get a – usually extremely long – list of photos or videos. Even though the list is sorted corresponding to relevance, this is not a proper answer for such a question. We rather preferred to get a compact presentation of a predefined length, which gives us a summary, composed from the views of many people that have witnessed the event.

Therefore, I have implemented an algorithm for the summarization of real-life events based on community-contributed content. Additionally, different aspects regarding the capability of community-contributed content for this task are investigated. A preliminary study [57] showed that emotions play a critical role in the intentions of a people for taking images and videos. People tend to capture precious moments they want to remember. This observation led to this event summarization approach. If many people share those situations that are worth seeing in their opinion, a new and rich view of a real-life event should emerge.

A summary of a social event should consider the three aspects for a summary in [92]: (1) quality, (2) diversity and (3) coverage. (1) Photos and videos of poor visual quality should be not included into the summary. (2) Similar photos or videos should not be included more than once. (3) The resulting summary should cover the event as good as possible showing as many situations that occurred as possible. As the quality aspect has been intensively studied, I concentrate in this chapter on the two other aspects. During the generation of the summaries the focus is on the maximum diversity of the content.

This summarization algorithm may not produce the best summary possible, but it creates a representation that emerges from the information people provide when uploading the content to a social media sharing platform as well as from the most relevant and most popular contents related to a certain event. This view is usually very rich and contains a lot of interesting, even surprising elements. Of course, it may also contain garbage and even malicious content, but this is out of scope of this investigation. Maximal diversity in the generated summaries should be achieved. Coverage is not considered in the summarization algorithm, nevertheless, this aspect is included in the evaluation (with surprisingly good results despite explicit consideration).

4.5.1 Summarization Algorithm

A summary is built according to search terms, specified by the user, such as: *Royal wedding of William and Kate*. First the content is clustered, based on the available

textual descriptions. After that wrongly located content is filtered based on GPS information. At last, a summary is created from the remaining content. A flow chart, which illustrates these steps, is shown in Figure 4.5.

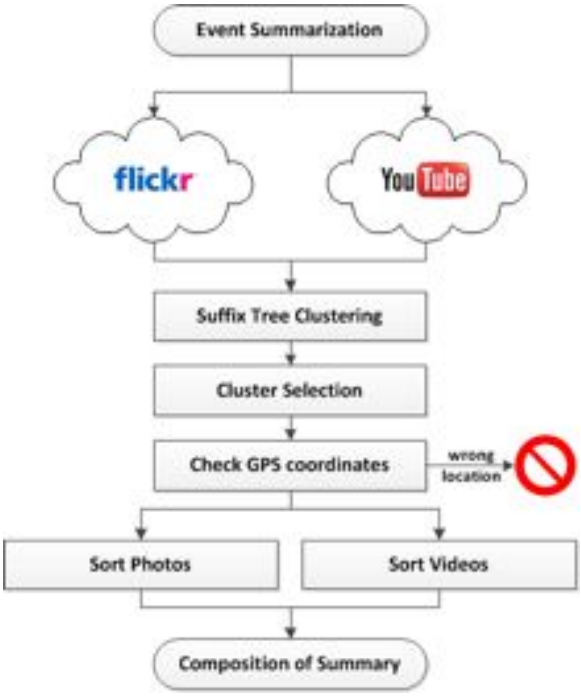

Figure 4.5: Flow chart of the summarization algorithm [18]

The composition of an event summary is influenced by six parameters:

1. Search terms to describe the event with keywords
2. Number of photos or videos to be shown in parallel
3. Maximum duration of the summary in seconds
4. Location
5. Start of the time span the content must have been produced
6. End of the time span the content must have been produced

The search terms are passed to Flickr and YouTube as text queries. The more comprehensive the query, the better focused the retrieved content will be. Therefore, different queries lead to different summaries. The results of both platforms are sorted by relevance. I rely on the relevance calculations of both platforms and do not perform my own ones. This is the default sorting mode of both platforms. If people use the web interfaces of Flickr or YouTube they also get the results sorted by relevance. A summary may consist of more than a single sequence of photos and videos. Figure 4.6 shows a screenshot of an event summary, which consists of four parallel streams.

Figure 4.6: Screenshot of an event summary [18]

The Flickr queries are limited to content that has been produced within the indicated time span. I decided to use this query option of Flickr to investigate how reliable this information can be used for a temporal alignment of the content. In the presented experiments all photos available on Flickr are considered. Copyrighted material is not excluded to get a better and more realistic representation of the Vision of Crowds. The YouTube API, unfortunately, does not allow stating a capturing or uploading date for the query. All videos are retrieved regardless of their timestamps. A post-processing step is performed where those videos are eliminated that do not

fit into the given time span. I am well aware that the timestamps of the photos and videos may be wrong or even missing. I am going to pay attention to this fact in the evaluation.

The queries do not return the photos and videos themselves, but only their metadata. For runtime reasons I decided only to gather the 5000 most relevant results from Flickr and YouTube. Actually, for the latter I had to be satisfied with only 1000 videos, because the YouTube API limits search results to this amount. The amount of photos and videos considered for the summary generation is still much larger than a user would manually examine when clicking through Flickr and YouTube results. Therefore, this limitation should be reasonable.

4.5.2 Clustering

In the next step the photos and videos are clustered based on their textural descriptions. For that purpose the text suffix tree clustering algorithm introduced in [123] is used. It has already successfully been applied to web document clustering and shows some interesting properties that can be exploited for this task.

1. **Relevance:** Documents relevant to the user's query are grouped separately from irrelevant ones.

2. **Browsable Summaries:** A concise and accurate description is provided for each cluster.

3. **Overlap:** Since documents may have multiple topics, documents can belong to more than one cluster.

4. **Snippet-tolerance:** High quality clusters are produced even if only text snippets of the documents are available.

5. **Speed:** Big amounts of documents are clustered fast.

6. **Incrementality:** Each document is processed as soon as it is received.

Relevance and speed is always of high importance in information retrieval. The snippet-tolerance is well suited for multimedia content, as for photos and videos typically only short descriptions are available. For each photo and video retrieved from Flickr or YouTube title, description and tags are extracted. This information is the input for the clustering algorithm. At the end, several clusters consisting of photos and videos are remaining. For each cluster a summary in form of a *dominant phrase* is provided by the clustering algorithm. For the content selection the largest cluster is chosen, of which the dominant phrase matches the search terms of the query.

4.5.3 Content Selection and Composition of Event Summaries

Photos and videos often have misleading descriptions regarding their location. The GPS coordinates of the content are investigated to get this problem better under control. The location indicated in textual form is translated in GPS coordinates using the Google Geocoding API[1]. Using the retrieved GPS coordinates and the level of detail (country, region, city or street) content can be eliminated, which has been produced in a wrong place. If ambiguities are possible (e.g. a city named Paris exists not only in France, but also in Texas) the location must be specified precisely (e.g. "Paris, France" or "Paris, Texas"). Otherwise wrong content may be included in the summary.

The selection of photos for the summary is based on the number of how frequently a photo has been viewed on Flickr. The selection of videos is based on the user ratings (up to 5 stars), the number of views and the number of *likes* a video has on YouTube. For each event summary the content is selected in such a way that the amount of time when photos are shown and the amount of time when videos are played are approximately equal. While videos have a natural length, a default duration of 7 seconds is defined for still images in the summary. In a single sequence this may be too long to show a single image, but as soon as more than one sequence is shown in parallel the viewers need more time to look at all photos. For example, for a

[1]Google Geocoding API: `http://code.google.com/apis/maps/documentation/geocoding/`

video with a duration of 28 seconds four photos are added to the summary. This ratio is automatically adapted if the number of either the photos or the videos is too low. It may happen that no videos are included in a summary, because the selected cluster does not contain videos at all or the length of the contained videos exceeds the maximum duration of the summary.

One important aspect of the summarization of content is to avoid redundancy [105]. The algorithm relies on visual image features to identify redundant photos. Each image selected as a candidate for a summary is matched against all other photos that are already in the summary. If the distance to a photo in the summary is too low the candidate image will not be added. For the estimation of the visual similarity the Color and Edge Directivity Descriptor (CEDD) [11] is extracted from each photo. The CEDD can be extracted fast and it showed good results in an evaluation of different image features for video summarization [42].

4.5.4 Summary Format and Presentation

Finally, when all photos and videos are selected the whole contents are sorted based on their creation timestamps. With this simple approach I want to investigate how good timestamps are suited to make a temporal alignment of the content.

In the resulting event summary the videos are played first and then slide shows of the photos are shown. I am of the opinion that the viewers get a good impression and an overview by watching the videos first, while photos are better suited to cover certain aspects in detail that the videos may miss.

ViNo [94] is used for the formal definition of event summaries. With the help of ViNo it is possible to define the temporal as well as the spatial alignment of multimedia units.

The generated summaries can be watched with my Video Browser [20], which is depicted in Figure 4.6. This video browser is able to interpret ViNo expressions and allows showing several streams of videos and photos in parallel. The audio playback

is selected from one of the presented videos by default or by mouse-over on one of the videos. More details on the presentation of compositions are given in Chapter 5.

4.5.5 Evaluation

For the evaluation I chose four well-known social events that took place in the last three years: (1) the inauguration of Barack Obama[2], (2) the Royal Wedding of William and Kate[3], (3) the FIFA World Cup Final 2010[4] and (4) the UEFA Champions League Final 2011[5]. All four events took place on one single day, were attended by several thousands of people and attracted the attention of millions of people around the world.

The same algorithm was used for all four summaries. It was not tuned according to the events. All event summaries in this evaluation consist of 4 parallel streams and have a maximum duration of 5 minutes. The time span I used for the queries starts with the day the event took place and ends one month after that. Other investigations showed that even a time interval of 7 days is sufficient [52]. Screen captures of the four composed event summaries are available online[6].

Table 4.1 lists the *Search terms* that were used as input for the summary generation and gives details about the retrieved content. I tried to use as few search terms as possible to describe the events, because people also tend to use only a few terms when searching for multimedia content online [107].

The same query, which is used for the summary generation, has also been used to query the Flickr (*Flickr results*) and the YouTube (*YouTube results*) website to get a first impression of the available content. For the first two queries much more

[2]Inauguration of Barack Obama: `http://en.wikipedia.org/w/index.php?title=Inauguration_of_Barack_Obama&oldid=439374433` (Permalink)

[3]Royal Wedding of William and Kate: `http://en.wikipedia.org/w/index.php?title=Wedding_of_Prince_William_and_Catherine_Middleton&oldid=440475841` (Permalink)

[4]FIFA World Cup Final 2010: `http://en.wikipedia.org/w/index.php?title=2010_FIFA_World_Cup_Final&oldid=439386816` (Permalink)

[5]UEFA Champions League Final 2011: `http://en.wikipedia.org/w/index.php?title=2011_UEFA_Champions_League_Final&oldid=440623020` (Permalink)

[6]Demo videos: `http://soma.lakeside-labs.com/?page_id=279`

Search terms	inauguration obama	royal wedding	fifa world cup final 2010	champions league final 2011
Flickr results	59643	47372	2535	1529
YouTube results	15800	52500	547000	186000
Photos/Users selected	1062/182	1516/343	668/81	161/22
Videos/Users selected	1/1	211/211	114/90	83/72
Photos with GPS	333	437	333	42
Videos with GPS	0	7	7	0
Wrong location	81	211	160	1
Photos/Users in summary	168/51	73/28	81/17	83/14
Videos/Users in summary	0/0	5/5	4/4	5/5

Table 4.1: Details about community-contributed data related to certain social events

photos can be retrieved from Flickr than for the two soccer matches. The reason for that is that more specific text queries were used for the two soccer matches consisting of 4 and 5 terms, compared with only 2 terms for the first two queries. The more specific a query is the less results are returned from Flickr. Interestingly, for the two soccer matches a huge amount of videos is available. A closer examination shows that people played these matches also on their gaming consoles and published videos of that computer games online.

The event summary algorithm originally included the 5000 most relevant Flickr and the 1000 most relevant YouTube results. Finally, even a smaller subset – as produced by the clustering – is used for the content selection. The rows *Photos/Users selected* and *Videos/Users selected* list how many photos and videos were included in the final cluster for the summary generation and how many distinct users uploaded these contents. It can be seen that several photos are selected from each included Flickr uploader, while in most cases the included YouTube videos have different users.

In the created summaries 3 to 6 photos of a single uploader (*Photos/Users in summary*) are included. Each video in these summaries (*Videos/Users in summary*) has a single uploader. The summary of the inauguration of Barack Obama only consists of photos. The cluster selected for the summary only contains one video of his oath, but its length exceeds the maximum duration of the summary. In general, these summaries include content from a variety of users, thus these summaries are really conveying a broad view of people that witnessed the selected events.

The retrieved data shows that the available GPS data provide only a strongly limited support to estimate the location where the content was produced. For only 25 – 50 % of the selected photos (*Photos with GPS*) are the GPS data available and videos (*Videos with GPS*) hardly have this data associated at all. Nevertheless, many photos could be filtered that were taken in a wrong location. The relatively high amount of photos excluded due to wrong semantic location (*Wrong location*) can be easily explained. The events chosen for the summaries were broadcasted all over the world. The excluded content was produced by people somewhere else on the world. In most cases people celebrated parties to follow the original event in a group on TV. The content produced at those parties was annotated with textual descriptions related to the original event. Therefore, it was initially included in the results sets retrieved by Flickr and YouTube.

The coverage of the created summaries is compared against a manually obtained ground truth. The most important *situations* of the chosen events were figured out with the help of Wikipedia articles (see the footnotes 2-5). For each event a corresponding set of situations was identified. A situation may be a temporal happening, such as *exchange of the rings*, a location, such as the Westminster Abbey or even persons, such as *Prince Harry*. Table 4.2 lists the identified situations for all four events. Further information about these situations can be obtained from the Wikipedia articles.

I decided to rely on Wikipedia, because it is difficult to find an objective evaluation metric for the quality of summaries. Summaries are always somehow based on

subjective opinions as [77] showed. Wikipedia articles usually have several authors, who perform discussions and have to agree on the text of the article. Therefore, Wikipedia articles convey the common opinion of a crowd of people. I take advantage of that common opinion to get a more objective ground truth for the evaluation of the coverage of the generated event summaries.

I compared my event summaries with a standard web search on Flickr and YouTube. As the evaluated summaries have a duration of 5 minutes, I have limited the number of Flickr and YouTube results to amounts that could approximately be browsed in that time span. The first 120 photos from Flickr and the first 20 videos from YouTube are investigated for each query. If I compare the coverage of the generated summaries with the Flickr and YouTube results in the following parts of this evaluation, I always refer to result sets of that size (indicated by *Flickr* resp. *YouTube* in the following diagrams).

The results are shown in Figure 4.7. In all cases the first Flickr results only include few situations of interest. The reason for that is that people tend to photograph themselves when visiting an event. Therefore, a lot of images show visitors of the event and only few photos show situations as they were identified based on the Wikipedia entries. Except for the inauguration of Obama the YouTube results show more interesting situations than the Flickr results. The event summarization algorithm shows in all cases the best performance. It includes as much situations as Flickr or YouTube or even more.

If content is examined regardless of the searched situations, it can be recognized that the precision of the Flickr results is high. They include a high amount of content that is related to the searched events. Figure 4.8 shows the percentage of true positive photos and videos in the Flickr and YouTube results as well as in the event summaries. For the latter I distinguish between photos and videos. A photo or video is regarded to be a true positive if it is somehow related to the event. The Flickr results contain a lot of true positives, which also has a positive effect on the photos in the summaries. Except for the Champions League final the YouTube results have a

Inauguration Obama	**Royal Wedding**
1. United States Capitol	1. Westminster Abbey
2. Music live performances	2. Bride (Kate)
3. Invocation by pastor	3. Groom (William)
4. Aretha Franklin singing	4. Pippa Middleton
5. Oath of Vice President	5. Prince Harry
6. Oath of Barack Obama	6. Queen Elisabeth II.
7. Inaugural address	7. Young bridemaids
8. Prayers	8. Pageboys
9. Departure of former president	9. Arrival of Kate
10. Signing of first orders	10. Exchange of rings
11. Luncheon	11. Lesson
12. Parade	12. Sermon
13. Inauguration balls	13. Leaving Westminster Abbey
14. National prayer service	14. Return to palace in coach
15. Oath of office	15. Lunchtime reception
	16. Appearing on balcony
	17. Harpist performance
	18. William & Kate leaving with car
	19. Private dinner
	20. Wedding cake
	21. Merchandise
	22. Broadcasting
World Cup	**Champions League**
1. Soccer City Stadium	1. Wembley Stadium
2. de Jong's kick against Alonso	2. Chance Hernandez (ManU)
3. Chance Robben (NED)	3. Chance Villa (Barca)
4. Chance Sneijder (NED)	4. Chance Villa (Barca)
5. Chance Ramos (ESP)	5. Goal Pedro (Barca)
6. Red card Heitinga (NED)	6. Goal Rooney (ManU)
7. Goal Iniesta (ESP)	7. Chance Messi (Barca)
8. Award ceremony	8. Chance Messi (Barca)
	9. Goal Messi (Barca)
	10. Chance Messi (Barca)
	11. Chance Xavi (Barca)
	12. Goal Villa (Barca)
	13. Chance Rooney (ManU)
	14. Chance Nani (ManU)
	15. Award ceremony

Table 4.2: Important situations of four social events

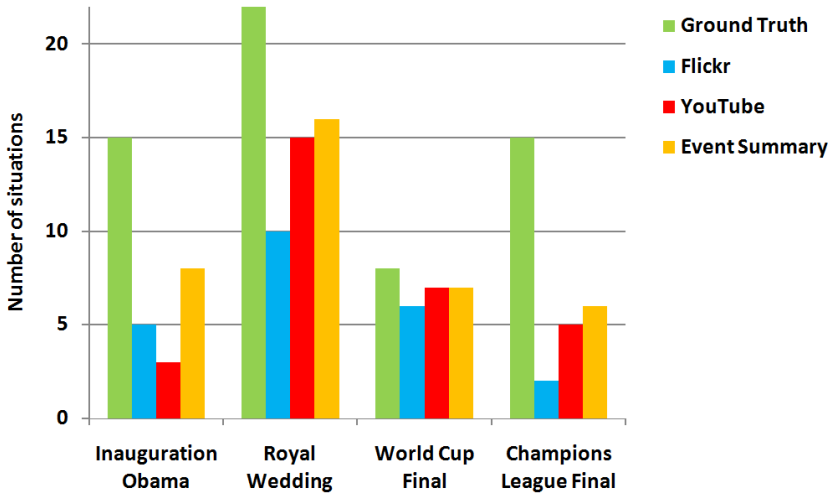

Figure 4.7: Comparison of situations found (coverage)

lot of false positives, although only the 20 most relevant results returned by YouTube are considered. The event summarization algorithm also includes false positives in the summaries, but the ratio of true positives is much better than the one of the YouTube results. This is an effect of the suffix tree clustering of the content. As the biggest cluster is chosen, which is related to the query, it is more likely that this cluster includes relevant content. Note that false positives include photos and videos, which are not wrong, rather strange. For example, if some people record the movements of the police at the royal wedding (as they did indeed), this is topic for a non-technical discussion, whether or not these images are misplaced.

The comparison of the coverage shows that quite a lot of the defined situations of interest are not included in the summaries nor in the Flickr and YouTube results. Therefore, a closer look at the situations found is taken. Figure 4.9 shows the situations detected for the inauguration of Barack Obama. It can be noticed that a lot of photos are showing the parade (situation no. 12) after the inauguration. That was somehow expected, because the parade was watched by a lot of people along the track and thus a lot of photos have been made. For the other situations it can be stated that people especially took photos of the highlights, like the oath of Obama

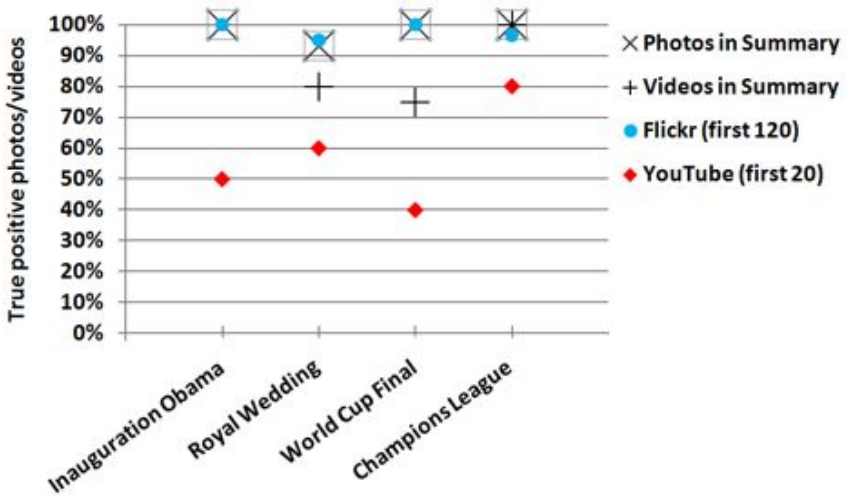

Figure 4.8: Amount of true positive photos or videos

(6), his inaugural address (7) or the departure of the former president Bush (9). Also the society events like the luncheon (11) and the balls (13) seem to attract people. The oath of the Vice-President (5), prayers (8) or events that took place in the office of Obama, like the signing of the first orders (10) or his second oath (15) are not covered by the content received from Flickr and YouTube.

Figure 4.10 shows the identified situations of the royal wedding in detail. As it can be seen the involved people like Kate (2), William (3), Pippa (4), Prince Harry (5) or the Queen (6) get a lot of attention. Also the appearing on the balcony (16) or situations that took place in the streets or in front of the church (9, 13 and 14) are included often. The reason is again that for public situations a lot of content is produced, while for private ones like the family celebrations (15) or the private dinner (19) in the evening nothing can be found.

I also wanted to investigate events where the interesting situations may be clearer. Therefore, I decided to investigate event summaries of two soccer matches that attracted the attention of millions of people around the world. The identified situations for the two games are shown in Figure 4.11 and Figure 4.12. For both games it can be stated that all goals are included in the created summaries, but nearly all chances,

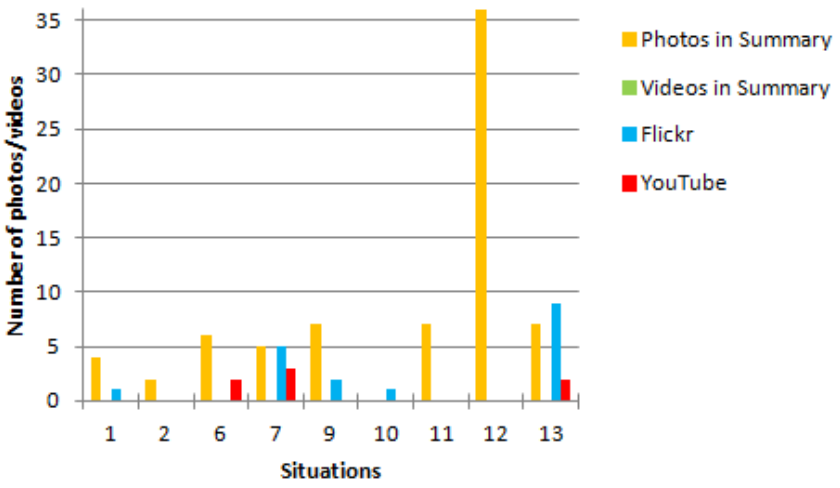

Figure 4.9: Results for the Inauguration of Obama

which did not result in goals, are missed. In addition to the goals both summaries also include a lot of situations showing the venue and the award ceremonies of the winning teams.

Regarding the temporal alignment of the content it must be stated that the timestamps of the content are not sufficient for good ordering of the content. By simply watching the generated summaries it can be seen soon that the content is mixed up temporarily in all summaries. It seems that people do not care about their cameras having correct date and time settings. Nevertheless, this could change, if people notice in the future that innovative tools can make good use of this information.

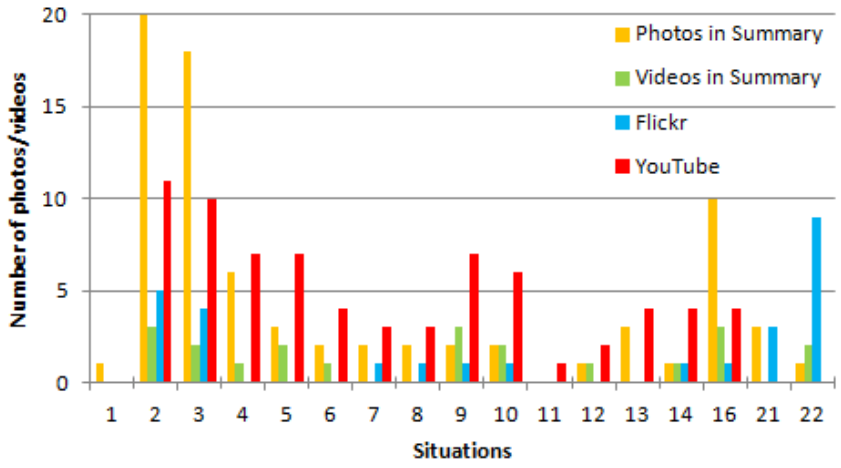

Figure 4.10: Results for the Royal Wedding of William and Kate

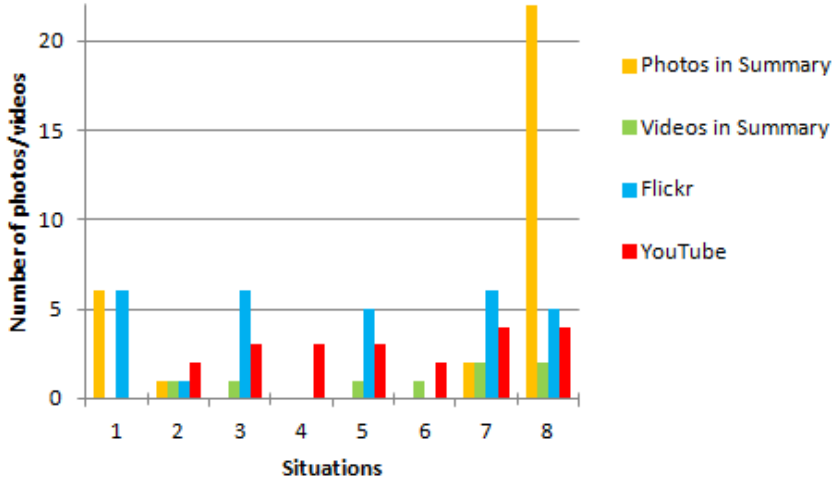

Figure 4.11: Results for the FIFA World Cup Final 2010

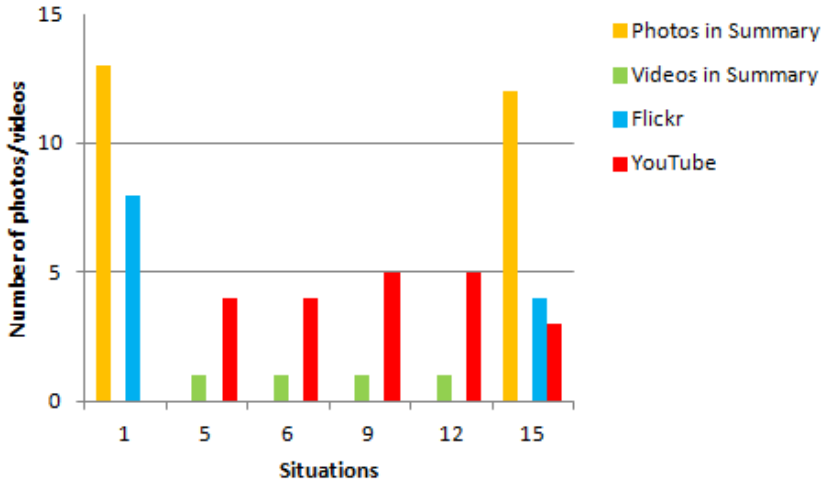

Figure 4.12: Results for the UEFA Champions League Final 2011

CHAPTER 5

Non-Sequential Multimedia Presentation

In the fields of image and video retrieval many efforts are done in improving the retrieval process itself. The problem is that a gap will always remain between the intention of the user, which is expressed by a query, and the results that are regarded to be relevant. In many cases users only use fuzzy queries to specify their requests [107], which leads to large result sets. It is a time consuming task to browse them. Even if users formulate precise queries, it is hard to find an appropriate ordering of the results. Different users, which state the same query, may have different intentions in mind when searching for multimedia content. In such a case it is not the solution to provide the same ordering of results to all users. The presentation of search results must be adjusted to the intentions of a user [44].

In this thesis a new approach is introduced that allows people to individually compose their own video presentations consisting of video segments and photos from different sources. Composing video presentations does not necessarily mean putting videos together in a sequence in order to show one after another. Users can even watch several photos, videos or respectively sequences of them in parallel. Instead of

browsing through static presentations of search results, users are enabled to watch content in a way that meets their individual requirements. This flexibility in the presentation of content needs also presentation means that are able to cope with an individual arrangement of multimedia content.

Therefore, I present an approach for the formal description of multimedia presentations in this chapter. It allows a specification of the temporal and the spatial alignment of multimedia content in a video browsing application. A formal language for the temporal and spatial alignment of multimedia content is not really an innovation. SMIL[1] is a well known example for such a language, which exists already for many years. The decision to use an own formalism for this task originates from the research project SOMA, where the approaches presented in this thesis contributed to. In SOMA the formal description of presentations is also used for the optimization of the transport of the content over a network [94]. If the delivery mechanisms are aware of how the presentation looks like, decisions can be made which units to deliver first and which ones may also be accepted in lower quality, e.g. if a unit is only shown on a small part of the screen.

Furthermore, a video browser for effective browsing of multimedia content is also presented, which is able to interpret and display presentations that have been formally described with my approach. The video browser enables users to explore video content in a hierarchical, non-sequential way. Using hierarchical browsing mechanisms, video content can be displayed in a well-arranged way, helping users to get a better overview of the relations between certain video segments and to find searched segments faster. Approaches presented so far (section 2.2.3) only use key frames for visualizing the content of a video. In my opinion, a lot of important information about a video gets lost, which may be useful to explore the content of a video (audio stream, volume, motion intensity), if a user can only look at key frames. Therefore, in my approach users are able to interact with several video players at the same time. Exploring

[1] W3C SMIL Homepage: `http://www.w3.org/AudioVideo/`

different video segments at the same time they are able to perceive the content in a better way compared to just looking at key frames.

5.1 Formal Description of Multimedia Presentations

For the formal description of compositions an own formalism called *Video Notation (ViNo)* [94] is used. Photos and video segments are called units. Each unit has a unique identifier, which is used to access the corresponding photo or video. In ViNo expressions only these identifiers are used to reference units. In cases where the exact identifier is unknown a question mark (?) can be used as placeholder.

ViNo provides a big flexibility for the composition of video presentations. It is a multipurpose multimedia language, which can be used to define even complex compositions in a compact way. ViNo can be used to describe the delivery of units over the network (see more on this in [7]), but the basic concepts fit as well for the temporal and spatial description of multimedia presentations. Any units in any order under arbitrary QoS constraints may be composed to a new video.

5.1.1 Temporal Alignment of Units

The general syntax and semantics of ViNo are given by the following definitions relying on [94].

Definition 4 *A composition is an expression defined inductively by these rules:*

1. *A single unit is a composition.*
2. *Let $c_1, c_2, ..., c_n$ with $n \geq 2$ be compositions, which have already been defined. Then, the following expressions are compositions, too:*
   (a) *$[c_1||c_2||\ldots||c_n]$ is called a parallel composition.*

(b) $(c_1 \leftarrow_{Q_1} c_2 \leftarrow_{Q_2} \ldots \leftarrow_{Q_{n-1}} c_n)$ *is called a sequential composition. A symbol* Q_i*, where* $i = 1, \ldots, n-1$*, represents an optional QoS parameter.*

Definition 5 *Semantics.*

1. *If* $c = c_1||c_2$ *for some compositions* c_1 *and* c_2*, then the playback of* c *starts as soon as* c_1 *or* c_2 *starts, whatever is earlier; and it is finished when the playback of both* c_1 *and* c_2 *is completed.*

2. *If* $c = c_1 \leftarrow_Q c_2$ *then the playback of* c_2 *must not start before the completion of* c_1*; the QoS predicate* Q *applies to the time period between completion of* c_1 *and completion of* c_2*.*

3. *The semantics of* $c = c_1 \leftarrow_{Q_1} c_2 \leftarrow_{Q_2} c_3$ *is defined as that of* $(c_1 \leftarrow_{Q_1} c_2) \leftarrow_{Q_2} c_3$*.*

A single unit is by definition also regarded to be an atomic composition. Therefore, ViNo allows to recursively define compositions of compositions. For example, units consisting of single frames can be composed to shots. Furthermore, shots can be composed to scenes and scenes, which may originate from different videos, can be composed to a new video. In Chapter 4 I presented two examples, one for a sequential composition and one for a parallel composition. The sequential example in Figure 4.1 can be expressed with ViNo as

$$u1 \ \leftarrow \ u2 \ \leftarrow \ u3 \ \leftarrow \ u4$$

The ViNo expression for the parallel composition in Figure 4.2 is

$$(u1 \ \leftarrow \ u5) \ || \ (u2 \ \leftarrow \ u6 \ \leftarrow \ u8 \ \leftarrow \ u10) \ || \ (u3 \ \leftarrow \ u7 \ \leftarrow \ u9) \ || \ u4$$

The event summaries presented in Chapter 4 are also described using ViNo expressions. Each event summary is a ViNo expression that consists of one or more sequences to be shown in parallel. The ViNo expression that describes a summary is stored in a separate file next to the photos and videos, which are contained in the summary.

5.1.2 Spatial Alignment of Units

The temporal alignment of units defines in which order the content has to be presented. In case of a parallel presentation, the spatial alignment of units on the screen must additionally be specified. Again, the multimedia language ViNo can be used for this task. The spatial alignment of units is done in the same expression where the temporal alignment is described. Only additional square brackets are needed to group content and to describe spatial relations between different units. Therefore, the formal description of presentations still remains compact. Content that should be shown on the same spatial level must be grouped with surrounding square brackets. Spatial relations can be expressed with nested terms. To explain this concept in detail four examples for the spatial alignment of content are shown in Figure 5.1.

In example (a) a parallel presentation of nine units is illustrated. Each unit is presented at the same size. All nine units are organized in three rows, which form three presentation levels. Therefore, each row is grouped with surrounding square brackets. The resulting ViNo expression is:

$$[u1 \parallel u2 \parallel u2] \parallel [u4 \parallel u5 \parallel u6] \parallel [u7 \parallel u8 \parallel u9]$$

A spatial arrangement where a different size for the presentation of units is used is shown in example (b). The whole screen is divided into two presentation levels. On the first level one unit can use the whole space, while on the second level three units must share the available space. The following ViNo expression describes this view:

$$[u1] \parallel [u2 \parallel u3 \parallel u4]$$

The third example (c) presents a view that can be described with nested presentation levels. Three additional presentation levels are defined within a level:

$$[u1 \parallel [u2] \parallel [u3] \parallel [u4]]$$

These examples show only single units to be shown in parallel. By definition each unit can be a composition as well, thus each unit in the examples in Figure 5.1 can

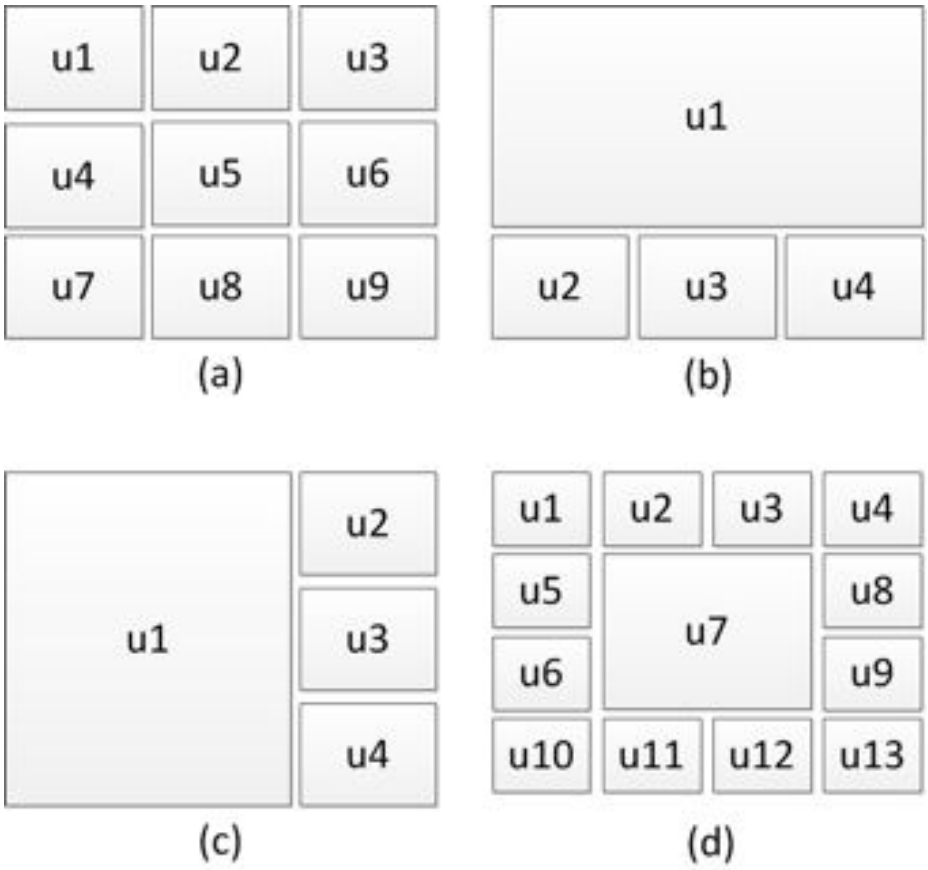

Figure 5.1: Schematic examples for the spatial alignment of units

be replaced by a sequential composition of units in order to define more complex compositions. Furthermore, a nesting of spatial descriptions is also possible, like it is shown in example (d). The corresponding ViNo expression is:

$$[u1 \;||\; u2 \;||\; u3 \;||\; u4] \;||\; [[u5] \;||\; [u6] \;||\; u7 \;||\; [u8] \;||\; [u9]] \;||\; [u10 \;||\; u11 \;||\; u12 \;||\; u13]$$

The exact presentation size of a unit is not specified by this formal description of the spatial alignment of content. Only spatial relationships are expressed. How these relationships are presented is up to the application that interprets and displays a composition. This provides a big flexibility in the development of such applications. While a two-dimensional video player may interprets different presentation levels as rows, like in the examples shown, a three-dimensional video player could display presentation levels completely different in the 3D space. The players must not violate the temporal ordering of the content and the spatial relationship and grouping of content, but the exact size of the presented content and the alignment of different, unrelated presentation levels can be realized in a different way in different players. ViNo is conceptually similar to SMIL, it is, however, much simpler and is much more compact. Nevertheless, it would be easy to translate a ViNo expression into a SMIL presentation, as the presented concepts of ViNo are fully covered by the SMIL specification. For the work done in the context of this thesis I preferred the easy to use and compact syntax of ViNo.

5.2 Non-Sequential Video Browsing Without Content Analysis

Usually, video browsing solutions are based on content analysis of the underlying video. Almost all proposed solutions use shot segmentation as a first step and provide browsing mechanisms based on the shot structure. Content analysis – of a newly stored video file – takes quite an amount of time. In some scenarios, e.g. when only a quick overview of the content of a video is required, it is an overkill to perform

a deep content analysis. If single shot videos are browsed, shot detection does not help at all. Examples for single shot videos can be typically found in surveillance applications. For such scenarios it is much better to provide quick, yet powerful, interactive navigation means.

I propose a novel approach for instant video browsing that requires no content analysis at all. The presented application can immediately and efficiently be used for scenarios where a quick inspection of a newly recorded video is required. While video retrieval tools typically provide better content-based search functions, they first need to perform a deep content analysis step requiring a lot of processing time (often in dimensions of several hours). Users, who just quickly want to get an overview of a new video or to find some specific segments in it, do usually not accept long delays before they can use the tool. From a preliminary user study [84] it is known that users in such situations rather employ common video players for interactive browsing although they provide only poor navigation features. The tool presented in this chapter has been designed to provide a real alternative to common video players for such situations.

5.2.1 Hierarchical Video Browsing

Every video is divided into as many parts of equal length, as there are video windows opened on the screen. The dimension of the window matrix (n) can be increased or reduced by the user with a single click. Two different views are available for browsing the content: a *parallel* and a *tree based view*. With both of them it is possible to traverse the content in a hierarchical way down, until the frame level is reached, and up again.

An example of the parallel view is given in Figure 5.2, where a news video is divided into nine parts of equal length. If one of the parts is selected by clicking the right mouse button, the user gets down into a deeper level with more details. That means that the selected part is divided into n parts of equal length again. To get a coarser view again, it is possible to go back to a higher level. The parallel view

Figure 5.2: Parallel Browsing View [20]

only shows one level of the browsing hierarchy at a glance. In contrast the tree based view shows all levels simultaneously in a treelike structure, thus the context of the video windows is better preserved. Figure 5.3 shows an example[2] of the tree based view with a highlight video of a soccer match. Each row represents one level of the browsing hierarchy. The navigation path in the example shows a scoring chance of one of the teams. The browsing history from the top to the bottom level is preserved by coloring the selected video parts on each layer with a green border. This should help the user to quickly find an alternative browsing path. If a part is selected, a new row that shows only that part is added to the tree. Browsing through the content of a video this way can be compared with navigating through a tree structure. Having found the required scene the user may select the starting point of it as the new root. This enables the user to quickly locate a number of interesting scenes in a video.

[2]The red lines between the horizontal window rows have been added to the screen shot for a better visualization of the tree-based browsing concept.

Figure 5.3: Tree-Like Browsing View [20]

The tree-like concept has also been realized in a three-dimensional interface using a carousel representation for the different levels of the tree [85]. It is an advantage to display the tree-like view in a 3D space because a better overview of the browsing history can be preserved with such a representation. A screenshot, which shows this novel view, is shown in Figure 5.4. In addition to the two-dimensional interface it has the ability to show several parts of one level at the same time, while the two-dimensional interface can only display one part of the video on each level.

5.2.2 Parallel Video Browsing

Beyond hierarchical browsing, the tool also offers parallel playback. All shown parts or only selected ones can be watched in parallel and the playback speed can be adjusted. The slider at the bottom of the container window can be used to scroll through selected videos in parallel. The users can get an impression of the whole video in a fraction of the overall duration. The audio playback is only enabled for

Figure 5.4: 3D interface that implements the tree-like browsing concept [85]

one single selected video window (where the mouse points at). The ability to play the audio stream only of parts regarded to be interesting helps the users in getting a better browsing experience.

Regarding the performance of the tool it can be stated that at least nine 720p videos can be decoded and played in parallel with normal playback speed of 25 fps on a standard desktop computer (Dual Core Processor with 2.8 GHz, 4 GB RAM).

5.2.3 Additional Features

The introduced views are not limited to single video files. They can be applied to small video collections as well. Opening a video archive adds an additional level to the browsing hierarchy, which means that on the first level all videos of the selected collection are shown, serving as starting point for hierarchical browsing of the whole video archive.

Another feature of the video browser is that segments of interest can be selected and stored in a playlist for later use. Moreover, selected segments can also be exported as a single file, which can be opened with a common video player. Thus, this video browser is also a ”poor men’s” video cut tool.

My video browsing tool offers a simple plug-in architecture. With new plug-ins it can be extended by further presentation views and also by video analysis and video processing algorithms. By combining different plug-ins it can be easily adjusted to the needs of the users and the peculiarities of different video domains.

One plug-in for the presentation of ViNo expressions has been developed in the context of this thesis. It parses a ViNo expression from a text file and loads and displays units as specified. The content must be placed in the same directory, where the text file with the ViNo expression is located. Regarding the spatial alignment of content the plug-in displays the different presentation levels as specified in the examples in Figure 5.1.

CHAPTER 6

Conclusion

This dissertation is concerned with the non-sequential decomposition, composition and presentation of multimedia content. In this chapter the presented topics are summarized and the thesis is concluded by discussing the results and future research directions.

6.1 Concluding Remarks

In the last decade a transition occurred how people are dealing with multimedia content. While people carefully selected the scenes to be captured a decade ago, today people tend to capture photos and video more spontaneously. Huge amounts of content are produced and shared on the web. It takes a lot of time to browse such web-based photo or video collections manually. In order to save time, non-sequential usage patterns of multimedia content emerged. Photo collections or videos are not watched as a whole anymore, but only photos or video segments regarded to be

interesting to a person. In this thesis I investigated several questions that come up in the context of non-sequential consumption of multimedia content.

A comprehensive survey of video scene segmentation algorithms published in the last decade was presented. I classified them into seven different classes on three abstraction levels: (1) feature level (visual-based, audio-based, metadata-based), (2) algorithmic level (graph-based, statistics-based), (3) conceptional level (film-editing-rule-based) and (4) hybrid approaches that combine methods from all three levels. Additionally, use cases for the presented approaches were identified. I considered also use cases which have not been taken account by the authors of the original papers.

Besides giving a qualitative overview, a quantitative comparison of the presented algorithms was also desirable. Unfortunately, no unified test set and evaluation method for scene segmentation is available. Therefore, it is hard to make exact comparisons. All solutions have their individual strengths and weaknesses. Recent algorithms achieve good results. Using the presented classification scheme and the identification of appropriate use cases for the presented algorithms, this survey can be used as a guide where to start with future research activities in video scene segmentation. Better comparable results with unified test sets and evaluation methods are needed. In most cases the complexity of algorithms is not stated. I regret this as a poor methodic practice not to make algorithms comparable in their performance.

An own algorithm was presented that identifies recurring patterns of motion sequences within a video stream. In the sports domain motion patterns that can be found throughout a video seem to comply with important semantic scenes. This interesting fact can be used to automatically segment video streams into high-level scenes that are typically larger than shots and that contain more meaningful information. The evaluation has shown that the algorithm delivers promising results.

Furthermore, I introduced a novel idea of summarizing real-life events based on community-contributed multimedia data in this dissertation. People can gather information about a social event by watching the content produced by other people

that witnessed that event. A new and richer view emerges from the different views of different people. We call this principle "The Vision of Crowds".

For the creation of live event summaries a case study has been conducted, where content produced by visitors was used to inform all people about ongoing activities and hot spots at that event. Additionally, I implemented a user interface for the semi-automatic generation of event summaries. Empirical observations showed that people tend to capture mainly situations that are interesting for them. Other people with the same interests can be guided by that information.

The initial results led to further investigations of this field in a bigger context. An algorithm for the summarization of real-life events based on community-contributed multimedia content from Flickr and YouTube was developed. I composed four summaries of events that attracted a lot of people during the last three years. The coverage of my summaries was evaluated by comparing them with Wikipedia articles that report about the corresponding events. This innovative evaluation technique allows us to identify the important happenings of social events without doing manual observations of these events, but by relying on the common opinion of a group of people that created and edited the corresponding articles. In addition to the evaluation of the summaries, several characteristics of community-contributed content with respect to event summarization were investigated. The composed summaries show a good coverage of interesting situations that happened during the selected events. Nevertheless, some challenges remain, like the correct temporal alignment of content or the identification of malicious content.

Finally, I introduced a novel concept for the presentation of photos and videos from different sources. A compact formalism (ViNo) is used for the description of the temporal and the spatial alignment of multimedia content on the screen. This formal description is very flexible, because only the temporal and spatial relationships between contents are strictly defined. The concrete realization is up to the video player. Therefore, a composition can look completely different in two different applications, e.g. a 2-dimensional and a 3-dimensional interface.

A video browser was introduced, which is able to interpret and display compositions defined with ViNo. It focuses on easy to use video browsing concepts for instant usage. Beside the presentation of compositions it can be used for the normal playback of single videos or video collections. It offers a parallel view, which can be used to get an overview of the content of a video by using parallel playback or parallel scrolling. A tree-like view provides mechanisms for quickly exploring different search paths within a video and thus it is better suited for searching for a particular scene. Both approaches refrain from content analysis and work for single-shot videos as well. They provide a flexible user interface for non-sequential hierarchical video browsing and are suggested particularly for situations, in which video analysis is not adequate (e.g. due to lack of rich semantics) or would take too much time.

All three covered research topics are my answer to the question how to support people in the non-sequential usage of multimedia content more effectively and efficiently in future multimedia information systems. An intelligent indexing of content allows a faster access to interesting parts of photo collections or videos. The usage of context information for the composition of new video streams consisting of units from different sources creates a new, compact view of a topic. A flexible presentation concept allows arranging content in such a way on the screen that best fits the needs and intentions of a user.

6.2 Future Research Directions

I identified some important aspects and trends that are just changing our daily life. Therefore, even well investigated areas like scene segmentation need to be investigated under a new point of view. It seems that major improvements in accuracy are not possible in automatic video scene segmentation. Current approaches reach already a high accuracy. The question is whether it is worth to put much effort on improving and tuning algorithms to achieve minor, hardly recognizable improvements. The accuracy could be enhanced considerably by incorporating human knowledge

in video segmentation approaches. Interactive segmentation of videos by combining automatically retrieved scene candidates with an interactive segmentation tool may be a possible solution. More powerful than incorporating only one user into the segmentation task is to take advantage of the knowledge of many users (the Wisdom of Crowds [101]). In recent years web communities and social networks had incredible growth rates. Many research efforts have already been made in this field, but there is still potential for further investigations. Especially, how the Wisdom of Crowds can be applied to video segmentation tasks.

During the last years the amount of non-professionally produced content has increased tremendously. Widespread digital cameras and smartphones enable people to capture photos and videos anytime and anywhere. The content is shared using social networks and web communities. Only little research work has been conducted so far to investigate the scene structure of non-professionally produced content, although especially in the news coverage of TV channels and news sites on the Internet journalists very often fall back to amateur content. For example, if no professional team was on-site at an incident or one was there but missed a certain situation of interest. A new field for video scene detection using amateur web videos can be identified. Such videos are rather short, often contain additional handheld camera motion and typically have a low resolution. The new challenge is not detecting scenes in these short videos, but rather how to identify interesting scenes across a number of different videos, from different sources and in different qualities. The aim is to create a richer experience for the viewer by combining the different personal experiences of different people that contributed content.

Regarding the rise of video platforms and content distribution networks, it is not sufficient anymore to investigate only logical segmentation of videos, but also the physical one. An intelligent physical segmentation can guarantee a fast delivery of requested video segments and a reduction of network load, because video segments which are not watched need not be transmitted. The segmentation of videos into chunks is not a new idea in video delivery. Recently presented approaches like HTTP

Live Streaming [70] or ISS Smooth Streaming [122] have also proposed delivery mechanisms that transmit small video units instead of full videos or real streaming. Both solutions segment videos into very small chunks of fixed size. The big difference to the delivery strategies of SOMA [7] is that units of different size are proposed, which contain semantically meaningful information. Furthermore, the idea is to use units from different sources for the composition of new videos, while the other two solutions aim at the transmission of a single video for a streaming-like scenario. *Wang et al.*[112] introduced a P2P Video-on-Demand system that is aware of the shot and scene structure of videos. Clients can request certain shots or scenes from different peers instead of the full video, but they also restrict their consideration to single videos as a source.

Event related research is a relatively young topic. In the field of event summarization still a lot of challenges remain. The temporal alignment of the content is a hard challenge, the timestamps from the camera metadata are not sufficient. Future approaches should incorporate additional sources of information, like textual descriptions of events, for the temporal alignment as well as for the selection of content. Further investigations have to be done what data can be extracted from the context information of content and how trustworthy that information is.

In future, events that last longer than one day (e.g. the whole FIFA World Championship), events that have many parallel sub-events (e.g. Olympic Games), and small events (which only attract the attention of a small audience) must also be investigated. But not only events that are related to entertainment are of interest. The presented approach can also be applied to spontaneous real-life events like a traffic jam on a motorway or a catastrophe scenario like a heavy earthquake. If summaries of such events are constructed from the content that involved people or witnesses have captured, emergency response teams may profit from that information and may be steered and coordinated in a better way.

Last but not least, the presented event summary approaches must be investigated form a user perspective. User tests must be performed to evaluate if users perceive

an added value by using the presented exploration tools and by watching the created event summaries in order to optimize the presented approaches to the needs of the users.

APPENDIX A

Overview of Scene Segmentation Approaches

	Method	Domain	Similarity Matching	Decomposition	Hierarchy
Rui et al. [78]	Time-adaptive grouping	movies	color, activity, temporal constraint	full	key frame(s) – shots – scenes
Hanjalic et al. [32]	Overlapping links	movies	color	full	key frame(s) – shots – scenes
Kwon et al. [46]	Overlapping links	movies	color, motion, shot duration	full	key frame(s) – shots – scenes
Wang et al. [113]	Overlapping links	movies	color, motion	full	key frame(s) – shots – scenes
Mitrovic et al. [61]	Overlapping links	artistic archive documentaries	color, texture, SIFT key points	full	key frame(s) – shots – scenes
Zhao et al. [127]	Temporal sliding window	general	color, temporal constraint	full	key frame(s) – shots – scenes
Cheng et al. [115]	Temporal sliding window	general	color	full	key frame(s) – shots – scenes
Lin et al. [50]	Force competition	sports videos	color, texture	full	key frame(s) – shots – scenes
Rasheed et al. [76]	Backward Shot Coherence	movies, sitcoms	color, motion, shot duration	full	key frame(s) – shots – scenes
Odobez et al. [69]	Spectral Structuring	home videos	color	full	key frame(s) – shots – scenes
Chasanis et al. [10]	Pattern Matching	movies, TV series	color	full	key frame(s) – shots – scenes
Ngo et al. [65]	Motion-Based Scene Change Detection	general	motion, temporal constraints	full	key frame(s) – background images – shots – scenes
del Fabro et al. [17]	Motion Pattern Detection	sports videos	motion	partial	motion sequences – scenes (most frequent motion pattern)

Table A.1: Overview of visual-based scene segmentation

	Method	Domain	Similarity Matching	Decomposition	Hierarchy
Liu et al. [51]	Rally Scene Detection	racquet sports videos	audio	partial	audio segments – scenes (rally scenes)
Friedland et al. [25]	Joke-o-mat	sitcoms	audio	partial	audio segments – scenes (jokes)
Niu et al. [68]	Semantic audio textures	commercials, sitcoms	audio	full	audio segments – scenes

Table A.2: Overview of audio-based scene segmentation

	Method	Domain	Similarity Matching	Decomposition	Hierarchy
Cour et al. [16]	Screenplay and closed captions	movies, TV series	text	full	scenes

Table A.3: Overview of metadata-based scene segmentation

	Method	Domain	Similarity Matching	Decomposition	Hierarchy
Yeung et al. [120]	Scene Transition Graph	general	color, temporal constraints	full	key frame(s) – shots – scenes
Sidiropoulos et al. [90]	Scene Transition Graph	general	color, temporal constraints, audio	full	audio segments – audio scenes / key frame(s) – shots – scenes
Ngo et al. [66], [65]	Scene Transition Graph	general	color, texture, temporal constraints	full	key frame(s) – shots – scenes
Lu et al. [55]	Scene Transition Graph	general	color, number of shots, temporal constraints	full	key frame(s) – shots – scenes
Rasheed et al. [75]	Shot Similarity Graph	movies, sitcoms	color, motion, temporal constraints	full	key frame(s) – shots – scenes
Zhao et al. [128]	Shot Similarity Graph	movies, sitcoms	color, temporal constraints	full	key frame(s) – shots – scenes
Sakarya et al. [79]	Dominant Sets	movies	color, temporal constraints	full	key frame(s) – shots – scenes
Sakarya et al. [80]	Shot Similarity Matrix	movies, sitcoms	color, motion, shot duration, temporal constraints	full	key frame(s) – shots – scenes
Zhang et al. [126]	Shot Similarity Matrix	general	color, temporal constraints	full	key frame(s) – shots – scenes
Weng et al. [114]	Graph of social relationships	movies, sitcoms	face detection	full	key frame(s) – shots – scenes
Sakarya et al. [81]	Scene Transition Graph	movies	color, motion, temporal constraints	full	key frame(s) – shots – scenes

Table A.4: Overview of graph-based scene segmentation

	Method	Domain	Similarity Matching	Decomposition	Hierarchy
Huang et al. [34]	Hidden Markov Models	general	color, audio, motion	full	shots – scenes
Xie et al. [118]	Hidden Markov Models	soccer videos	color, motion	full	scenes (play and break segments)
Yasaroglu et al. [119]	Hidden Markov Models	movies, sitcoms, TV series	face detection, audio, location change, motion	full	shots – scenes
Vinciarelli et al. [110]	Hidden Markov Models	news broadcasts	audio, text/social relationships	full	speaker segments – scenes
Hsu et al. [33]	Maximum Entropy, Boosting, SVM	news broadcasts	color, face detection, audio, motion	full	shots – scenes
Goela et al. [29]	SVM	general	color, audio	full	audio segments – shots – scenes
Zhai et al. [124]	Markov Chain Monte Carlo	movies, home videos	color, shot duration	full	shots – scenes
Gu et al. [30]	Energy Minimization	movies, home videos	shot energy	full	key frame(s) – shots – scenes

Table A.5: Overview of statistics-based scene segmentation

	Method	Domain	Similarity Matching	Decomposition	Hierarchy
Adams et al. [2]	Tempo	movies	motion, shot duration	full	shots – scenes
Cheng et al. [15]	Tempo	movies	motion, shot duration	full	shots – scene units – scenes
Aner et al. [3]	Predefined Plots	sitcoms, sports videos	color	partial	key frame(s) – background images – shots – scenes (locations, events defined by plots)
Chen et al. [12]	Rule-Based	movies	color, shot duration	partial	key frame(s) – shots – scenes (action or dialog)
Zhou et al. [129] *Tavanapong et al.* [102]	Shot-Weave, Rule-Based	movies	color, shot activity	full	key frame(s) – shots – scenes
Truong et al. [104]	Neighborhood Coherence, Tempo	movies	color, temporal constraint, motion, shot duration	full	key frame(s) – shots – scenes
Geng et al. [26]	Audio-Motion correlation	movies	motion, audio	partial	shots – scenes (dialog or action)
Chen et al. [13]	Neighborhood Coherence	general	color, texture	full	key frame(s) – background images – shots – scenes

Table A.6: Overview of film-editing-rules-based scene segmentation

	Method	Domain	Similarity Matching	Decomposition	Hierarchy
Huang et al. [35]	Simultaneous image, audio and motion change	recorded TV broadcasts	color, audio, motion	full	audio segments – audio scenes / key frame(s) – shots – scenes
Lienhart et al. [49]	Audio-Visual Detection	movies	color, texture, audio, face detection	full	audio segments – audio scenes / key frame(s) – shots – scenes
Sundaram et al. [99]	Audio-Visual Detection	movies	color, audio	full	audio segments – audio scenes / key frame(s) – shots – scenes
Chen et al. [12]	Audio-Visual Detection	general	color, texture, audio	full	audio segments – audio scenes / key frame(s) – shots – scenes
Nitanda et al. [67]	Audio-Visual Detection	general	color, audio	full	audio segments – audio scenes / key frame(s) – shots – scenes
Javed et al. [39]	Graph-, Statistics- and Film-Rule-Based	talk and game shows	color, temporal constraints, shot duration	full	key frame(s) – shots – scenes
Chaisorn et al. [8]	Audio, Visual, Statistics and Meta-Data	news broadcasts	color, motion, audio, text	full	key frame(s) – shots – scenes
Zhai et al. [125]	Graph, Audio, Visual	news broadcasts	color, audio, text	full	key frame(s) – shots – scenes
Ariki et al. [5]	Audio-Visual Detection	baseball videos	color, audio, text	partial	key frame(s) – shots – scenes (pitcher scenes in baseball videos)
Arifin et al. [4]	Pleasure-Arousal-Dominance Model	movies	color, audio, temporal constraints	full	key frame(s) – shots – scenes
Zhu et al. [130]	TV broadcast segmentation	recorded TV broadcasts	color, texture, temporal constraints	full	key frame(s) – shots – scenes
Liang et al. [48] *Sang et al.* [82]"	Role-Based Movie Segmentation	movies	face detection, text	full	scenes
Janin et al. [37]	Joke-o-mat HD	sitcoms	audio, text	partial	audio segments – scenes (joke and dialogs)
Ellouze et al. [24]	Kohonen maps + Tempo approach merged	movies	color, texture, temporal constraints, motion, audio, shot frequency	full	key frame(s) – shots – scenes
Wang et al. [111]	Broadcast TV segmentation	recorded TV broadcasts	color, audio, text	full	key frame(s) – shots – scenes – TV program

Table A.7: Overview of hybrid scene segmentation

Bibliography

[1] *MM '09: Proceedings of the 17th ACM international conference on Multimedia*, New York, NY, USA, 2009. ACM. 433097.

[2] B. Adams, C. Dorai, and S. Venkatesh. Toward automatic extraction of expressive elements from motion pictures: tempo. *IEEE Transactions on Multimedia*, 4(4):472–481, December 2002.

[3] Aya Aner and John Kender. Video Summaries through Mosaic-Based Shot and Scene Clustering. In Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, editors, *Computer Vision ECCV 2002*, volume 2353 of *Lecture Notes in Computer Science*, chapter 26, pages 45–49. Springer Berlin / Heidelberg, Berlin, Heidelberg, April 2006.

[4] Sutjipto Arifin and Peter Y. K. Cheung. Affective level video segmentation by utilizing the Pleasure-Arousal-dominance information. *IEEE Transactions on Multimedia*, 10(7):1325–1341, November 2008.

[5] Yasuo Ariki, Masahito Kumano, and Kiyoshi Tsukada. Highlight scene extraction in real time from baseball live video. In *Proceedings of the 5th ACM*

SIGMM international workshop on Multimedia information retrieval, MIR '03, pages 209–214, New York, NY, USA, 2003. ACM.

[6] M. Barbieri, G. Mekenkamp, M. Ceccarelli, and J. Nesvadba. The color browser: a content driven linear video browsing tool. *Multimedia and Expo, 2001. ICME 2001. IEEE International Conference on*, pages 627–630, 2001.

[7] Laszlo Böszörmenyi, Manfred del Fabro, Marian Kogler, Mathias Lux, Oge Marques, and Anita Sobe. Innovative directions in self-organized distributed multimedia systems. *Multimedia Tools and Applications*, 51:525–553, 2011.

[8] L. Chaisorn, Tat-Seng Chua, and Chin-Hui Lee. The segmentation of news video into story units. In *Multimedia and Expo, 2002. ICME '02. 2002 IEEE International Conference on*, volume 1, pages 73–76, 2002.

[9] Linjun Chang, Yichen Yang, and Xian-Sheng Hua. Smart video player. In *Multimedia and Expo, 2008 IEEE International Conference on*, pages 1605–1606, 23 2008-April 26 2008.

[10] Vasileios T. Chasanis, Aristidis C. Likas, and Nikolaos P. Galatsanos. Scene Detection in Videos Using Shot Clustering and Sequence Alignment. *IEEE Transactions on Multimedia*, 11(1):89–100, January 2009.

[11] Savvas Chatzichristofis and Yiannis Boutalis. CEDD: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval. In Antonios Gasteratos, Markus Vincze, and John Tsotsos, editors, *Computer Vision Systems*, volume 5008 of *Lecture Notes in Computer Science*, pages 312–322. Springer Berlin / Heidelberg, 2008.

[12] Lei Chen and M.T. Ozsu. Rule-based scene extraction from video. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, 2002.

[13] Liang-Hua Chen, Yu-Chun Lai, and Hong-Yuan Mark Liao. Movie scene segmentation using background information. *Pattern Recognition*, 41:1056–1065, March 2008.

[14] Kai-Yin Cheng, Sheng-Jie Luo, Bing-Yu Chen, and Hao-Hua Chu. Smartplayer: user-centric video fast-forwarding. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, pages 789–798, New York, NY, USA, 2009. ACM.

[15] Wengang Cheng and Jun Lu. Video scene oversegmentation reduction by tempo analysis. In *Natural Computation, 2008. ICNC '08. Fourth International Conference on*, volume 4, pages 296 –300, 2008.

[16] Timothee Cour, Chris Jordan, Eleni Miltsakaki, and Ben Taskar. Movie/Script: Alignment and Parsing of Video and Text Transcription. In David Forsyth, Philip Torr, and Andrew Zisserman, editors, *Computer Vision ECCV 2008*, volume 5305 of *Lecture Notes in Computer Science*, chapter 12, pages 158–171. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2008.

[17] M. del Fabro and L. Böszörmenyi. Video scene detection based on recurring motion patterns. In *Advances in Multimedia (MMEDIA), 2010 Second International Conferences on*, pages 113 –118, 2010.

[18] Manfred del Fabro and Laszlo Böszöermenyi. Summarization and presentation of real-life events using community-contributed content. In *Proceedings of the 18th International Conference on Multimedia Modeling*, Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2012.

[19] Manfred del Fabro and Laszlo Böszörmenyi. The vision of crowds: Social event summarization based on user- generated multimedia content. In Christine Robson, Sean Kandel, Jeff Heer, and Jeff Pierce, editors, *ACM CHI 2011 Workshop Data Collection By The People For The People*, pages 1–5,

http://databythepeople.com/ (May 2011), may 2011. published on workshop homepage.

[20] Manfred del Fabro, Klaus Schoeffmann, and Laszlo Böszörmenyi. Instant video browsing: A tool for fast non-sequential hierarchical video browsing. In Gerhard Leitner, Martin Hitz, and Andreas Holzinger, editors, *HCI in Work and Learning, Life and Leisure*, volume 6389 of *Lecture Notes in Computer Science*, pages 443–446. Springer Berlin / Heidelberg, 2010.

[21] Anhai Doan, Raghu Ramakrishnan, and Alon Y. Halevy. Crowdsourcing systems on the world-wide web. *Commun. ACM*, 54:86–96, April 2011.

[22] Ina Döhring and Rainer Lienhart. Mining tv broadcasts for recurring video sequences. In *CIVR '09: Proceeding of the ACM International Conference on Image and Video Retrieval*, pages 1–8, New York, NY, USA, 2009. ACM.

[23] Pierre Dragicevic, Gonzalo Ramos, Jacobo Bibliowitcz, Derek Nowrouzezahrai, Ravin Balakrishnan, and Karan Singh. Video browsing by direct manipulation. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 237–246, New York, NY, USA, 2008. ACM.

[24] Mehdi Ellouze, Nozha Boujemaa, and Adel Alimi. Scene pathfinder: unsupervised clustering techniques for movie scenes extraction. *Multimedia Tools and Applications*, 47(2):325–346, April 2010.

[25] Gerald Friedland, Luke Gottlieb, and Adam Janin. Joke-o-mat: browsing sitcoms punchline by punchline. In *Proceedings of the seventeen ACM international conference on Multimedia*, MM '09, pages 1115–1116, New York, NY, USA, 2009. ACM.

[26] Yuliang Geng, De Xu, and Aimin Wu. Effective Video Scene Detection Approach Based on Cinematic Rules. In Rajiv Khosla, Robert J. Howlett, and

Lakhmi C. Jain, editors, *Knowledge-Based Intelligent Information and Engineering Systems*, volume 3682 of *Lecture Notes in Computer Science*, chapter 165, pages 1197–1203. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2005.

[27] Andreas Girgensohn, Frank Shipman, and Lynn Wilcox. Adaptive clustering and interactive visualizations to support the selection of video clips. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, pages 34:1–34:8, New York, NY, USA, 2011. ACM.

[28] Nikolaos Gkalelis, Vasileios Mezaris, and Ioannis Kompatsiaris. Automatic event-based indexing of multimedia content using a joint content-event model. In *Proceedings of the 2nd ACM international workshop on Events in multimedia*, EiMM '10, pages 15–20, New York, NY, USA, 2010. ACM.

[29] Naveen Goela, Kevin Wilson, Feng Niu, Ajay Divakaran, and Isao Otsuka. An SVM framework for Genre-Independent scene change detection. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 532–535, July 2007.

[30] Zhiwei Gu, Tao Mei, Xian-Sheng Hua, Xiuqing Wu, and Shipeng Li. EMS: Energy Minimization Based Video Scene Segmentation. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 520–523, July 2007.

[31] M. Guillemot, P. Wellner, D. Gatica-Perez, and J-M. Odobez. A hierarchical keyframe user interface for browsing video over the internet. In *Proceedings of the 9th IFIP TC13 International Conference on Human-Computer Interaction (INTERACT 2003)*, 2003.

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

[33] W. H. M. Hsu and Shih-Fu Chang. Generative, discriminative, and ensemble learning on multi-modal perceptual fusion toward news video story segmentation. In *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, volume 2, pages 1091–1094, 2004.

[34] Jincheng Huang, Zhu Liu, and Yao Wang. Joint scene classification and segmentation based on hidden markov model. *IEEE Transactions on Multimedia*, 7(3):538–550, June 2005.

[35] Jincheng Huang, Zhu Liu, and Wang Yao. Integration of audio and visual information for content-based video segmentation. In *Image Processing, 1998. ICIP 98. 1998 International Conference on*, pages 526 –529 vol.3, October 1998.

[36] W. Hürst. Interactive audio-visual video browsing. In *Proceedings of the 14th annual ACM international conference on Multimedia*, pages 675–678. ACM New York, NY, USA, 2006.

[37] Adam Janin, Luke Gottlieb, and Gerald Friedland. Joke-o-Mat HD: browsing sitcoms with human derived transcripts. In *Proceedings of the international conference on Multimedia*, MM '10, pages 1591–1594, New York, NY, USA, 2010. ACM.

[38] M. Jansen, W. Heeren, and B. van Dijk. Videotrees: Improving video surrogate presentation using hierarchy. In *Content-Based Multimedia Indexing, 2008. CBMI 2008. International Workshop on*, pages 560 –567, june 2008.

[39] O. Javed, Z. Rasheed, and M. Shah. A framework for segmentation of talk and game shows. In *Computer Vision, 2001. ICCV 2001. Eighth IEEE International Conference on*, 2001.

[40] Rene Kaiser, Michael Hausenblas, and Martin Umgeher. Metadata-driven interactive web video assembly. *Multimedia Tools and Applications*, 41:437–467, 2009.

[41] E. Katz, F.M. Klein, and R.D. Nolen. *The film encyclopedia*. Film Encyclopedia. HarperPerennial, 1998.

[42] Marian Kogler, Manfred del Fabro, Mathias Lux, Klaus Schoeffmann, and Laszlo Böszörmenyi. Global vs. local feature in video summarization: Experimental results. In *Proceedings of the 10th International Workshop of the Multimedia Metadata Community on Semantic Multimedia Database Technologies (SeMuDaTe09) in conjunction with the 4th International Conference on Semantic and Digital Media Technologies (SAMT 2009).*

[43] Marian Kogler and Mathias Lux. Pursuing the holy grail by interrelating user intentions and bag of visual words to perform retrieval adaptation. In *Proceedings of the ACM International Conference on Multimedia*, Scottsdale, AZ, USA, December 2011.

[44] Marian Kogler, Mathias Lux, and Oge Marques. Adaptive visual information retrieval by changing visual vocabulary sizes in context of user intentions. In *Proceedings of MMWeb2011, IEEE*, 2011.

[45] T. Kohonen. The self-organizing map. *Neurocomputing*, 21(1-3):1–6, November 1998.

[46] Yong-Moo Kwon, Chang-Jun Song, and Ig-Jae Kim. A new approach for high level video structuring. In *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, 2000.

[47] Robert Laganière, Raphael Bacco, Arnaud Hocevar, Patrick Lambert, Grégory Païs, and Bogdan E. Ionescu. Video summarization from spatio-temporal features. In *TVS '08: Proceedings of the 2nd ACM TRECVid Video Summarization Workshop*, pages 144–148, New York, NY, USA, 2008. ACM.

[48] Chao Liang, Yifan Zhang, Jian Cheng, Changsheng Xu, and Hanqing Lu. A Novel Role-Based Movie Scene Segmentation Method. In Paisarn Muneesawang,

Feng Wu, Itsuo Kumazawa, Athikom Roeksabutr, Mark Liao, and Xiaoou Tang, editors, *Advances in Multimedia Information Processing - PCM 2009*, volume 5879 of *Lecture Notes in Computer Science*, chapter 82, pages 917–922. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2009.

[49] R. Lienbart, S. Pfeiffer, and W. Effelsberg. Scene determination based on video and audio features. In *Multimedia Computing and Systems, 1999. IEEE International Conference on*, volume 1, pages 685–690, 1999.

[50] Tong Lin, Hong-Jiang Zhang, and Qing-Yun Shi. Video scene extraction by force competition. *Multimedia and Expo, IEEE International Conference on*, 0:192, 2001.

[51] Chunxi Liu, Qingming Huang, Shuqiang Jiang, Liyuan Xing, Qixiang Ye, and Wen Gao. A framework for flexible summarization of racquet sports video using multiple modalities. *Computer Vision and Image Understanding*, 113(3):415–424, March 2009.

[52] Xueliang Liu, Raphaël Troncy, and Benoit Huet. Finding media illustrating events. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, pages 58:1–58:8, New York, NY, USA, 2011. ACM.

[53] Lie Lu, Rui Cai, and A. Hanjalic. Audio elements based auditory scene segmentation. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 5, page V, May 2006.

[54] Lie Lu, Hong-Jiang Zhang, and Hao Jiang. Content analysis for audio classification and segmentation. *IEEE Transactions on Speech and Audio Processing*, 10(7):504–516, October 2002.

[55] Shi Lu, I. King, and M.R. Lyu. A novel video summarization framework for document preparation and archival applications. In *Aerospace Conference, 2005 IEEE*, pages 1 –10, 2005.

[56] Mathias Lux, Christoph Kofler, and Oge Marques. A classification scheme for user intentions in image search. In *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*, CHI EA '10, pages 3913–3918, New York, NY, USA, 2010. ACM.

[57] Mathias Lux, Marian Kogler, and Manfred del Fabro. Why did you take this photo: a study on user intentions in digital photo productions. In *Proceedings of the 2010 ACM workshop on Social, adaptive and personalized multimedia interaction and access*, SAPMIA '10, pages 41–44, New York, NY, USA, 2010. ACM.

[58] Mathias Lux, Oge Marques, Klaus Schöffmann, Laszlo Böszörmenyi, and Georg Lajtai. A novel tool for summarization of arthroscopic videos. *Multimedia Tools and Applications*, 46:521–544, 2010.

[59] Jarmo Makkonen, Riitta Kerminen, Igor D. D. Curcio, Sujeet Mate, and Ari Visa. Detecting events by clustering videos from large media databases. In *Proceedings of the 2nd ACM international workshop on Events in multimedia*, EiMM '10, pages 9–14, New York, NY, USA, 2010. ACM.

[60] Knut Manske. Video browsing using 3d video content trees. In *Proceedings of the 1998 workshop on New paradigms in information visualization and manipulation*, NPIV '98, pages 20–24, New York, NY, USA, 1998. ACM.

[61] Dalibor Mitrović, Stefan Hartlieb, Matthias Zeppelzauer, and Maia Zaharieva. Scene Segmentation in Artistic Archive Documentaries. In Gerhard Leitner, Martin Hitz, and Andreas Holzinger, editors, *HCI in Work and Learning, Life and Leisure*, volume 6389 of *Lecture Notes in Computer Science*, chapter 27, pages 400–410. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2010.

[62] James Monaco. *How to Read a Film: The World of Movies, Media, Multimedia: Language, History, Theory.* Oxford University Press, USA, 3 edition, January 2000.

[63] Arthur G. Money and Harry Agius. Video summarisation: A conceptual framework and survey of the state of the art. *J. Vis. Comun. Image Represent.*, 19(2):121–143, February 2008.

[64] Saul B. Needleman and Christian D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443 – 453, 1970.

[65] Chong-Wah Ngo, Yu-Fei Ma, and Hong-Jiang Zhang. Video summarization and scene detection by graph modeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(2):296–305, February 2005.

[66] Chong-Wah Ngo, Yu-Fei Ma, and Hong-Jiang Zhang. Video summarization and scene detection by graph modeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(2):296–305, February 2005.

[67] N. Nitanda, M. Haseyama, and H. Kitajima. Audio signal segmentation and classification for scene-cut detection. In *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pages 4030 – 4033 Vol. 4, May 2005.

[68] F. Niu, N. Goela, A. Divakaran, and M. Abdel-Mottaleb. Audio scene segmentation for video with generic content. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 6820 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, January 2008.

[69] Jean-Marc Odobez, Daniel Gatica-Perez, and Mael Guillemot. Spectral Structuring of Home Videos. In Erwin Bakker, Michael Lew, Thomas Huang, Nicu Sebe, and Xiang Zhou, editors, *Image and Video Retrieval*, volume 2728 of *Lecture Notes in Computer Science*, chapter 31, pages 85–90. Springer Berlin / Heidelberg, Berlin, Heidelberg, June 2003.

[70] R. Pantos and May W. HTTP live streaming (draft). `http://tools.ietf.org/html/draft-pantos-http-live-streaming-07`, September 2010. [Online].

[71] S. Papadopoulos, C. Zigkolis, Y. Kompatsiaris, and A. Vakali. Cluster-based landmark and event detection for tagged photo collections. *Multimedia, IEEE*, 18(1):52 –63, jan. 2011.

[72] F. Pletzer and B. Rinner. Distributed task allocation for visual sensor networks: A market-based approach. In *Self-Adaptive and Self-Organizing Systems Workshop (SASOW), 2010 Fourth IEEE International Conference on*, pages 59–62, sept. 2010.

[73] F. Pletzer, R. Tusch, L. Böszörmenyi, B. Rinner, O. Sidla, M. Harrer, and T. Mariacher. Feature-based level of service classification for traffic surveillance. In *Intelligent Transportation Systems (ITSC), 2011 14th IEEE International Conference on*, pages 1015–1020, oct. 2011.

[74] F. Pletzer, R. Tusch, B. Rinner, Böszörmenyi L., M. Harrer, and T. Mariacher. Avss 2011 demo session: Level of service classification for smart cameras. In *Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on*, pages 543–544, sept. 2011.

[75] Z. Rasheed and M. Shah. Detection and representation of scenes in videos. *IEEE Transactions on Multimedia*, 7(6):1097–1105, December 2005.

[76] Zeeshan Rasheed and Mubarak Shah. Scene detection in hollywood movies and tv shows. volume 2, page 343, Los Alamitos, CA, USA, 2003. IEEE Computer Society.

[77] G. J. Rath, A. Resnick, and T. R. Savage. The formation of abstracts by the selection of sentences. part i. sentence selection by men and machines. *American Documentation*, 12(2):139–141, 1961.

[78] Yong Rui, Thomas S. Huang, and Sharad Mehrotra. Constructing table-of-content for videos. *Multimedia Systems*, 7(5):359–368, September 1999.

[79] U. Sakarya and Z. Telatar. Video scene detection using dominant sets. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 73 –76, 2008.

[80] Ufuk Sakarya and Ziya Telatar. Graph-based multilevel temporal video segmentation. *Multimedia Systems*, 14(5):277–290, November 2008.

[81] Ufuk Sakarya and Ziya Telatar. Video scene detection using graph-based representations. *Signal Processing: Image Communication*, 25(10):774–783, November 2010.

[82] Jitao Sang and Changsheng Xu. Character-based movie summarization. In *Proceedings of the international conference on Multimedia*, MM '10, pages 855–858, New York, NY, USA, 2010. ACM.

[83] K. Schoeffmann, M. Taschwer, and L. Böszörmenyi. Video browsing using motion visualization. In *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*, pages 1835–1836, August 2009.

[84] Klaus Schoeffmann and Laszlo Böszörmenyi. *Advances in Semantic Media Adaptation and Personalization*, volume 2, chapter Interactive Video Browsing of H.264 Content Based on Just-in-Time Analysis, pages 159–180. CRC Press, February 2009.

[85] Klaus Schoeffmann and Manfred del Fabro. Hierarchical video browsing with a 3d carousel. In *Proceedings of the ACM International Conference on Multimedia*, Scottsdale, AZ, USA, December 2011.

[86] Klaus Schoeffmann, Frank Hopfgartner, Oge Marques, Laszlo Böszörmenyi, and Joemon M. Jose. Video browsing interfaces and applications: a review. *SPIE Reviews*, 1(1), 2010.

[87] Klaus Schoeffmann, Mathias Lux, Mario Taschwer, and Laszlo Böszörmenyi. Visualization of video motion in context of video browsing. In *Proceedings of*

the IEEE International Conference on Multimedia and Expo, New York, USA, July 2009. IEEE.

[88] Klaus Schoeffmann, Mario Taschwer, and Laszlo Böszörmenyi. The video explorer: a tool for navigation and searching within a single video based on fast content analysis. In *MMSys 10: Proceedings of the first annual ACM SIGMM conference on Multimedia systems*, page 247258, New York, NY, USA, 2010. ACM.

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

[90] Panagiotis Sidiropoulos, Vasileios Mezaris, Ioannis Kompatsiaris, Hugo Meinedo, and Isabel Trancoso. Multi-modal scene segmentation using scene transition graphs. In *Proceedings of the seventeen ACM international conference on Multimedia*, MM '09, pages 665–668, New York, NY, USA, 2009. ACM.

[91] I. Simon, N. Snavely, and S. M. Seitz. Scene summarization for online image collections. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, October 2007.

[92] Pinaki Sinha, Sharad Mehrotra, and Ramesh Jain. Summarization of personal photologs using multidimensional content and context. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, pages 4:1–4:8, New York, NY, USA, 2011. ACM.

[93] Alan F. Smeaton, Paul Over, and Wessel 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.

[94] Anita Sobe, Laszlo Böszörmenyi, and Mario Taschwer. Video Notation (ViNo): A Formalism for Describing and Evaluating Non-sequential Multimedia Access. *International Journal on Advances in Software*, 3(1 & 2):19–30, sep 2010.

[95] Anita Sobe, Wilfried Elmenreich, and Laszlo Böszörmenyi. Towards a self-organizing replication model for non-sequential media access. In Alberto Del Bimbo, Shih-Fu Chang, and Arnold Smeulders, editors, *Proceedings of the 18th International Conference on Multimedea 2010*, pages 3–8, New York, jan 2010. ACM.

[96] Anita Sobe, Wilfried Elmenreich, and Laszlo Böszörmenyi. Replication for bio-inspired delivery in unstructured peer-to-peer networks. In Markus Kucera and Thomas Waas, editors, *Proceedings of the Ninth Workshop on intelligent solutions for embedded systems*, page 6pp., Los Alamitos, CA, USA, jul 2011. IEEE.

[97] Anita Sobe, Wilfried Elmenreich, and Laszlo Böszörmenyi. Storage balancing in self-organizing multimedia delivery systems. Technical Report TR/ITEC/01/2.13, Institute of Information Technology (ITEC), Klagenfurt University, Klagenfurt, Austria, oct 2011.

[98] M. Strohmaier, M. Lux, M. Granitzer, P. Scheir, S. Liaskos, and E. Yu. How do users express goals on the web? - an exploration of intentional structures in web search. In *WISE*, pages 67–78, 2007.

[99] H. Sundaram and Shih-Fu Chang. Video scene segmentation using video and audio features. In *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, 2000.

[100] H. Sundaram and Shih-Fu Chang. Computable scenes and structures in films. *IEEE Transactions on Multimedia*, 4(4):482–491, December 2002.

[101] James Surowiecki. *The Wisdom of Crowds*. Anchor, 2005.

[102] W. Tavanapong and J. Zhou. Shot Clustering Techniques for Story Browsing. *IEEE Transactions on Multimedia*, 6(4):517–527, August 2004.

[103] Mohamed Riadh Trad, Alexis Joly, and Nozha Boujemaa. Large scale visual-based event matching. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, pages 53:1–53:7, New York, NY, USA, 2011. ACM.

[104] Ba T. 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.

[105] Ba T. Truong and Svetha Venkatesh. Video abstraction: A systematic review and classification. *ACM Trans. Multimedia Comput. Commun. Appl.*, 3(1):3+, 2007.

[106] Roland Tusch, Armin Fuchs, Horst Gutmann, Marian Kogler, Julius Köpke, Laszlo Böszörmenyi, Manfred Harrer, and Thomas Mariacher. A multimedia-centric quality assurance system for traffic messages. In Julia Düh, Hartwig Hufnagl, Erhard Juritsch, Reinhard Pfliegl, Helmut-Klaus Schimany, and Hans Schönegger, editors, *Data and Mobility*, volume 81 of *Advances in Intelligent and Soft Computing*, pages 1–13. Springer Berlin / Heidelberg, 2010.

[107] Roelof van Zwol, Börkur Sigurbjornsson, Ramu Adapala, Lluis Garcia Pueyo, Abhinav Katiyar, Kaushal Kurapati, Mridul Muralidharan, Sudar Muthu, Vanessa Murdock, Polly Ng, Anand Ramani, Anuj Sahai, Sriram Thiru Sathish, Hari Vasudev, and Upendra Vuyyuru. Faceted exploration of image search results. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 961–970, New York, NY, USA, 2010. ACM.

[108] J. Vendrig and M. Worring. Systematic evaluation of logical story unit segmentation. *IEEE Transactions on Multimedia*, 4(4):492–499, December 2002.

[109] Sami Vihavainen, Sujeet Mate, Lassi Seppälä, Francesco Cricri, and Igor D.D. Curcio. We want more: human-computer collaboration in mobile social video remixing of music concerts. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, CHI '11, pages 287–296, New York, NY, USA, 2011. ACM.

[110] Alessandro Vinciarelli and Sarah Favre. Broadcast news story segmentation using social network analysis and hidden markov models. In *Proceedings of the 15th international conference on Multimedia*, MULTIMEDIA '07, pages 261–264, New York, NY, USA, 2007. ACM.

[111] Jinqiao Wang, Lingyu Duan, Qingshan Liu, Hanqing Lu, and Jesse S. Jin. A multimodal scheme for program segmentation and representation in broadcast video streams. *IEEE Transactions on Multimedia*, 10(3):393–408, April 2008.

[112] Xin Wang, Changyi Zheng, Zhenyuan Zhang, Hong Lu, and Xiangyang Xue. The design of video segmentation-aided VCR support for P2P VoD systems. *IEEE Transactions on Consumer Electronics*, 54(2):531–537, May 2008.

[113] Xuejun Wang, Shigang Wang, Shigang Xuejun, and M. Gabbouj. A shot clustering based algorithm for scene segmentation. In *Computational Intelligence and Security Workshops, 2007. CISW 2007. International Conference on*, pages 259 –252, 2007.

[114] Chung-Yi Weng, Wei-Ta Chu, and Ja-Ling Wu. RoleNet: Movie analysis from the perspective of social networks. *IEEE Transactions on Multimedia*, 11(2):256–271, February 2009.

[115] Cheng Wengang and Xu De. A novel approach of generating video scene structure. In *TENCON 2003. Conference on Convergent Technologies for Asia-Pacific Region*, volume 1, pages 350 – 353 Vol.1, 2003.

[116] Utz Westermann and Ramesh Jain. Toward a common event model for multimedia applications. *IEEE Multimedia*, 14(1):19–29, 2007.

[117] Kent Wittenburg, Clifton Forlines, Tom Lanning, Alan Esenther, Shigeo Harada, and Taizo Miyachi. Rapid serial visual presentation techniques for consumer digital video devices. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, UIST '03, pages 115–124, New York, NY, USA, 2003. ACM.

[118] L. Xie. Structure analysis of soccer video with domain knowledge and hidden markov models. *Pattern Recognition Letters*, 25(7):767–775, May 2004.

[119] Yağiz Yaşaroğlu and A. Alatan. Summarizing video: Content, features, and HMM topologies. In Narciso García, Luis Salgado, and José M. Martínez, editors, *Visual Content Processing and Representation*, volume 2849 of *Lecture Notes in Computer Science*, chapter 15, pages 101–110. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2003.

[120] M. Yeung. Segmentation of Video by Clustering and Graph Analysis. *Computer Vision and Image Understanding*, 71(1):94–109, July 1998.

[121] Jinhui Yuan, Huiyi Wang, Lan Xiao, Wujie Zheng, Jianmin Li, Fuzong Lin, and Bo Zhang. A formal study of shot boundary detection. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(2):168 –186, feb. 2007.

[122] A. Zambelli. ISS smooth streaming (technical overview). `http://download.microsoft.com/download/4/2/4/4247C3AA-7105-4764-A8F9-321CB6C765EB/IIS_Smooth_Streaming_Technical_Overview.pdf`, March 2009. [Online].

[123] Oren Zamir and Oren Etzioni. Web document clustering: a feasibility demonstration. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 46–54, New York, NY, USA, 1998. ACM.

[124] Yun Zhai and M. Shah. Video scene segmentation using markov chain monte carlo. *IEEE Transactions on Multimedia*, 8(4):686–697, August 2006.

[125] Yun Zhai, Alper Yilmaz, and Mubarak Shah. Story segmentation in news videos using visual and text cues. In Wee-Kheng Leow, Michael Lew, Tat-Seng Chua, Wei-Ying Ma, Lekha Chaisorn, and Erwin Bakker, editors, *Image and Video Retrieval*, volume 3568 of *Lecture Notes in Computer Science*, chapter 13, pages 92–102. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2005.

[126] Zhenyuan Zhang, Bin Li, Hong Lu, and Xiangyang Xue. Scene segmentation based on video structure and spectral methods. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, pages 1093 –1096, 2008.

[127] Li Zhao, Shi-Qiang Yang, and Bo Feng. Video scene detection using slide windows method based on temporal constrain shot similarity. In *Multimedia and Expo, 2001. ICME 2001. IEEE International Conference on*, pages 1171 – 1174, 2001.

[128] Yanjun Zhao, Tao Wang, Peng Wang, Wei Hu, Yangzhou Du, Yimin Zhang, and Guangyou Xu. Scene segmentation and categorization using ncuts. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1 –7, 2007.

[129] Junyu Zhou and Wallapak Tavanapong. Shot Weave: A Shot Clustering Technique for Story Browsing for Large Video Databases. In Akmal Chaudhri, Rainer Unland, Chabane Djeraba, and Wolfgang Lindner, editors, *XML-Based Data Management and Multimedia Engineering EDBT 2002 Workshops*, volume 2490 of *Lecture Notes in Computer Science*, chapter 17, pages 529–533. Springer Berlin / Heidelberg, Berlin, Heidelberg, November 2002.

[130] Songhao Zhu and Yuncai Liu. Video scene segmentation and semantic representation using a novel scheme. *Multimedia Tools and Applications*, 42(2):183–205, April 2009.