Automatic summarization of narrative video

Citation for published version (APA):
Barbieri, M. (2007). *Automatic summarization of narrative video*. [Phd Thesis 2 (Research NOT TU/e / Graduation TU/e), Frits Philips Inst. Quality Management]. Technische Universiteit Eindhoven. https://doi.org/10.6100/IR632360

DOI:
10.6100/IR632360

Document status and date:
Published: 01/01/2007

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Automatic Summarization of Narrative Video

ISBN: 978-90-74445-82-5

Cover design by Bertina Senders, B-Design Grafische Vormgeving

The work described in this thesis has been carried out at the Philips Research Laboratories in Eindhoven, The Netherlands, as part of the Philips Research programme.

Automatic Summarization of Narrative Video

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van
de Rector Magnificus, prof.dr.ir. C.J. van Duijn,
voor een commissie aangewezen door het College
voor Promoties in het openbaar te verdedigen op
donderdag 29 november 2007 om 16.00 uur

door

Mauro Barbieri

geboren te Cento, Italië

Dit proefschrift is goedgekeurd door de promotor:

prof.dr. E.H.L. Aarts

Copromotor:
dr.ir. J.H.M. Korst

Contents

1

Introduction

Advances in digital processing and storage technologies are contributing to an enormous and steadily growing availability of video content. Despite the enormous investments in digital video technologies, the capabilities of an average user to manipulate, interact with and manage video (collections) are still far behind what average users can achieve with other types of media such as text or images. This is mainly due to the temporal and multi-modal nature of video and the size of the associated medium.

Hard-disk drives and digital video compression technology have created the possibility of time-shifting live television and recording a large number of TV shows in high quality without having to worry about availability of tapes or other removable storage media. A hard-disk video recorder with a storage capacity of 1 Terabyte can store up to 20 days of non-stop standard definition video in DVD quality. If we consider the current growth factor of 1.5 in the capacity of hard-disk drives per year [Chip, 2005], in the next decade, a single hard-disk will be sufficient for storing more than one year of non-stop video.

At the same time, digital television standards such as DVB [DVB, 2007] have multiplied the number of content sources for the average user. Hundreds of channels are available using a simple parabolic antenna and a digital receiver. Thousands of potentially interesting programmes are broadcast every day and can be recorded and stored locally for later access.

Digital television is not the only source of video content. Internet has evolved in the last decades from a mainly text-based collection of hyper-linked documents to a cornucopia of multimedia content of any type and genre. ADSL technology has provided the required breakthrough in bandwidth that was necessary to bring high-quality video streaming, web television and video-on-demand to the home. Digital video cameras in the form of camcorders, web-cams, or video-phones have become pervasive and the amount of user-generated video published on the Internet is predicted to surpass professionally produced content. New technologies such as peer-to-peer television and peer-to-peer content sharing have emerged although still illegal or not completely free from copyright infringements. As a result of these technological advances, average users are overwhelmed with enormous amounts of content.

To locate items of interest in this ocean of multimedia content, users have adopted services such as electronic program guides, TV web-portals, and web search engines that aggregate information relevant to the users' queries and allow them to find easily the content they are looking for. Additionally, databases containing meta-data associated with commercial and professional multimedia productions (e.g. music and film) have been made available to the public and have been turned into successful businesses (e.g. Internet Movie Database [IMDB, 2007] and CD Database [CDDB, 2007]).

However, while content offer and availability for the average users have increased enormously, free time for consuming content has not increased much. The key-problem of each consumer is to make efficient use of the free-time available for enjoying content. At the same time, content providers and distributors fight to attain and hold customers' attention.

The main issues are filtering-out uninteresting information and choosing the right content for one's needs (e.g. infotainment, education). New technology is needed to get insights into multimedia content in order to help users making choices, as simply as possible, while being entertained. *Recommender* technology [Smyth and Cotter, 2000] addresses these problems by estimating the degree of likeness of a certain programme for a certain user and automatically ranking content items. This is done by comparing a user's profile with reference characteristics or with profiles of other users with similar tastes.

Recommenders can be seen as tools for filtering out unwanted content and bringing interesting content to the attention of the user. However, even after filtering, the amount of content that might match a user's preferences could still be too large. Furthermore, the choice of what to watch does not only depend on general and static preferences, but it depends on the actual situation. In other words, automatic recommenders should not only model the general preferences of certain users, but they should be capable of understanding the needs of each user every time a choice has to be made. This last requirement is far beyond the capabilities of current recommender technology and one might wonder whether this goal will ever by achieved. Additionally we should not forget that users need to feel a certain degree of control, especially in making choices and expressing their preferences. Besides recommenders, there is a need for new tools to facilitate content selection from the large variety of available options.

This thesis addresses the problem of helping users selecting a video content item from a large set.

Automatic video summarization aims at creating efficient representations of video for facilitating browsing, search and, more generically, management of digital multimedia content. Automatically generated summaries can support users in navigating large video archives and in taking decisions more efficiently regarding selecting, consuming, sharing, or deleting content.

This thesis presents an approach to the automatic summarization of narrative video that is based on the application of knowledge of the media production domain to content analysis and synthesis of video summaries. The research aims at finding a formal model of media production practices that can facilitate automatic analysis of narrative video but also, if not especially, improve the effectiveness of the summarization process.

1.1 Focus of the thesis

We address the problem of designing an automatic system along with novel algorithms that can create efficient representations of narrative video content items to assist users in choosing what to watch among vast collections of videos.

1.1.1 Research questions

At the foundation of the research questions we address, there is the assumption that aesthetic phenomena including light, color, motion, sound, and representations of space and time, play a fundamental role in shaping the message conveyed by video. During production, these media elements are employed following precise usage patterns also known as *film grammar* [Phillips, 1999]. For example, to convey the message that two persons are involved in a dialogue, it is a convention to use close-up shots of the two persons alternated with medium shots showing the two persons together in the same location. Other conventions regard for example using the camera angle for event intensification: when a camera takes a looking-up position (low-angle) with respect to an object, person or event, the object or the person seems to be more powerful than when the camera looks at the event straight-on (eye level) or down (high-angle) [Phillips, 1999].

A new research area called *computational media aesthetics* [Dorai and

Venkatesh, 2002; Dorai and Venkatesh, 2003] is emerging that tries to make use of aesthetic principles and elements of media production for indexing, searching and browsing multimedia. Media aesthetics differs considerably from the traditional aesthetic theories that try to formalize what is beautiful and artistic and what is not. It is more concerned with how the audience perceives certain aesthetic variables and their combinations in television and film productions. It differs also from semiotic theories that analyze film and video as text to discover how its signs function and ultimately create higher-order meaning. Media aesthetics deals with the properties and structure of basic elements such as light, space, time, and sound [Dorai and Venkatesh, 2002]. Our hypothesis is that the manipulation of these syntactic elements significantly contributes to the semantic meaning carried by the video medium. We propose to leverage these syntactic elements to achieve better semantic understanding of video. In this context, the first question that we address is:

1. How can media production knowledge be modeled for the analysis of narrative video content aiming at automatic summarization?

The previous question can be better investigated if we restrict somehow the scope to a specific application domain. We are interested in the automatic generation of *video previews*, a particular type of summary that helps users in selecting content in large collections. In this context, a new question arises:

2. What are the users' requirements with respect to video previews?

Given a suitable model of media production practices and aesthetic principles, a third question arises:

3. Which approach should be adopted for the automatic creation of efficient video previews?

The approach should be validated by verifying that its results fulfill the original requirements. This should be done taking into consideration the user needs. Evaluating and benchmarking video summarization algorithms and methods is a difficult but important problem that should not be disregarded. Therefore we address also the following question:

4. How can we evaluate video summarization results taking into consideration the users' point of view?

1.2 Outline of the thesis

The rest of the thesis is structured as follows.

In Chapter 2, starting from a general description of a typical video summarization system, we propose a taxonomy for the classification of video summarization methods and systems. We then apply it to the existing body of related work in video summarization to provide an extensive overview and a comparative analysis.

In Chapter 3 we introduce the problem of automatic generation of video previews. We provide and analyze a list of user requirements collected from related literature and elicited by means of user studies.

In Chapter 4 we present our formal model of the narrative video summarization problem. The requirements and concepts informally introduced in the previous chapters are formalized to obtain a suitable mathematical formulation of the problem. The automatic creation of a video preview is formulated as the problem of selecting a subset of a given duration of the original content item that satisfies a set of constraints and that maximizes an objective function.

In Chapter 5 we describe our solution approach. Multimedia content analysis and film production domain knowledge are applied to analyze the content to be included in the video preview. A generative approach is used for the composition of a preview that satisfies all requirements.

Our solution approach is validated in Chapter 6 by means of a user study.

Chapter 7 is devoted to conclusions and discussion of future work.

1.3 Thesis contribution

The main contributions of the research described in this thesis are:

- Identification and categorization of an extensive set of requirements for automatic video summaries based on a broad analysis of literature and user panels;
- A formalized *generative method* for automatic video summarization based on optimization;
- A methodology to validate video summarization algorithms by means of user studies.

2

Video summarization: definitions and related work

In this chapter we describe the video summarization research domain and achieve new levels of understanding. For this purpose in Section 2.1 we provide definitions of the terms that are used in the research community. In Section 2.2 we define a taxonomy of classification criteria that we use to classify previous work on video summarization. In Section 2.3 we discuss validation and evaluation of summarization algorithms and systems. In Section 2.4 we present a comparative analysis of related work based on our taxonomy.

2.1 Definitions

Video summarization is the process of condensing video content into a shorter descriptive form. Different definitions of video summarization exist addressing slightly different issues and for summaries that serve different purposes. As we will later discuss, there is a variety of flavors that have been considered under the topic of summarization: *video skimming*, *highlights*, and various types of *multimedia summaries*. We distinguish between local summaries for part of a programme (e.g. for a scene), global summaries for the entire programme, and meta-level summaries of a set of programmes.

2.1.1 Flavors of video summaries

Keyframe-based summary has been a popular approach that was investigated extensively by researchers. Keyframes are representative images capturing the salient information in a video *shot*. A *shot* is a continuous recording by a camera without switching on and off. The summary is usually presented in a static form as a storyboard or in a dynamic form as a slide show. The MPEG-7 international standard has a multimedia summarization description scheme including both static and dynamic summaries [Salembier and Smith, 2001]. [Dimitrova et al., 1997] presented a keyframe-based summary for digital video recorders shown in Figure 2.1.

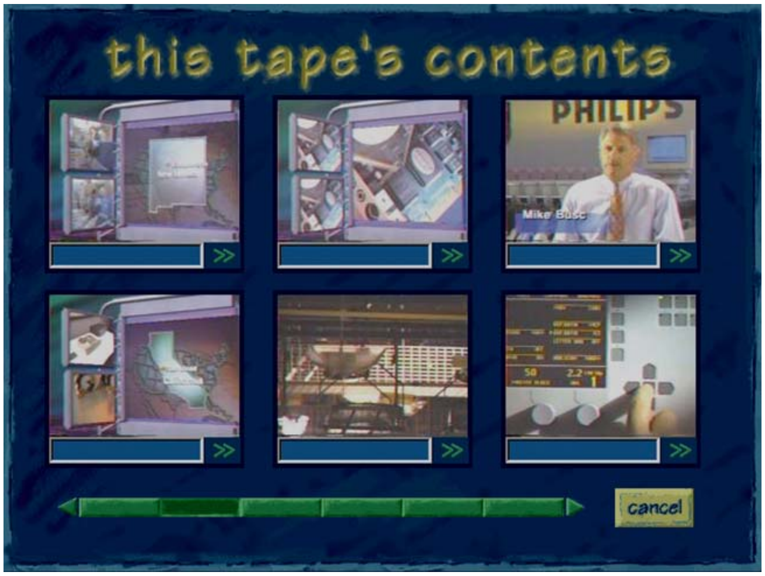


Figure 2.1. Example of keyframe-based summary.

Video skim is a temporally condensed form of a video stream that preferably preserves the most important information. It is typically a single video generated by combining a set of short portions automatically selected from the original video. A method for generating video skims based on scene analysis and scene visual complexity is presented by [Sundaram et al., 2002]. [Wactlar et al., 1996] condensed long videos into short skims by combining video segments that contain automatic speech recognition (ASR) transcripts matching the user's query terms. [Ma et al., 2002] proposed an *attention model* that includes visual, audio, and text modalities for summarization of videos. As synonyms to video skims, researchers have used the terms *preview* and *trailer* in the literature. However, video previews and video trailers are not meant to convey all the information of the original content and are not a comprehensive summary or an abstract. The difference between a trailer and a preview is that a trailer is a commercial teaser showing specific appealing and attractive segments of the video with the purpose of attracting audience, while a preview aims at conveying key-aspects of a program to allow users to quickly see what it is about.

Video highlights is a form of summary that aims at including the most important events in the video. Unlike video skims that are presented as single condensed videos, the highlight-based summary is usually presented as an organized list or table of interesting events (e.g. pitching, running, goals) along with some associated metadata (e.g. event statistics). In some cases, summaries of the events are further augmented by expanding the visual background (e.g. *visual mosaic*) or by using geospatial maps, or time lines. In [Irani and Anandan, 1998], summaries of object trajectories are overlaid on top of the soccer or tennis field. In [Christel et al., 2002] events in the broadcast news videos are plotted over a world map or a time line to summarize the relations between the events. Various methods have been introduced for extracting highlights from specific sub-genres of sports programmes: goals in soccer video [Dagtas et al., 2000; Li and Sezan, 2001], hits in tennis video or pitching in baseball [Zhong et al., 2001], important events in car racing video [Petkovic et al., 2002], touch downs in football [Dagtas et al., 2000], abnormal activities in surveillance videos [Connell et al., 2004] and others.

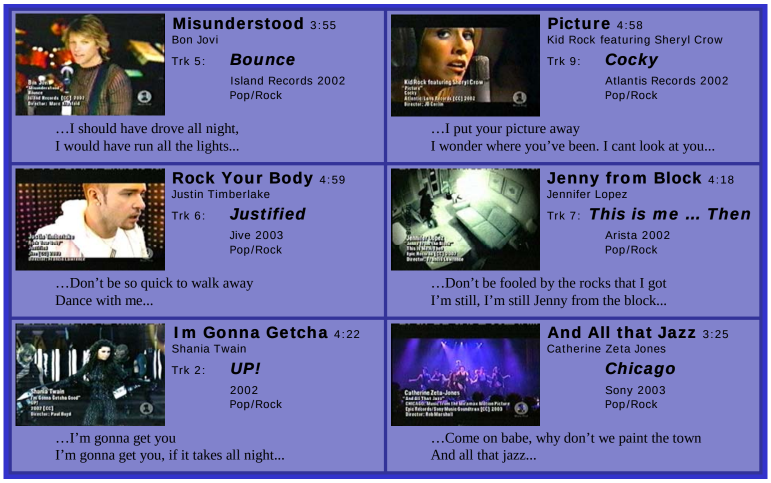
Multimedia video summary is a collection of audio, visual, and text segments that preserve the essence and the structure of the underlying video. A multimedia system that analyzes and summarizes music video clips is described in [Agnihotri et al., 2004]. Figure 2.2 shows a snapshot of the user interface of the system.

A multimedia video summary of audio-video presentations is presented in [He et al., 1999]. The summarization system uses slide-transitions in video, pitch in audio and user interaction with presentations in order to generate a multimedia summary. Another multimedia summary [Ebadollahi et al., 2002] is developed in the health care context in which patient records, medical literature, textbooks, and ultrasonic videos of the heart are integrated in the overall summary. Layouts of the constituent media components (e.g. text summaries, term definitions, video storyboards, and video skims) are customized according to the specific medical concepts being explored and the specific profiles of the users.

2.1.2 Levels of video summaries

Depending on whether the summarization is applied to parts of the video, the whole video or even multiple videos, the *summarization level* is respectively defined as: *local*, *global*, or *meta-level*.

Scene level (local) summarization compresses segments at the scene level and tries to eliminate redundant information considering local information. A scene is a



Misunderstood 3:55
Bon Jovi
Trk 5: ***Bounce***
Island Records 2002
Pop/Rock

…I should have drove all night,
I would have run all the lights...

Picture 4:58
Kid Rock featuring Sheryl Crow
Trk 9: ***Cocky***
Atlantis Records 2002
Pop/Rock

…I put your picture away
I wonder where you've been. I cant look at you...

Rock Your Body 4:59
Justin Timberlake
Trk 6: ***Justified***
Jive 2003
Pop/Rock

…Don't be so quick to walk away
Dance with me...

Jenny from Block 4:18
Jennifer Lopez
Trk 7: ***This is me … Then***
Arista 2002
Pop/Rock

…Don't be fooled by the rocks that I got
I'm still, I'm still Jenny from the block...

Im Gonna Getcha 4:22
Shania Twain
Trk 2: ***UP!***
2002
Pop/Rock

…I'm gonna get you
I'm gonna get you, if it takes all night...

And All that Jazz 3:25
Catherine Zeta Jones
Chicago
Sony 2003
Pop/Rock

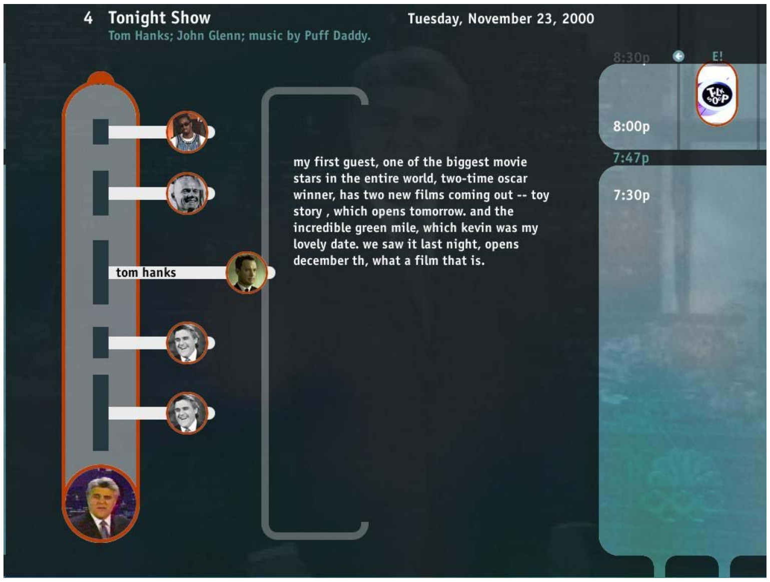
…Come on babe, why don't we paint the town
And all that jazz...

Figure 2.2. Multimedia summary of a collection of music video clips.

set of shots showing videos usually taken at the same location, and is typically used to capture information about the environment or events occurring at the location. Sundaram's *constrained utility maximization* method [Sundaram et al., 2002] maps the complexity of a shot into the minimum time required for its comprehension in order to summarize a scene.

Surface level (global) summarization looks for cues that have been predefined based on the domain knowledge. For example, [Agnihotri et al., 2001] present a surface level summarization method for talk shows that includes representative elements for the host portion and each of the guests (see Figure 2.3). The domain knowledge is the structure of the programme and the different cues that signal important segments. The method utilizes the text transcript for analyzing the talk show structure, then applies temporal constraints and domain knowledge in order to segment the important section segments. Any summarization method based on detecting elements of the content in general belongs to this approach. For a shot change based summarization [Dimitrova et al., 1997] the vocabulary is “radical change in signal properties”.

Entity level (global) summarization approaches build an internal representation of video, modeling 'video entities' and their relationships. A natural choice for the video entity is a shot. Other examples of video entities include people, locations, and objects (e.g. vehicles). These tend to represent patterns of connectivity in the video (e.g. graph topology) to help determine what is salient and then create a summary containing the most salient entities. Relationships between entities include similarity (color similarity, similar faces) and co-occurrence. [Li et al., 2001] presented a method of clustering frames with similar color in order to gen-



4 Tonight Show
Tom Hanks; John Glenn; music by Puff Daddy.
Tuesday, November 23, 2000
8:30p
E!
8:00p
7:47p
7:30p
tom hanks

my first guest, one of the biggest movie stars in the entire world, two-time oscar winner, has two new films coming out -- toy story , which opens tomorrow. and the incredible green mile, which kevin was my lovely date. we saw it last night, opens december th, what a film that is.

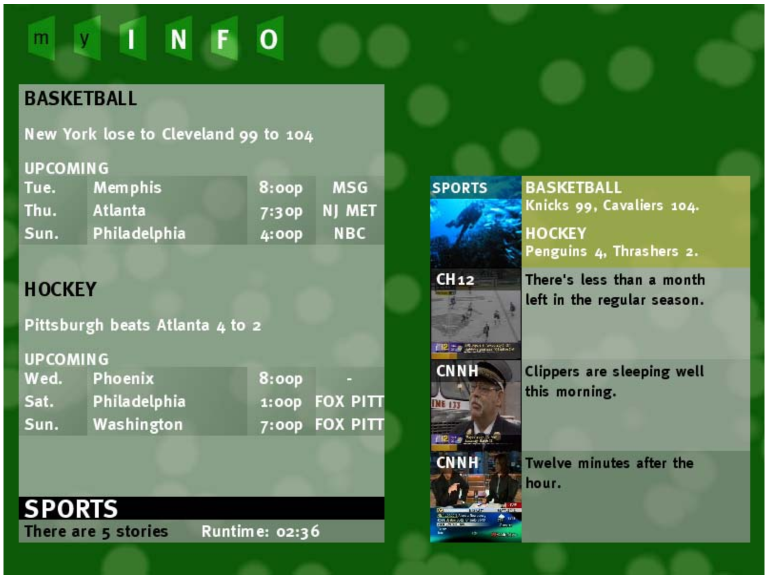
Figure 2.3. Surface level talk show summarization example.

erate summaries of programmes. [Tsoneva et al., 2007] have proposed a method to create a summary that preserves the story line of a video item based on representing semantic relationships among video scenes with a graph. A relationship between non-consecutive scenes is established by detecting co-occurrences of keywords in the speech transcript and the presence of the same characters. Scenes are then ranked based on their relationships with respect to the graph topology. A summary is created by selecting the scenes that have the highest rank.

Meta-level summaries provide an overview of a whole cluster of related videos. For example, a meta-summary of all available news items from Web and TV sources is provided by the MyInfo system as shown in Figure 2.4 [Dimitrova et al., 2003]. The summary of the news items is extracted by text summarization and visual analysis of the reportage and presented according to a personal profile.

2.2 Dimensions for classification

Video is a very rich medium including moving images, audio and sometimes text. Information is transmitted using combinations of multiple perceptual channels and cues. For this reason, its summarization can be approached from many different points of view using a large variety of techniques and aiming at providing solutions to help users accomplish different tasks. Furthermore, video can be classified into a large number of types, genres and sub-genres (e.g. TV programmes, produced



BASKETBALL

New York lose to Cleveland 99 to 104

UPCOMING

Tue.	Memphis	8:00p	MSG
Thu.	Atlanta	7:30p	NJ MET
Sun.	Philadelphia	4:00p	NBC

HOCKEY

Pittsburgh beats Atlanta 4 to 2

UPCOMING

Wed.	Phoenix	8:00p	-
Sat.	Philadelphia	1:00p	FOX PITT
Sun.	Washington	7:00p	FOX PITT

SPORTS

There are 5 stories Runtime: 02:36

SPORTS	BASKETBALL Knicks 99, Cavaliers 104. HOCKEY Penguins 4, Thrashers 2.
CH12	There's less than a month left in the regular season.
CNNH	Clippers are sleeping well this morning.
CNNH	Twelve minutes after the hour.

Figure 2.4. Meta-summary of news programmes in MyInfo.

film, home videos, educational, surveillance, multimedia presentations, etc.), each one having a typical usage model and particular characteristics.

In order to classify existing approaches to video summarization, we propose a set of criteria for classification as depicted in Figure 2.5. These criteria have been thought while reading literature and trying to get an overview of existing work in the domain of video summarization. At the highest level, the classification model has three branches:

1. Usage.
2. Content.
3. Method.

In the next three sections we present our classification along each one of these branches. In Section 2.2.4, we present a comparative analysis of existing published works on video summarization using the proposed criteria.

2.2.1 The usage branch

Video summarization addresses the generic problem that arises from a specific need of the users: to handle video more efficiently, thus doing more with fewer resources. There are three related aspects to the manner in which summaries are utilized: *task*, *intent* and *user's category*. Although summaries are used for different applications, the tasks are fairly general: users are searching and selecting,

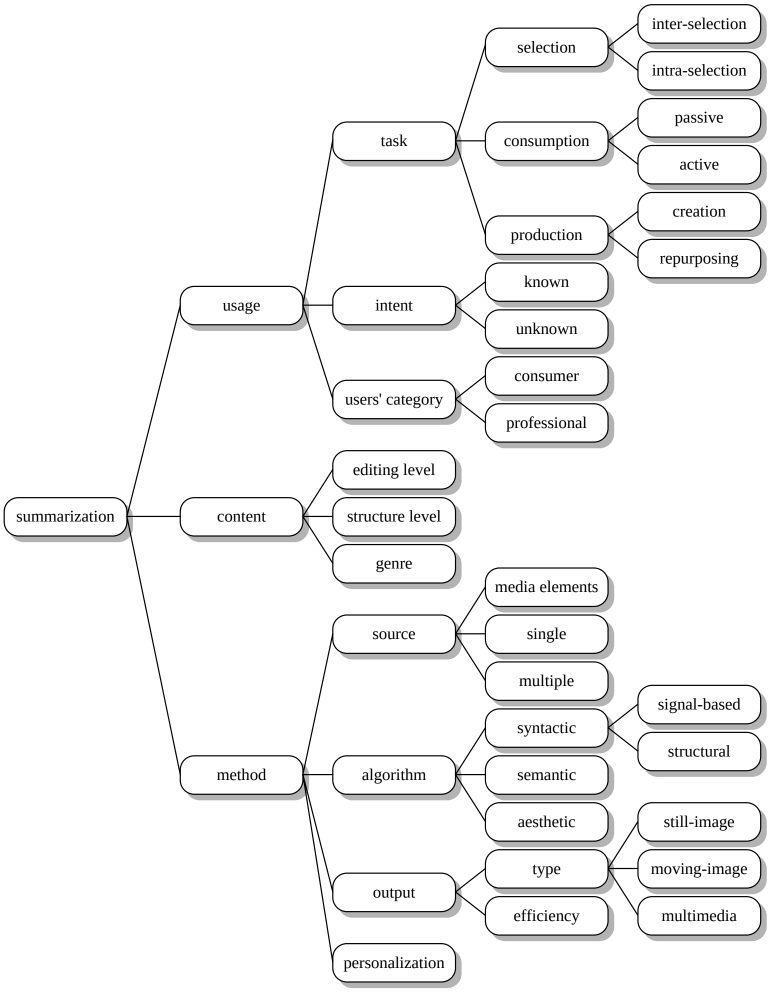


Figure 2.5. Criteria for classifying video summarization methods and systems.

consuming, or producing new content. The intent of the users in these tasks also plays an important role in the creation and usage of summaries. Users' background and profession define the users needs and manner of usage.

Task

The goal of a summary differs substantially depending on the task users need to perform. Generally speaking, tasks can be divided into *selection*, *consumption* and *production*. In this classification, *selection* includes:

- *inter-selection*, meaning selecting a single item among multiple items available in a collection;
- *intra-selection*, meaning selecting a portion of a single item.

By providing a condensed and descriptive form of video content, summaries can help users selecting (either searching or browsing) among a large collection of items. Inter-selection tasks require global summaries that are extracted using either surface level or entity level summarization systems. Intra-selection summaries offer substantial benefits in selecting and accessing scenes out of a single video [Li et al., 2000; Dagtas et al., 2000; Li et al., 2001]. This requires local scene level summarization methods. A summary targeting content selection should allow users to quickly see what the content is about, remember whether it has been seen before or decide whether to choose it for consumption. In order to fulfill these requirements, *inter-selection* summaries make use or explicitly include in the results additional metadata (e.g. title, textual summary, etc.) that is bundled together with the content. In general, during selection, users need to evaluate quickly many different content items, thus a summary created for the selection task should require very short time for comprehension. Most summarization systems we analyzed (see the comparative analysis in Section 2.2.4) were designed for quick inter-selection [Wactlar et al., 1996; Uchihashi et al., 1999; Hermes and Schultz, 2007]. Many systems allow at the same time inter and intra selection [Yeo and Yeung, 1997; Hanjalic and Zhang, 1999; Peker et al., 2006; Wang et al., 2007].

Summaries can also be generated for more efficient consumption of content. These summaries are synthesized *content digests* that can be consumed more efficiently in place of the original content. Content consumption can be *active* or *passive*. Active consumption involves users that lively interact with content - *lean forward* mode - while passive consumption is the typical *lean backward* way of experiencing content. Summarization methods targeting the consumption tasks need to ensure that the essential information the content creators meant to convey to the content consumers is preserved. The content digests can be in the form of meta-level summaries that contain information present in a number of sources.

In surveying existing summarization systems for active or passive usage, it is noticeable that most systems do not clearly take into consideration any requirement in terms of usage modality. Summaries meant to be consumed in a passive way are usually in the form of video skims [Hermes and Schultz, 2007; Mekenkamp et al., 2002; Paulussen et al., 2003; Adami and Leonardi, 2001; Sundaram et al., 2002; Syeda-Mahmood and Ponceleon, 2001; Ionescu et al., 2006; Chen et al., 2004; Peker et al., 2006; Benini et al., 2007] while summaries that require active users' participation are typically embedded in multimedia documents [Hauptmann and Smith, 1995; Yeo and Yeung, 1997; Petkovic et al., 2002; Uchihashi et al., 1999; Yahiaoui et al., 2001; De Silva et al., 2005]. Highlight-based summaries can be intended for both passive and active usage.

The production tasks deal with creation of new content from scratch (*creation*) or with reuse of existing material (*repurposing*). Summaries aiming at supporting content producers need to be tailored to their specific needs. Authoring is a creative process that involves not only semantic elements but also aesthetic aspects. Depending on users' category, either professional or consumer, creating new content based on automatic summaries requires attention to intangible qualities of the summaries as well as to the semantic information that is included. Perhaps, for this reason or because the multimedia research community seems to pay more attention to the selection and consumption tasks, not many systems generate useful summaries for content creators. Only [Hauptmann and Smith, 1995; Lienhart, 2000; Russell, 2000] take somehow into consideration the requirements of content producers.

Intent

The usage of a summary depends on the intent or goal users set in performing a specific task: *known* or *unknown intent*. A user with a known goal in manipulating or consuming video expects summaries to adhere to requirements that arise from its specific and present intent. Users without a known goal will not pose strict requirements on the informative elements included in a summary. Most of the summarization techniques are designed for users that do not have a specific goal in mind. In this sense, all summarization systems developed so far are generic and do not aim at satisfying the need of a specific individual or group of users. However some semi-automatic systems (e.g. [Farin et al., 2002; Toklu et al., 2000]) require explicit input from active users that must have some specific goals in their minds. For example in [Toklu et al., 2000] users interested in particular parts of the video can interact with the summarization system to change or refine the results of the segmentation process prior to summarization. The final summary will be more focused on the parts indicated explicitly by the user.

User’s category

The usage of video and summaries changes drastically depending on the context: *professionals* have probably different requirements than home users or generally speaking, *consumers*. The differences between the professional world and the consumer world are reflected in properties, form and substance of video summarization for professional or consumer applications. As for active and passive consumption (see Section 2.2.1), most summarization systems neglect the distinction between consumers and professionals. All surveyed systems are meant to be useful for consumers although probably most of them expect sophisticated PC users.

2.2.2 The content branch

There is an enormous assortment of video types and genres for different application domains and with different intrinsic properties. Current summarization tools can be classified based on the characteristics of the content they can process.

Level of editing

Across different video genres and application domains, content is produced at different *levels of editing* and contains different *levels of structure*. A piece of content has a high *level of editing* when its production requires considerable and conscious human intervention and manipulation during or after capturing and synthesis. If the video production is the result of a continuous live video capturing process, it usually has a low level of editing.

Level of structure

Independent of the production process, content can have different *levels of structure*. An example of structured video content is a news programme that is broadcast periodically at a precise time and has a very distinctive and consistent pattern of events such as anchorperson item followed by reportage video. Another example is a tennis match that is organized according to codified rules into sets, games, etc.

Summarization methods have exploited video structure and editing levels to the point that some techniques heavily rely on them and can be considered dedicated approaches tailored to a particular content genre (e.g. news).

Genre

Automatic video summarization techniques can be classified depending on their specialization or applicability to specific domains or *genres*. Application domains such as video surveillance, television, home videos, video presentations, etc. capture content meant to convey different types of messages and have different levels of editing and structure.

The most popular domains are television programmes and films. Most summarization systems target specifically these domains while little attention has been paid so far to develop dedicated systems for continuous live capture (e.g. wearable video [Aizawa et al., 2001]), video presentations (e.g. [He et al., 1999; Russell, 2000; Yokoi and Fujiyoshi, 2006]), meetings (e.g. [Erol et al., 2003]), and health care (e.g. [Ebadollahi et al., 2002]).

Developing summarization algorithms for a specific domain allows exploiting the structure that is typical for that genre (domain knowledge). Many systems are designed based on assumptions and heuristics that are valid within a domain only and fail when applied to other domains.

Although most of the existing techniques seem to be limited to a particular domain, some approaches claim to be independent and sufficiently generic for multiple application domains [Hanjalic, 2003; Chiu et al., 2000; DeMenthon et al., 1998; Farin et al., 2002; Hanjalic and Zhang, 1999; Ma et al., 2002; Nam, 1999; Omoigui et al., 1999; Peker et al., 2001; Syeda-Mahmood and Ponceleon, 2001; Uchihashi et al., 1999; Lee et al., 2006; Gong and Liu, 2001; Mekenkamp et al., 2002; Paulussen et al., 2003]. These systems usually adopt a purely signal-based method (see Section 2.2.3).

Genre-dependent summarization techniques exploit specific structures and properties of particular genres. Television programmes represent a category of video that can be easily divided into many different genres. For this reason, most summarization systems that target this application domain are also specialized to handle a single or multiple specific genres. The most popular genres are *news* [Kim et al., 2003; Agnihotri et al., 2001; Dimitrova et al., 2003; Hauptmann and Smith, 1995; Jasinschi et al., 2001; Merlino and Maybury, 1999; Merialdo et al., 1999; Toklu et al., 2000], *sport* [Petkovic et al., 2002; Zhong et al., 2001; Ekin et al., 2003; Babaguchi et al., 2001; Dagtas et al., 2000; Petkovic et al., 2002; Li and Sezan, 2001; Kolekar and Sengupta, 2006], *sitcoms* [Aner and Kender, 2002; Li et al., 2001; Yahiaoui et al., 2001; Jung et al., 2004], *talk shows* [Agnihotri et al., 2001; Jasinschi et al., 2001], and *feature films* [Hermes and Schultz, 2007; Pfeiffer et al., 1996; Sundaram et al., 2002; Ionescu et al., 2006; Chen et al., 2004; Benini et al., 2007].

For example, for the genre talk-shows, [Agnihotri et al., 2001] identify important segments using textual cues that are typically used only in talk-shows. For the genre feature films, [Pfeiffer et al., 1996] assume that explosions and gun shots are important events that should appear in the summary and they develop specific algorithms to detect them.

2.2.3 The method branch

Figure 2.6 shows a generic summarization system for multimedia documents. Content is analyzed in order to evaluate and select the most important parts. A summary is then composed and presented to users. In some cases, users can interact with the summary during or after its creation.

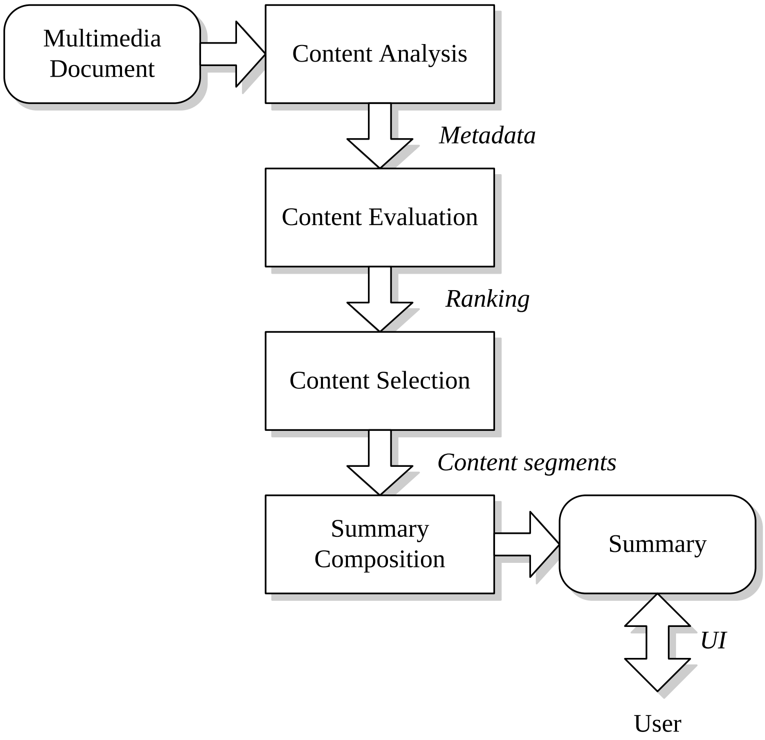


Figure 2.6. Block diagram of a generic video summarization system.

When analyzing summarization systems and methods, other sub-criteria can be defined with respect to the source of information (video, audio, text transcript, from single or multiple videos) that are considered, to the type of algorithm that is employed, to the output that is produced and to whether the summary is tailored to the specific needs and preferences of an individual or a particular group. We can further specialize the method criterion in the following sub-criteria: *source*, *algorithm*, *output*, and *personalization*.

Source

Video is usually delivered as a collection of images with synchronized audio and text plus additional metadata. Summarization systems can be classified depending on the type and source of input *media elements* that they employ (e.g. visual, audio, subtitles, user input, metadata bundled with the content or extracted from related documents retrieved from the Internet, etc.) and on the actual content items they consider in input.

Almost all video summarization systems analyze the audio, visual and text portions of the video signal. Audio can also be separately analyzed to detect specific cues or events such as explosions [Pfeiffer et al., 1996], goals [Dagtas et al., 2000], etc. Because generic semantic understanding based only on audio and video analysis is still far from being reachable, many systems rely on speech transcripts obtained from closed captions [Agnihotri et al., 2001; Dagtas et al., 2000; Farin et al., 2002; Li et al., 2000; Li et al., 2001; Paulussen et al., 2003; Toklu et al., 2000; Tsoneva et al., 2007] or automatically generated using speech recognition [Hauptmann and Smith, 1995; Zhu and Penn, 2006]. There are examples of semi-automatic systems that use extra information related to users' behavior in consuming and browsing video [He et al., 1999; Syeda-Mahmood and Ponceleon, 2001] or entered directly by a user [Farin et al., 2002; Toklu et al., 2000]. A few systems have investigated the possibility of using extra information from a parallel channel related to a specific video. For example, [Aizawa et al., 2001] employ as parallel information source, cues derived from measuring the brain waves of a user during live video capture.

Summaries can be extracted from a *single* or *multiple sources*. Summarizing multiple related videos (e.g. episodes of the same TV series) can be considered part of the multiple sources category. So far, most of the researchers have been targeting single source summarization. The only systems that take into consideration more than one source are [Dimitrova et al., 2003] and [Yahiaoui et al., 2001], but they do not eliminate redundant information across different videos to provide more concise summaries. In [Xie et al., 2005], recurrent news topics are automatically discovered from multiple broadcast video channels. Stories in each video programme are automatically mapped to the topic clusters based on multi-modal feature analysis and statistical pattern mining.

Algorithm

The algorithms employed in current summarization systems fall into three categories: *syntactic* based, *semantic* based or *aesthetic* based. Syntactic based algorithms try to determine what is relevant in the source to be summarized and decide how to condense its content using syntactic properties of the medium.

Syntactic based algorithms can further be distinguished depending on the use of *signal-based* properties or *structural* information. Signal-based algorithms rely on the results of the audio and video signal analysis for selecting relevant parts of the original content to include in the summary. Cues obtained by signal analysis include presence and location of faces and superimposed text, camera motion, shot and story boundaries, speaker change, etc. Most approaches rely on the combination of multiple heuristics either coded in rules (e.g. [Lienhart, 2000; Mekenkamp et al., 2002; Paulussen et al., 2003; Pfeiffer et al., 1996]) or embedded in importance functions (e.g. [Hauptmann and Smith, 1995; Ma et al., 2002; Uchihashi et al., 1999]) for the classification and selection of relevant content. These are examples of surface level summarization techniques.

Some methods rely on the consideration that a summary should contain little redundancy and try to approach the problem in a numeric way by segmenting and classifying video according to certain models of similarity [Aner and Kender, 2002; Chiu et al., 2000; DeMenthon et al., 1998; Farin et al., 2002; Gong and Liu, 2001; Hanjalic and Zhang, 1999; Yeo and Yeung, 1997; Uchihashi et al., 1999; Lee et al., 2006]. These are examples of entity level summarization techniques.

Some systems are designed for handling specific types of content that presents a repetitive clearly identifiable structure. Structural cues can be employed by refined heuristics to locate interesting events and to include parts deemed relevant in the summary [Agnihotri et al., 2001; Dimitrova et al., 2003; Hauptmann and Smith, 1995; He et al., 1999; Li et al., 2001; Russell, 2000; Saarela and Merialdo, 1999; Toklu et al., 2000; Kolekar and Sengupta, 2006]. Video highlights are extracted based on these interesting events. In practice, many systems consider both signal-based properties and structural cues.

Semantic based approaches determine what is relevant in the source based on some sort of semantic understanding of the content. Currently many systems rely on textual information associated with the video for obtaining semantic cues. If textual information is available from speech transcription (either manual or automatic) then the summarization problem is reduced to the textual domain for which many techniques have been developed in the field of textual information retrieval (e.g. [Salton et al., 1997]). Most of the summarization systems that have been developed in the research community are hybrid systems: they employ aspects of syntax and semantics.

Aesthetic based approaches try to mimic human perception of aesthetics in videos by making use of measurable properties of visual and audio signals [Russell, 2000; Sundaram et al., 2002]. The aim is to design better algorithms for determining the most salient parts in a video content to include in the summary. Related work includes analyzing and representing the *affective* properties of video content such as the intensity and type of feeling that viewers experience when watching video [Hanjalic and Xu, 2005]. In [Hanjalic, 2006] Hanjalic uses affective content analysis to detect exciting moments in sport and composing a summary containing only the most exciting highlights.

Output

Different methods for video summarization can produce radically different types of summaries: *still-image, moving-image* and *multimedia* summaries. A still-image summary is a small set of images extracted or generated from the original video. A moving-image summary is a video composed of a set of image sequences with accompanying audio extracted or generated from the original content but of considerably shorter duration. Multimedia summaries combine still-images with moving-images, audio and text to provide a multimedia presentation of the summary.

Most still-image summaries are a collection of key-frames extracted from the original video displayed simultaneously in a so-called *storyboard* or *pictorial overview* (e.g. [Chiu et al., 2000; DeMenthon et al., 1998; Farin et al., 2002; Hanjalic and Zhang, 1999; Merialdo et al., 1999; Saarela and Merialdo, 1999; Yahiaoui et al., 2001]). The key-frames can be arranged in a spatial layout with different sizes depending on their relative importance (e.g. [Yeo and Yeung, 1997; Uchihashi et al., 1999; Wang et al., 2007]). Furthermore they can be composed into a multimedia presentation and offer access to the parts of the original video to which they correspond (e.g. [Agnihotri et al., 2001; Dimitrova et al., 2003; Hauptmann and Smith, 1995; He et al., 1999; Jasinschi et al., 2001; Li et al., 2001; Merlino and Maybury, 1999; Petkovic et al., 2002; Toklu et al., 2000]). New visualizations for presenting the output are proposed in [Christel et al., 2002] where the results of a query are presented as a collage, which is a presentation of text and images derived from multiple video sources. Another flavor of still-image summaries is the so-called *mosaics* (e.g. [Taniguchi et al., 1997; Irani and Anandan, 1998; Aner and Kender, 2002]). They are still-images obtained by concatenating successive frames captured during some sort of camera movement. Using global motion estimation techniques, it is possible to calculate the amount of displacement and the geometric transformation required to combine multiple successive frames into one panoramic view of a scene. Figure 2.7 shows two examples of mosaic images generated from videos.

Being composed out of multiple frames, a single mosaic contains more information than one single key-frame. However, due to its dynamic and temporal nature, video can be better represented with moving images. [Ding et al., 1999] found that video summaries including both text and imagery are more effective than either modality alone.

Video skims are obtained by concatenating portions of the original video (e.g. [Aizawa et al., 2001; Babaguchi et al., 2001; Dagtas et al., 2000; Gong and Liu, 2001; Lienhart, 2000; Ma et al., 2002; Mekenkamp et al., 2002; Omoigui et al., 1999; Adami and Leonardi, 2001; Russell, 2000; Sundaram et al., 2002; Syeda-Mahmood and Ponceleon, 2001; Benini et al., 2007]). Other forms of moving-

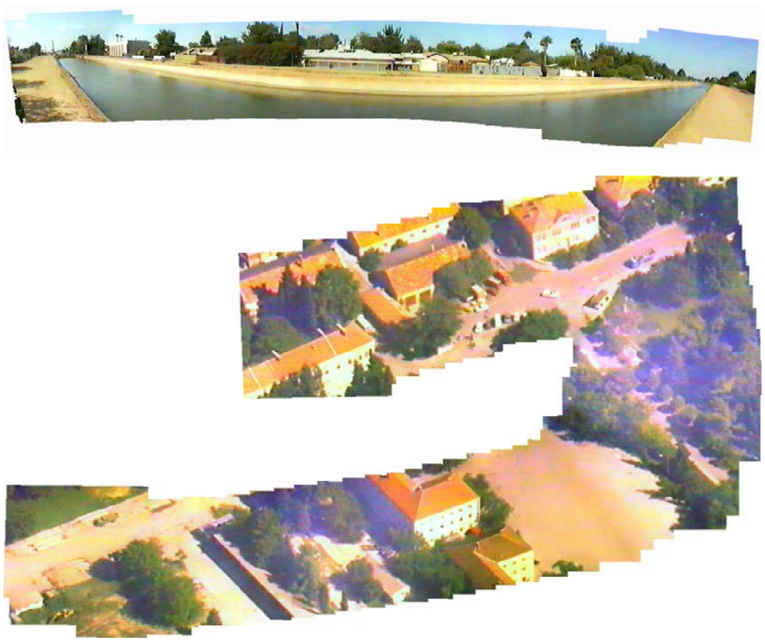


Figure 2.7. Two example of mosaics generated from videos.

image summaries are presented in [Nam, 1999; Paulussen et al., 2003; Peker et al., 2001; Peker et al., 2006] where the playback speed is changed to achieve a shorter duration while preserving relevant information.

The output of a summarization system can be further classified depending on the time required for consumption compared to the length of the original video: *high* vs. *low efficiency*. Most of the systems can be considered highly efficient because they have a very low consumption time (e.g. pictorial summary). [Nam, 1999; Omoigui et al., 1999; Peker et al., 2001] produce shorter versions of the original video that can be consumed in place of the entire content. However, the time required to consume the summary is still considerably high when compared, for example with still-image summaries (e.g. [Yeo and Yeung, 1997; Uchihashi et al., 1999]).

Personalization

Video summarization literature covers generic methods of summary extraction. Most of the works in summarization do not take into account the specific needs and preferences of users. Actually the ideal summary should be tailored to the specific needs of an individual or a group of users for a given task, time, context, and environment. A study that looks at users generated text summaries concluded that individual users select very different things to be included in the summary [Salton et al., 1997]. This gives a strong argument for personalized summaries. [Agnihotri et al., 2003] conducted a study that looked at user requirements for personalized multimedia summaries and found that summaries need to be personalized based not only on user, but also based on task, time, and environment. However, the only attempt at personalization so far has been in the area of content selection for personalized news [Dimitrova et al., 2003; Merialdo et al., 1999; Agnihotri, 2005], music videos and talk shows [Agnihotri, 2005], or sport programmes [Babaguchi et al., 2001; Zhong et al., 2001].

2.2.4 Comparative analysis

In Figure 2.8 we map existing published summarization systems and methods according to the proposed classification dimensions. The graph reveals where research efforts have placed their focus. In the usage dimension, it is apparent that content selection and consumption have been explored in detail while content creation and repurposing have been neglected. Also, fewer systems have been devoted to professional users rather than consumers. Most systems cater for unknown user intent, which means that they present to the users a generic summary that is independent of the task the user may have in mind.

With respect to the content aspects, the most evident is the focus on produced video in the TV domain. Summarization of home video and life recording from wearable video has not been extensively explored yet. Most of the content analyzed in the TV domain is geared toward genre specific methods. Methods have been developed to address specific genres, especially highly edited video with high level of inherent structure.

Summarization methods have explored input from visual video signal much more than other modalities. Single source summarization systems are much more prevalent in the literature than multiple source systems. Systems that consider multiple sources of input are just beginning to be tackled. Signal-based approaches prevail while there are some attempts to use structural aspects of video as well. Semantics based methods are gaining momentum as evident from the table, while aesthetic methods are just starting to be developed. Output elements are mostly based on still or moving images and multimedia summaries are not in the focus yet. From efficiency point of view, the aim of most surveyed methods has been to provide a high level of efficiency as far as the time required for consuming the summary is concerned. The graph also reveals that personalization aspects are fairly new in multimedia summarization.

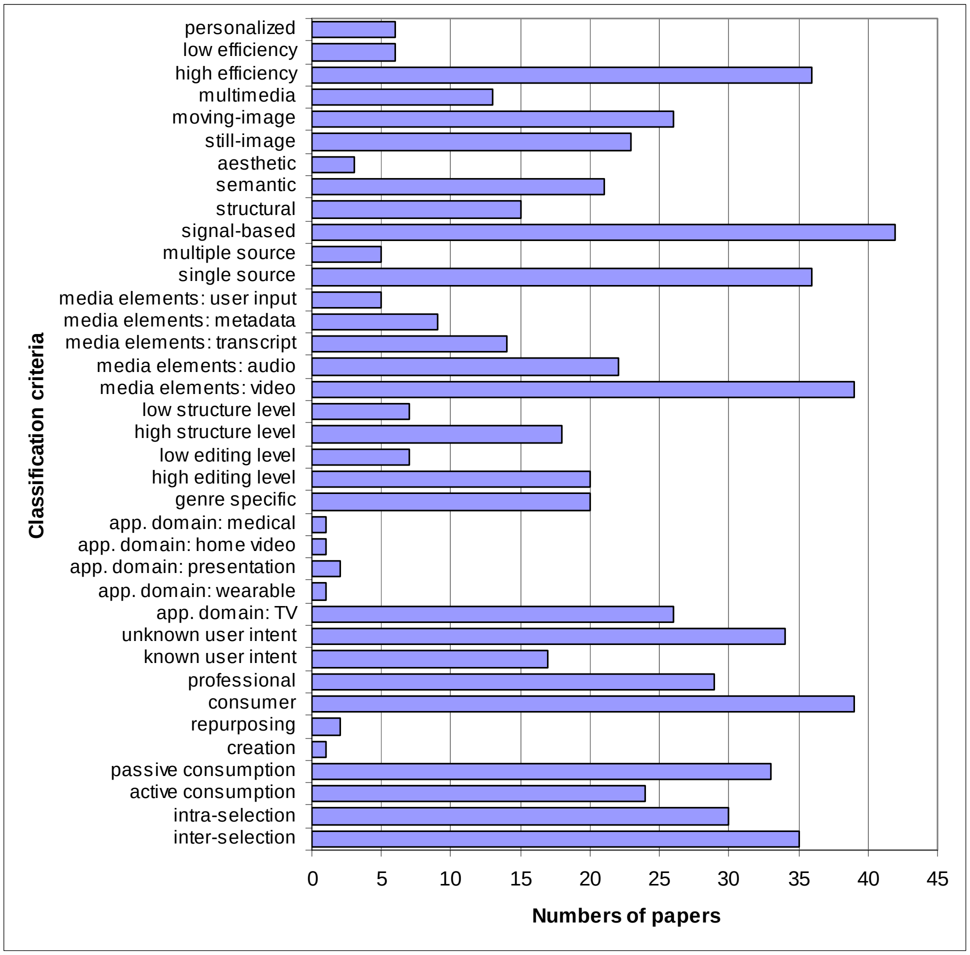


Figure 2.8. Comparative analysis of 45 publications on video summarization systems. Each leaf of the classification tree shown in Figure 2.5 represents a classification criterion derived from the four proposed basic dimensions.

2.3 Evaluation and validation

The evaluation is probably the most difficult part to set up for video summarization. In all the publications that were surveyed a similar sentiment was echoed. Studies show that users rarely agree on what a summary should include [Salton et al., 1997]. The ideal summary is hard to construct and rarely unique. Just as there are many ways to describe an event or a scene, users can produce many generic or user-focused summaries that they consider acceptable. Objective evaluation of results and benchmarking of different algorithms are still open challenges. In this section, we try to bring into focus the strategies required for evaluating summarization systems. A summarization system needs to be evaluated in the following aspects: algorithmic and user evaluation.

2.3.1 Algorithmic evaluation

In the algorithmic evaluation, the question is whether the algorithm extracts the right content elements, has right coverage, etc. [Agnihotri et al., 2001] benchmarked their talk-show summarization algorithm based on its accuracy to detect the guests in the talk show and their areas of fame (movie, music etc.). Additionally, the correctness of the introduction sentence that is included in the summary was evaluated.

One of the most advanced initiatives is TRECVID [Nist, 2007]. TRECVID offers good benchmarking opportunities for evaluation of some of the algorithms used by summarization systems. The goal of the project is to promote progress in content-based retrieval from digital video via open, metrics-based evaluation. In 2003 and 2004 a dataset of 120 hours of video was assembled. In 2005, 160 hours of news videos of three different languages (English, Chinese, and Arabic) were collected from multiple channels. The tasks include shot boundary detection, story segmentation, high-level feature extraction, and search. The queries revolve around object detection and identification in the video.

In 2005 and 2006, 50 hours of *rushes* material was added to the data set [Smeaton et al., 2006]. Rushes are raw (unedited) footage created during production of movies and TV shows. They contain all the camera takes that are shot for a scene of a movie or TV show, natural sounds, crew and director's voices, camera movements between takes as well as video of clapboards. Research groups were asked to provide tools for support of exploratory search on highly redundant rushes data. There was no standard evaluation of the results.

In 2007, TRECVID contained a pilot evaluation of video summaries for rushes aimed at compressing out redundant and insignificant content [Over et al., 2007]. The evaluation of the summaries was done by a small number of users (7) based on ground truth information on what an ideal summary should contain. This

TRECVID pilot study belongs to the *intrinsic* and *ground truth-based* evaluation methods discussed in the next section.

2.3.2 User evaluation

There are three different aspects of user evaluation: *intrinsic*, *extrinsic*, and *ground truth-based*. The distinction between intrinsic and extrinsic evaluation was introduced by Mani in the domain of automatic text summarization and information retrieval [Hahn and Mani, 2000; Mani, 2001].

Intrinsic evaluation

The intrinsic or *normative* evaluation applies to a user judging the quality of the summarization method by analyzing directly the summary. Intrinsic user evaluation was performed by [Sundaram et al., 2002] who asked users to rate the coherence of different summaries. [Merlino and Maybury, 1999] asked users to rate their satisfaction on a 1–10 scale for ten different types of news summaries. The summaries that had the most detail were rated much higher than the ones that contained just keyframes. Another example of intrinsic evaluation can be found in [Ma et al., 2002] where the aspects of *enjoyability* and *informativeness* were evaluated.

Extrinsic evaluation

For extrinsic evaluation, users judge the quality of a summary according to how it affects the completion of a task, such as how well it helps them performing a selection among a large collection. Extrinsic user evaluation was performed on multiple presentations of news summaries generated by the *broadcast news navigator* system by [Merlino and Maybury, 1999]. The users had to accomplish a series of identification and comprehension tasks. [Yahiaoui et al., 2001] presented multi-episode summarization and proposed the idea of a “simulated user” that evaluates whether their algorithm selects the optimal set of frames. The definition of optimality is based on measurable color properties of images. This should enable to perform algorithmic and user evaluation in an automatic way. They argue that this measurement would be very close to what a human being would do.

Ground truth-based evaluation

In certain domains such as news and security videos, user task and usage requirements can be better defined than in other domains. In such cases, domain experts can be asked to review and generate summaries of some selected test videos. Such expert-generated summaries can then be used as the “ground truths” to evaluate the quality of summaries automatically generated by computers. For example, overlaps between the automatic summary and the ground truths could be measured in terms of the concepts, entities, and video segments covered.

2.4 Discussion

Video summarization has in the last years become a very popular research area with a number of published works that is steadily increasing. However, the research community has not yet come to a common understanding of its problems and applications.

Our analysis reveals where research efforts have been most intense and also what are the areas that have not been addressed yet. In the usage dimension, it is apparent that content selection and consumption have been explored in detail. With respect to the content aspects, it is clear that most approaches have been geared toward produced video in the TV domain. In addition, methods have been developed to address specific genres. Summarization methods have explored input from the visual video signal much more than from other modalities. Signal-based approaches to summarization are dominant. Output elements are mostly based on still or moving images and multimedia summaries are starting to be developed. From efficiency point of view, the aim of most surveyed methods has been to provide high level of efficiency.

Future advancements are expected in the areas where little or no work has been done so far. In the usage dimension, summarization methods that aim at content creation and repurposing are expected. Systems devoted to professional usage are necessary for a wide variety of professions. According to the application domain we should expect a better recognition of personal video and of the wearable video summarization problem. There is also a great need for more generic summarization methods that should be genre independent and use a generalized framework. It is also foreseeable that more research will be devoted to semantics and aesthetics based summarization methods. An area that needs to be explored is the personalization of content presented in the summary. We need a framework that captures personal preferences, creates summaries, updates the preferences and the summaries and evolves over time. MPEG-7 has included a user preference description scheme to describe user’s preferred settings in multimedia information browsing or searching, but higher-level attributes of user’s personal preferences such as tastes are not addressed.

Having witnessed the last developments in video summarization, the biggest gap we observe is the lack of a common framework for methodological, large scale benchmarking.

In the next chapter we restrict the focus of the thesis to a particular type of summaries, *video previews*, that address the *inter-selection* task.

3

User requirements

As we have seen in the previous chapter, video summarization systems can serve different purposes, can be developed for certain categories of users or for specific types of content. In this chapter we refine the scope of this thesis (Section 3.1) and we present the analysis of requirements (Sections 3.2 and 3.3). The requirements are described from the users' perspective regardless of whether they can be incorporated into an algorithm. The translations of these requirements into computable elements is discussed in Chapters 4 and 5.

3.1 Scope of the thesis

In this thesis we address the problem of helping users selecting a video content item from a large set. Our aim is to design a system that can automatically create *video previews*, efficient representations of narrative videos, to assist users in choosing among a vast collection of content items. According to the taxonomy presented in the previous chapter, we focus on the *inter-selection task*.

Our definition of video preview is: a video sequence that gives a reliable impression of a video item. A preview should be different from a commercial movie trailer. Commercial movie trailers are one of the main advertising tools of the movie industry. They are not made to give a fair impression of a film, but to convince people to watch the movie. They try to impress the audience, generate curiosity and create expectations without any guarantee that they will be met in the real content. A movie trailer maker said [Americana, 2001]: “I cannot tell you how many times we have taken a really bad movie and gotten it to open in the top four or five positions. Curiosity will get people in at least one or two more weekends and if we can do that, we have done our job!”. A video preview aims at giving a faithful impression of the content. If a video is boring, so should be its preview.

Another important difference between a commercial trailer and a video preview is that a commercial trailer is built around an advertising and marketing strategy. Voice-over comment, music and images are selected to fit the marketing strategy, regardless of the actual content or storyline of the video item. An automatic system has only the item’s content available, and can base the generation of a preview only on this content.

How do people actually choose what to watch? In a recent study of [Gutterswijk, 2004] 15 subjects were interviewed about their TV watching behavior. People tend to choose the programme by zapping or with the help of a programme guide. People find it interesting to read the film overview for movies they want to watch. Furthermore they consult programme guides during viewing time for information about the programme they watch such as information about actors and what the story is about.

Based on this study, the assumption is that a video preview should help users decide whether to watch a programme or not. Users might have a clear idea of what type of video they would like to see, for example a comedy or a horror movie. They might also know some information about the content from the programme guide. However in order to make a good choice, users would like to know more about a programme. For example, whether the programme is what they expect it to be, or whether its atmosphere fits their mood, or whether they like the type of humour, etc. This leads to the question “What elements of a video programme should a video preview contain to help a user decide what to watch?”

In the next section we try to answer this question by presenting a set of requirements that a video preview should fulfill.

3.2 Requirements analysis

Our user requirements for fast and convenient content selection are derived from related literature on video summarization (see Chapter 2 and specifically [Ma et al., 2002; Pfeiffer et al., 1996]) and film production (i.e. [Mascelli, 1965; Zettl, 2001]). Additionally, to make the list more complete, we interviewed five “experts” such as professionals active in the film and video industry and scientists of the multimedia research community.

The correctness and completeness of our set of requirements has been validated in a user study by means of guided interviews with 10 subjects [Visser, 2005]. Each subject was shown previews and commercial trailers and was invited to discuss differences. Furthermore the subjects were asked their opinion about the hypothesized requirements and whether there were characteristics that a preview should have that had not been mentioned.

Although a set of 10 subjects has a very low statistical power, the confrontation of the requirements with such a group of people is considered to be sufficient to assess that none of the major requirements were missing and that no wrong assumptions were made [Bailey, 1996].

The outcome of this elicitation process is a list of more than thirty requirements that we have grouped in seven categories: *duration*, *continuity*, *priority*, *uniqueness*, *exclusion*, *structural* and *temporal order*.

A preview can be considered as a selection of segments out of the original content. *Duration* requirements deal with the duration of the segments and the whole preview. *Continuity* requirements concern the transitions between subsequent segments. *Priority* requirements specify what type of content should be preferably included in a preview and, in a complementary role, *exclusion* requirements specify what should not be included in a preview. *Uniqueness* requirements aim at avoiding redundancy in the preview to achieve maximal efficiency. Videos have certain structural properties, such as the presence of discernible scenes, that a preview should reflect. *Structural* requirements deal with preserving in the preview the structural properties of the video item. *Temporal order* requirements dictate the order in which the video segments composing the preview should be presented.

The next sections contain a complete list of requirements for each category that a video preview for content selection should fulfill according to our requirements elicitation study.

3.2.1 Duration requirements

Duration requirements deal with the durations of the whole preview and of its segments.

Requirement D.1. *Duration of the preview*.
Users should be able to choose the desired duration of a preview according to their needs or available time. From our requirement study [Visser, 2005] the preferred duration of a video preview should be between 30 and 120 seconds. The ideal duration depends on the actual usage scenario. For selecting a movie from a large collection of content, previews should be very short. On the other hand, to make a final choice among a few options, previews can last up to a couple of minutes.

Requirement D.2. *Minimum duration of each video segment*.
Each video segment included in the preview should be understandable out of its original context. The amount of time required by an average user to comprehend it depends on its type, complexity, and generically speaking, the amount of information it conveys. For example, scenes with many details need more time to be understood by an average viewer (see e.g. [Sundaram et al., 2002]).

Requirement D.3. *Maximum duration of a video segment*.
In order not to reveal too many details of the original content, each video segment included in the preview should not be longer than a certain maximum duration. Furthermore, given a certain maximum duration for a video preview, it is preferable to show many short segments instead of a few long ones.

3.2.2 Continuity requirements

Continuity is one of the most important *film grammar* rules. Director Joseph Mascelli says in its "Five C's of Cinematography" [Mascelli, 1965]:

> "A professional sound motion picture should present a *continuous*, smooth, logical flow of visual images, supplemented by sound, depicting the filmed event in a coherent manner. [...] A picture with perfect continuity is preferred because it depicts the events realistically. A picture with faulty continuity is unacceptable, because it *distracts* rather than *attracts*." [Mascelli, 1965]

A video preview should be as continuous as possible because users do not appreciate a preview with many abrupt "jumps". In other words, the transitions between the segments composing the preview should be smooth and not distracting for the viewers.

There are at least three aspects of a video preview, that correspond to three media types, in which continuity should be preserved: visual, audio, and text.

Requirement C.1. *Visual continuity*.
The visual scenes included in each segment of a preview should not contain abrupt interruptions of action. For example, a shot showing a woman running on a beach toward the water followed by a shot of the same woman running away from the water can disorient the viewers. Although interrupting a character's action is actually a technique used often in film to increase tension, in a preview, where much of the content is left out, it does not make much sense. The risk is to distract or even annoy the viewer, rather than increase the tension.

In the audio domain, the most important aspects in a video item is usually speech. The continuity requirement for audio can therefore be specialized for speech as follows:

Requirement C.2. *Speech continuity*.
To allow viewers to correctly comprehend spoken sentences, they should not be cut but included entirely.

The textual domain for a video item is represented by its associated closed captions or subtitles. They can be seen as an additional synchronized stream of information complementary to the audiovisual streams. The continuity requirement for the textual domain can be specialized as follows:

Requirement C.3. *Subtitles continuity*.
If present, superimposed subtitles or synchronized closed captions should be displayed for a sufficient amount of time to allow viewers to read them.

3.2.3 Priority requirements

Priority requirements indicate which content should be preferably included in a preview to convey as much information as possible on the video item in the shortest amount of time.

Requirement P.1. *Fast understanding*.
A video preview will contain scenes out of their original context. To allow users to grasp the content of an entire preview, they should be able to easily and quickly understand the content of each scene included in a preview.

Requirement P.2. *People and main characters*.
Most of the stories narrated in films are centered upon people. Viewers are naturally interested in seeing the actual characters that are part of a story. Therefore, sequences including persons should be preferred for being included in a video preview.

Many users base their choice for certain content on the actual cast of a video production. It is therefore important that a video preview includes scenes with the main characters.

Requirement P.3. *Action*.
A certain number of action scenes should be included in a video preview. Scenes showing characters' activities help users to quickly understand the genre of a video (e.g. action films vs drama or comedy) and parts of a story. Action scenes have to be preferred to quiet, still scenes because the audience is able to create context from it automatically [Zettl, 2001].

Requirement P.4. *Dialogues and speech*.
Dialogues are an essential element of narrated content. Meaningful dialogues between the main characters should be included in a video preview to allow viewers to pick-up elements of the story line.

Narrative content does not only use dialogues to develop a story-line. Monologues and voice-over descriptions are also essential elements in the narrative structure. Scenes with speech should have therefore high priority for being included in a video preview.

Requirement P.5. *Silence*.
To convey as much information as possible in a short amount of time, it is better to avoid silent sequences. For example, silent scenes should have lower priority than scenes including speech.

Requirement P.6. *Highlights and emotional moments*.
A video preview should include emotional scenes that represent highlights of the story narrated in the film.

Requirement P.7. *Story clues*.
Ideally, a video preview for content selection should provide enough information on the story line to allow users to understand it, and take it into consideration in forming their opinion on the video item. Semantically important sequences should therefore be included in a preview in order to give the audience clues on the story line.

3.2.4 Uniqueness requirements

To be as efficient as possible, a preview should provide unique, non-redundant information. Uniqueness requirements promote the efficiency of a preview by penalizing redundancies in the content.

Requirement U.1. *Non-repetition*.
A video preview should not contain any repetition of a sequence of the original video item.

Requirement U.2. *Visual uniqueness*.
To maximize the efficiency of the video preview in conveying information about video content, redundancy in the visual domain should be minimized. This means that visually, the scenes included in a video preview should be as different from each other as possible.

Requirement U.3. *Story clues uniqueness*.
Clues on the story line should be presented to the users without redundancies.

Requirement U.4. *Characters uniqueness*.
To avoid redundancy, scenes showing the main characters of a video item should not be repeated.

3.2.5 Exclusion requirements

Exclusion requirements indicate which content should not be included in the preview.

Requirement E.1. *Advertisements*.
Many broadcasters introduce commercial advertisements in their video programmes. In order not to distract the users during the task of selecting a programme, and to be efficient, the video preview of a recording of a broadcast should not include any commercial advertisements.

Requirement E.2. *Non-disclosure of end*.
In order not to spoil the plot and to allow users to still enjoy the content, a video preview should not disclose the end of the story.

3.2.6 Structural requirements

Structural requirements dictate rules that pay attention to the structural properties of video.

Requirement S.1. *Style*.
A video preview should reflect the style of the video item as established, for example, by aesthetic properties such as the color scheme [Wei et al., 2004] or the editing patterns.

Requirement S.2. *Uniform coverage*.
In order to provide a good overview, a video preview should uniformly include sequences from the entire content item.

Requirement S.3. *Distance between selected segments*.
The distances between segments of the original video that are selected for the preview should be as large as possible. Short gaps between selected segments should be avoided.

Including in the preview segments that are not far enough apart from each other (*jump cuts*) can be annoying for the viewers. In other words, including segments near each other but not adjacent, can create incongruous situations. Viewers might mistakenly think that the segments are adjacent and misinterpret or even miss the logical cause for a certain character's action or spoken sentence.

Requirement S.4. *Respect scene boundaries*.
The last shot of a scene and the first shot of the consecutive scene (*establishing shot*, see e.g. [Mascelli, 1965]) should not be both included in a video preview. The reason is that these two segments, even if contiguous in the original order, are related to different scenes and therefore very often do not relate to each other.

Showing them out of their original context, one after the other in the video preview, can confuse the viewers.

Requirement S.5. *Tempo*.
A video preview should reflect the *tempo* (or *pace*) of the original video.

Requirement S.6. *Balance action and dialogue*.
Action and dialogue scenes should be included in the preview with a certain balance. The proper balance can be set depending on the video genre and the users' preferences and it should reflect the balance of the original video.

3.2.7 Temporal order requirements

Temporal order requirements concern the temporal order of the sequences included in the preview. In this category, users have indicated conflicting requirements that depend on personal taste and style. Keeping the original order certainly helps users to understand the story line given the few clues provided by the preview. On the other hand, changing the order prevents revealing too much of the story line in case users want to later view the entire content. The choice of which requirement to follow can be left to the final user of the system.

Requirement O.1. *Dialogues at the beginning*.
In order to provide users as quickly as possible with clues on the story line, dialogues sequences should be grouped together at the beginning of the preview. Many motion pictures theatrical teasers present dialogue sequences at the beginning to introduce the characters and the story line. Together with O.2, this requirement is an alternative to O.3 and O.5.

Requirement O.2. *Action at the end*.
This requirement is complementary to O.1. The information that users acquire from the dialogue sequences is used as introduction to the last part of the preview that contains the main action scenes. In many professionally made movie trailers, action scenes are packed at the end to surprise and tease the viewers. Users are used to watch commercial trailers and that is why, perhaps, they propose this ordering. Together with O.1, this requirement is an alternative to O.3 or O.5.

Requirement O.3. *Alternate action and dialogue*.
This requirement conflicts with O.1 and O.2. The alternation of dialogue sequences giving clues on the story line and action sequences is considered by many users a good generic structure for previews of narrative content.

Requirement O.4. *Preserve order within a scene*.
Sequences belonging to the same scene in the original content should have the same temporal order in the video preview. Altering the original order of sequences

that in the original content belongs to the same semantic scene, might give viewers wrong clues on the story line.

Requirement O.5. *Preserve original order*.
Maintaining the original order can reduce the chances that an uncontrolled juxtaposition of segments might provide the audience with wrong clues on the story line. According to our user requirements validation study, people seems to generally prefer the original order.

Requirement O.6. *Main characters first*.
Segments showing the main characters should be included as early as possible in the preview to allow viewers to quickly recognize them and perhaps take a decision on whether to watch the entire video, even before the preview is finished.

Requirement O.7. *Time position*.
Sequences from the end of the video should have a higher priority than sequences from the beginning. Usually the rhythm of a film increases toward the end; therefore sequences from the end of the video contain more information than sequences from the beginning. This observation is in apparent contradiction to requirement E.2 of not disclosing too much information about the end of a movie.

3.3 Overview of the requirements and priorities

Before introducing a formal description of the requirements that were informally presented in the previous sections, we prioritize them and present an overview for easier reference and analysis.

In a real system, the implementation of each requirement has a cost in terms of processing power, memory consumption and time required for computation. To allow a more flexible design, we have tried to prioritize the requirements so that the implementation can be focused first on the requirements with the highest priority.

A distinction can be made between requirements that must be fulfilled and requirements whose degree of fulfillment influences the quality of the final result without invalidating it in case of incomplete fulfillment. We assign to the former case the highest priority score, 1, while all the other cases receive scores between 2 and 4. Table 3.1 contains the results of the analysis based on the author's opinion.

Table 3.1. Requirements overview and priorities.

Requirement	**Priority**
D.1. Duration of the preview	1
D.2. Minimum duration of each video segment	1
D.3. Maximum duration of each video segment	1
C.1. Visual continuity	1
C.2. Speech continuity	1
C.3. Subtitles continuity	1
P.1. Fast understanding	2
P.2. People and main characters	2
P.3. Action	2
P.4. Dialogues and speech	2
P.5. Silence	3
P.6. Highlights and emotional moments	2
P.7. Story clues	2
U.1. Non-repetition	1
U.2. Visual uniqueness	3
U.3. Story clues uniqueness	3
U.4. Characters uniqueness	4
E.1. Advertisements	1
E.2. Non-disclosure of end	1
S.1. Style	4
S.2. Uniform coverage	1
S.3. Distance between selected segments	2
S.4. Respect scene boundaries	1
S.5. Tempo	3
S.6. Balance action and dialogue	3
O.1. Dialogues at the beginning	4
O.2. Action at the end	4
O.3. Alternate action and dialogue	4
O.4. Preserve order within a scene	4
O.5. Preserve original order	4
O.6. Main characters first	4
O.7. Time position	4

4

A formal approach to movie summarization

In this chapter we present a formal approach to the summarization of narrative video content items. The requirements and specifications informally listed in Section 3.2 are analyzed and translated into a formal definition of the problem of generating video previews.

The rest of the chapter is structured as follows. Section 4.1 provides essential definitions and the notation necessary to formalize the problem of movie summarization and to describe the proposed solution. In Section 4.2 we present the construction of the formal model based on the requirements analysis. We conclude the chapter with the formal definition of the problem of video preview generation.

4.1 Definitions and notation

In this section we give formal definitions of video item, video preview, temporal segmentation, video and audio segments, and some other related and useful concepts. Although some of these elements might be intuitively known and used in other domains, this section provides formal definitions to avoid any possible ambiguity. For ease of reference, Table 4.1 at page 54 gives a complete list of the symbols used in this chapter.

Definition 4.1 (Video item). A *video item* $\boldsymbol{V}$ is a couple $(\mathcal{V},\mathcal{A})$, where $\mathcal{V}$ a finite sequence of frames $\mathcal{V} = (f_1,\ldots,f_n)$ meant to be displayed one after the other at a specific *frame rate* R, and $\mathcal{A}$ is the associated audio track. □

Definition 4.2 (Frame rate). The *frame rate* R of a video item is the number of video frames displayed per time unit. □

A typical frame rate for standard definition television and video DVDs in Europe is 25 frames per second. According to these definitions, a video item is a sequence of frames meant to be displayed one after the other, in a predefined order at a predefined frame rate. A video item usually has associated one or more audio tracks and has often synchronized subtitles. For simplicity, we will not include multiple audio tracks and subtitles in this definition of video item but we will treat them separately.

Our definition of video item applies to most video content, such as, for example, television broadcasts, or video programmes downloaded or streamed via the Internet. Production of video items is not necessarily linear, in the sense that the frames that belong to the video item frame sequence are not always produced in the same order as they are meant to be played back. A video item is often the result of a production process that starts with content capturing by means of video cameras and usually includes an editing procedure called *montage* that involves splitting, cutting, and pasting segments in a non-linear way from multiple sources (e.g. multiple video camera takes). Before and during production, it is possible to identify a certain structure in the video content. Unfortunately this structure is completely lost after production and video is mostly delivered as an unstructured stream of frame images.

An exception is represented by some physical video carriers, such as DVD, that usually include a superimposed explicit structure used for navigating through the content stored on the device.

We will not go further into the topic of video structuring and video browsing. A lot of research work has already been published and many innovative systems have been proposed and developed. One concept that is behind most of the work done in this direction is the concept of *temporal segmentation*. We give here below a formal definition that serves the purpose of this thesis.

Definition 4.3 (Video segment). For a given video item $\boldsymbol{V} = (\mathcal{V},\mathcal{A})$, $\mathcal{V} = (f_1,\ldots,f_n)$, a *video segment* is a finite sequence of consecutive frames:

$$v = (f_i,\ldots,f_j),$$

with $1 \leq i < j \leq n$. □

Definition 4.4 (Temporal segmentation). A *temporal segmentation* of a video item, $\mathcal{S}(\boldsymbol{V})$, is a partition of a video item into non-overlapping *video segments*:

$$\mathcal{S}(\boldsymbol{V}) = \{v_1, v_2, \ldots, v_c\},$$

with $v_i \cap v_j = \emptyset$ and $\bigcup v_i = \boldsymbol{V}\ \forall i, j \in [1, \ldots, c]$. □

A *video segment* can also be seen as a time interval in the video item that has associated a *start time*, a *duration* and an *end time* defined as follows:

Definition 4.5 (Video segment start time). Each video segment $v = (f_i, \ldots, f_j)$ has a *start time*, $t_\mathrm{s}(v)$, that represents the time instant, relative to the beginning of the video item, in which its first frame starts being displayed during play back at normal speed:

$$t_\mathrm{s}(v) = \frac{i-1}{R}\ .$$

□

Definition 4.6 (Video segment duration). Each video segment $v = (f_i, \ldots, f_j)$ has a *duration*, $d(v)$, that is obtained by dividing the number of frames $(j+1-i)$ by the frame rate R:

$$d(v) = \frac{j+1-i}{R}\ .$$

□

A video segment with 250 frames and a frame rate of 25 frames per second has a duration of 10 seconds. The frame rate of a video item is usually constant, therefore, given a specific frame rate, we can also measure the duration of a video segments in number of frames. In the rest of this document, unless otherwise specified, we will measure segment durations in seconds.

Definition 4.7 (Video segment end time). Each video segment $v = (f_i, \ldots, f_j)$ has an *end time*, $t_\mathrm{e}(v)$, that represents the time instant, relative to the beginning of the video item, in which its last frame stops being displayed during normal play back. □

For a given video segment v, the following relationship is valid (see Figure 4.1):

$$t_\mathrm{e}(v) = t_\mathrm{s}(v) + d(v)\ .$$

Definition 4.8 (Video segment time span). For each video segment v, the *start time* and the *end time* define a time interval $\delta_v = [t_\mathrm{s}(v), t_\mathrm{e}(v)]$ that we call *video segment time span*. □

Similarly to the definition of *video segment* and its associated definitions of *start time*, *end time*, *duration*, and *time span*, we can consider *audio segments*.

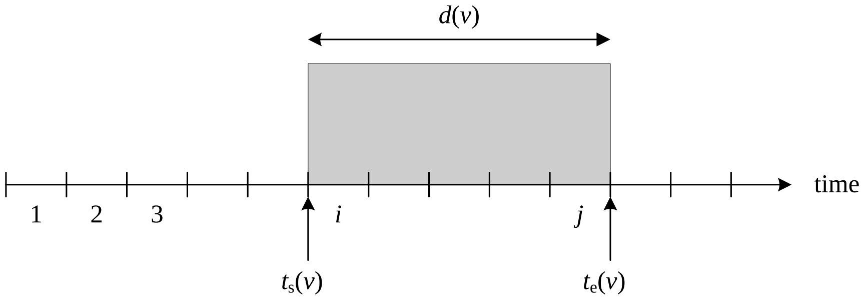


Figure 4.1. Video segment start time, end time and duration.

Definition 4.9 (Audio segment). An *audio segment* is a finite sequence of successive audio samples belonging to the audio track of a video item. □

The concept of *temporal segmentation* can be also applied to audio segments independently from video. For simplicity, unless otherwise specified, we shall consider for a video item, an audio segmentation that corresponds to the video segmentation. Each video segment v_i has a corresponding audio segment a_i with same start and end times and same duration.

Besides audio tracks, video items have often associated synchronized streams of *subtitles*. A *subtitle* is a string of text that has to be displayed at a certain time instant and for a certain duration. For subtitles we shall use the same definitions and notation given for segments. The only difference is that a temporal segmentation of a video item defines a partition into non-overlapping segments, while subtitles can be overlapping in time. It is possible that a subtitle starts to be displayed while the preceding one is still on the screen. For a brief period of time both subtitles are displayed on two or multiple separate lines.

Video items are usually the result of production procedures that involve what is called *montage* or *video editing* [Rubin, 1992]. Video is first captured in so-called *camera takes*. A *camera take* is a sequence of frames captured uninterruptedly by the same camera from the moment the camera starts capturing to the moment it stops. During *montage*, camera takes are trimmed, split, and inserted one after the other to compose an *edited version* of a video. The basic element of an edited video is called *shot*. A *shot* is a contiguous sequence of frames belonging to the same camera take in an edited video. Content-wise, shots usually possess some degree of uniformity, either visually or auditory. Based on this definition, we can define a particular type of temporal segmentation, the *shot segmentation*:

Definition 4.10 (Shot segmentation). A *shot segmentation* of a video item, $\mathcal{S}_\mathrm{h}(\boldsymbol{V})$, is a temporal segmentation in which each segment corresponds to a single shot. □

The concept of shot is derived from the physical nature of the montage procedure and, as such, it is easy to define and to understand. It is useful to define another type of segmentation that is also the result of editing but that is directly linked to the concept of narration in video. Narrative video productions are based on stories that are communicated by means of audiovisual media. Stories can be seen as sequences of independent but correlated events. Each event can potentially involve different characters, or be happening at a differ place or at a different time. In narrative, each element of a story that corresponds to such an event is called a *logical story unit*. In video production, logical story units are translated into *scenes*. A scene is a set of shots that corresponds to a logical story unit. Scenes can be formed by a sequence of consecutive shots, or by sets of temporally non-consecutive shots. Usually, although there are some exceptions, shots belonging to the same scene possess some degree of visual similarity. Based on the definition of scene, we can define another particular type of temporal segmentation, the *scene segmentation*:

Definition 4.11 (Scene segmentation). A *scene segmentation* of a video item, $\mathcal{S}_{\mathrm{c}}(\boldsymbol{V})$, is a temporal segmentation in which each segment corresponds to a single scene. □

The problem addressed in this thesis is the automatic creation of previews of narrative video items. Given the above definitions, a video preview can be formally defined as:

Definition 4.12 (Video preview). A video *preview* P is a subset of the original video item. It can be represented as a finite sequence of video segments belonging to the original video item: $P = (p_1, \ldots, p_N)$ where p_j is the j-th segment in the preview. □

The definition of video preview given above does not allow including in a preview additional content, audio or video, that is not part of the original video item but that might improve the users' experience. For example, a preview might be enhanced by mixing an additional audio track (e.g. music or voice-over commentary) or a preview could be made smoother by applying dissolve effects to segment transitions. Some of these enhancements are discussed in Section 5.9. For the moment we will stick to Definition 4.12 and not use any additional content that is not part of the original video. To summarize, Figure 4.2 provides a visual index to the definitions given so far.

4.2 Summarization as an optimization problem

In this section, the problem of automatically generating a video preview is formalized as an optimization problem. The requirements listed in the previous chapter

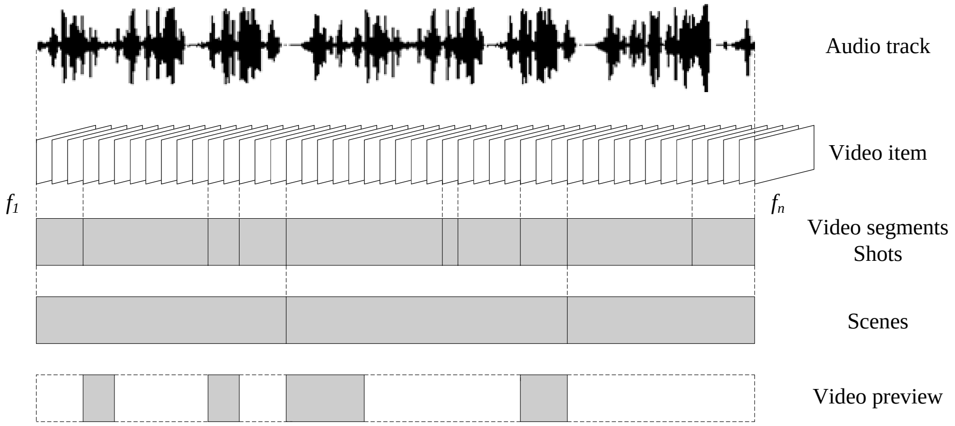


Figure 4.2. Hierarchical structure of a video item: frames, shots, scenes, and video preview. Although in the figure the segmentation and the preview seem to involve only the visual part of a video item, they include also audio.

are modeled either as formal constraints that a preview should satisfy, or as part of an objective function to maximize. The objective function is designed for estimating the quality of a preview based on numerical attributes computed directly from the content.

4.2.1 Objective function

In this section we introduce the objective function eval(P) of a given preview P as depending upon the functions: *priority score* $\pi(P)$, *redundancy score* $\rho(P)$, *structure score* $\eta(P)$, and *order score* $\omega(P)$:

$$\mathrm{eval}(P) = \mathcal{F}\left(\pi(P), \rho(P), \eta(P), \omega(P)\right) \ . \tag{4.1}$$

The *priority score* $\pi(P)$ models the priority requirements listed in Section 3.2.3. The *redundancy score* $\rho(P)$ is designed to take into consideration the uniqueness requirements defined in Section 3.2.4. The *structure score* $\eta(P)$ depends on the degree of fulfillment of the structural requirements described in Section 3.2.6. Finally, the *order score* $\omega(P)$ allows considering the temporal order requirements defined in Section 3.2.7.

In the next sections we design and make the structure of eval(P) and its components explicit. The requirements not directly included in eval(P) are modeled as explicit constraints that the segments included in P have to fulfill.

4.2.2 Modeling duration requirements

According to Requirement D.1, the duration of the preview can be assumed to be fixed by the user. If we indicate with D_{min} and D_{max} the minimum and maximum preview durations, we can model Requirement D.1 with the following constraint:

$$D_{\text{min}} \leq \sum_{j=1}^{N} d(p_j) \leq D_{\text{max}} \tag{4.2}$$

where p_j is the j-th segment in the preview $P = (p_1, \ldots, p_N)$ (see Definition 4.12).

Requirements D.2 and D.3 state that the duration of each segment in the preview should be bigger than a minimum value d_{min} and smaller than a maximum value d_{max}. This can be formalized by requiring for each segment p_j in the preview that:

$$\forall p_j \in P:\ d_{\text{min}} \leq d(p_j) \leq d_{\text{max}}\ . \tag{4.3}$$

4.2.3 Modeling continuity requirements

Shots are defined by film makers for being the elementary units constituting the video item [Block, 2001], and within a shot there are usually no abrupt interruptions of action [Mascelli, 1965]. Therefore, continuity Requirement C.1 can be fulfilled by including in the preview segments that belong to the shot segmentation of the video item:

$$\forall p \in P:\ p \in \mathcal{S}_{\text{h}}(\boldsymbol{V}) \tag{4.4}$$

where $\mathcal{S}_{\text{h}}(\boldsymbol{V})$ is a shot segmentation of the video item $\boldsymbol{V}$.

According to requirement C.2, spoken sentences in the audio track should be included entirely in the preview and not abruptly cut. To model this requirement, let us consider a temporal segmentation of the audio track $\mathcal{A}$ such that each audio segment does not contain any broken spoken sentence: $\mathcal{A} = (a_1, \ldots, a_Q)$ (a_i being the i-th audio segment and Q the number of audio segments). Let $\mathcal{A}_{\text{s}} \subseteq \mathcal{A}$ the subset of audio segments corresponding to complete spoken sentences. Requirement C.2 can be modeled by imposing:

$$\forall a \in \mathcal{A}_{\text{s}}\ ,\ \left(\delta_a \cap \bigcup_{p \in P} \delta_p\right) \neq \emptyset \Leftrightarrow \forall t \in \delta_a\ \exists q \in P : t \in \delta_q\ . \tag{4.5}$$

In words, each audio segment corresponding to a spoken sentence should be either completely included in the preview or not included at all.

Requirement C.3 states that the subtitles associated with the video item should be displayed for a sufficient amount of time to be read. If we represent the synchronized subtitles C as $C = (c_1, \ldots, c_S)$ (c_k being the k-th subtitle and S the total number of subtitles), the continuity Requirement C.3 can be formalized with the following constraint:

$$\forall c \in C \; , \; \left(\delta_c \cap \bigcup_{p \in P} \delta_p \right) \neq \emptyset \Leftrightarrow \forall t \in \delta_c \; \exists q \in P : t \in \delta_q \; . \tag{4.6}$$

In words, each subtitle overlapping with a preview segment should be completely included in the preview.

4.2.4 Modeling priority requirements

To model the set of priority requirements listed in Section 3.2.3, we define for each preview segment $p_j \in P$ a *priority score* $\pi(p_j)$ as follows:

$$\pi(p_j) = \boldsymbol{wA}(p_j)$$

where $\boldsymbol{w}$ is a vector of 7 weighting factors and $\boldsymbol{A}(p_j)$ is a column vector of numerical attributes associated with segment p_j in the range $[0,1]$:

$$\boldsymbol{A}(p_j) = \begin{pmatrix} a_1(p_j) \\ \vdots \\ a_7(p_j) \end{pmatrix} \; .$$

These 7 attributes correspond to the priority requirements P.1 till P.7 listed in Section 3.2.3 and they represent properties that can be computed directly from the audiovisual content. The higher the priority of p_j for being included in the video preview, the higher the values in $\boldsymbol{A}(p_j)$. The priority score $\pi(P)$ in the objective function (4.1) is directly proportional to the priority scores of the segments included in the preview:

$$\pi(P) = \frac{1}{N} \sum_{j=1}^{N} \pi(p_j) \; .$$

The relative importance of the various attributes can be linearly tuned using the weighting factors $\boldsymbol{w} = (w_1, \ldots, w_7)$, with:

$$\sum_{i=1}^{7} w_i = 1 \; , \; 0 \leq w_i \leq 1 \; , \; i = 1, \ldots, 7 \; .$$

In Chapter 5 we describe the exact nature of these attributes and how they can be computed directly from the raw audiovisual signal.

4.2.5 Modeling uniqueness requirements

Uniqueness requirements, defined in Section 3.2.4, promote uniqueness and penalize redundancy. Requirement U.1, *non-repetition*, can be easily modeled by imposing that all the segments included in a preview should contain different content:

$$\forall i, j\ , 1 \leq i < j \leq N\ ,\ p_i \cap p_j = \emptyset\ . \tag{4.7}$$

The *visual uniqueness* Requirement U.2 can be modeled by defining a *visual redundancy* score, $\rho_{\mathrm{v}}(P)$ for the preview that will have to be minimized:

$$\rho_{\mathrm{v}}(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \sigma_{\mathrm{v}}(p_i, p_j)$$

where $\sigma_{\mathrm{v}}(p_i, p_j)$ represents the visual similarity of segments p_i and p_j and can be computed based on automatically extracted visual features (see Chapter 5).

Requirement U.3, *story clues uniqueness* aims at eliminating redundancies in the information regarding the story line. It can be modeled by introducing an additional redundancy score, the *textual redundancy* score, $\rho_{\mathrm{t}}(P)$ that will have to be minimized for the preview:

$$\rho_{\mathrm{t}}(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \sigma_{\mathrm{t}}(p_i, p_j)$$

where $\sigma_{\mathrm{t}}(p_i, p_j)$ represents the textual similarity of segments p_i and p_j, or, in other words, the amount of redundant information provided in the textual domain.

To model Requirement U.4, *characters uniqueness*, we can follow a similar approach as for the previous requirement, and define a *character redundancy* score, $\rho_{\mathrm{c}}(P)$ that will have to be minimized and that measures how often each different character is shown in a preview:

$$\rho_{\mathrm{c}}(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \sigma_{\mathrm{c}}(p_i, p_j)$$

where $\sigma_{\mathrm{c}}(p_i, p_j)$ represents the characters similarity of segments p_i and p_j. In Chapter 5 it will be defined as the number of characters that are shown in both segments p_i and p_j of the preview. Although, in practice, it will be very difficult to compute it reliably due to the poor performances of current face recognition techniques [Turetsky and Dimitrova, 2004].

The *redundancy score* $\rho(P)$ introduced in Section 4.2.1, is defined as follows:

$$\rho(P) = r_1\rho_{\mathrm{v}}(P) + r_2\rho_{\mathrm{t}}(P) + r_3\rho_{\mathrm{c}}(P) \ ,$$

where r_1, r_2, and r_3 are normalization factors.

4.2.6 Modeling exclusion requirements

According to Requirement E.1, a video preview should not include commercial advertisements. This requirement can be modeled by forcing:

$$P \cap C_{\mathrm{a}}(\boldsymbol{V}) = \emptyset \tag{4.8}$$

where $C_{\mathrm{a}}(\boldsymbol{V})$ is a set of segments representing the commercial advertisements of video item $\boldsymbol{V}$.

Additionally, we can take into consideration the Requirement E.2 of not disclosing the end of the video item, by discarding from the set of candidate segments for the preview, a certain percentage of segments from the end of the video:

$$\forall p \in P,\ t_{\mathrm{e}}(p) \leq \alpha \sum_{p \in P} d(p) \quad \alpha \in [0,1] \ . \tag{4.9}$$

4.2.7 Modeling structural requirements

To model Requirement S.1, *style*, we define a function $\eta_1(P)$, to be maximized, that measures how visually similar the preview segments are to the original video with respect to the color distribution:

$$\eta_1(P) = \sigma_{\mathrm{v}}(P, \boldsymbol{V})$$

where $\sigma_{\mathrm{v}}(P, \boldsymbol{V})$ represents the visual similarity between P and $\boldsymbol{V}$ and can be computed based on automatically extracted visual features, as discussed in Chapter 5.

To model structural Requirement S.2 dealing with uniform coverage of the whole video item, we can consider a scene segmentation into L scenes:

$$\mathcal{S}_{\mathrm{c}}(\boldsymbol{V}) = \{U_1, U_2, \ldots, U_i, \ldots, U_L\}$$

where U_i is the set of video segments belonging to the i-th scene, $U_i \cap U_j = \emptyset$ and $\bigcup U_i = \boldsymbol{V}$, $\forall i, j \ , 1 \leq i < j \leq L$. We define another function $\eta_2(P)$ of the segments included in the preview, which has to be maximized, and measures how uniform the distribution of the preview segments is across the scenes. $\eta_2(P)$ is the product of the durations of the preview segments relative to their respective scene duration:

$$\eta_2(P) = \sqrt[L]{\prod_{j=1}^{L} \frac{c + \sum_{p \in P, p \in U_j} d(p)}{d(U_j)}} \ . \tag{4.10}$$

In the previous equation we extract the L-th root to bring η_2 to the same numerical range of the other elements of $\eta(P)$. Note that adding the constant c at the numerator of η_2 is necessary because, otherwise, if there is one scene of which no part is being used in P, then $\eta_2 = 0$.

Requirement S.3 about the *distance between selected segments*, can be modeled by defining another function to maximize, $\eta_3(P)$ as product of the distance between selected segments:

$$\eta_3(P) = \sqrt[N-1]{\frac{1}{\overline{\eta}_3} \prod_{i=1}^{N-1} |t_\mathrm{s}(p_{i+1}) - t_\mathrm{e}(p_i)|} \ , \tag{4.11}$$

where $\overline{\eta}_3$ is a normalization factor defined as:

$$\overline{\eta}_3 = \left(\frac{\sum_{v \in \mathcal{S}(\boldsymbol{V})} d(v) - \sum_{i=1}^{N} d(p_i)}{N-1} \right)^{N-1} .$$

To model Requirement S.4, *respect scene boundaries*, we impose that preview segments that are consecutive in the original video item should belong to the same scene:

$$\forall p_i, p_j \in P : \ t_\mathrm{e}(p_i) = t_\mathrm{s}(p_j) \Rightarrow \exists U_k \ : \ p_i, p_j \in U_k \ . \tag{4.12}$$

Requirement S.5, *tempo*, states that a video preview should reflect the tempo of the original video item. Directors set the film tempo during editing by adjusting the duration of the shots. Short shots induce a perception of action and fast pace. On the contrary, long shots induce a perception of calm and slow pace. Perceived film tempo is also influenced by the amount of action (actual motion) present in the video scenes and the audio loudness [Zettl, 2001]. To model this requirement, we need to consider the *tempo distribution* of a video item $\Psi_{\boldsymbol{V}}$ in order to create a preview with a tempo that is as close to the original as possible. In Chapter 5 we will show how to compute the tempo distribution of a video item.

To generate a preview that mimics the original video item's tempo we define a function to minimize, $\eta_4(P)$, that is the distance between the original video item's tempo distribution $\Psi_{\boldsymbol{V}}$ and the preview tempo distribution Ψ_P:

$$\eta_4(P) = \mathrm{dist}(\Psi_P - \Psi_{\boldsymbol{V}}) \ , \tag{4.13}$$

where $\mathrm{dist}(\Psi_P - \Psi_{\boldsymbol{V}})$ is a non-negative value that represents the distance between the original video item's tempo distribution and the preview tempo (formal definitions of Ψ_P and $\Psi_{\boldsymbol{V}}$ will be given in Chapter 5).

To model Requirement S.6, *balance action and dialogue*, we define two functions, $\mathrm{action}(p)$ and $\mathrm{dialogue}(p)$, that indicate whether a certain segment can be considered an action segment or whether it contains a dialogue:

$$\text{action}(p) = \begin{cases} 1 & \text{if } p \text{ is an action segment} \\ 0 & \text{otherwise} \end{cases}$$

$$\text{dialogue}(p) = \begin{cases} 1 & \text{if } p \text{ is a dialogue segment} \\ 0 & \text{otherwise} \end{cases}$$

A video segment is either considered an action segment or a dialogue segment:

$$\forall v \in \boldsymbol{V} : \text{action}(v) \veebar \text{dialogue}(v) \; .$$

To balance action and dialogue segments, we define an additional function to maximize, $\eta_5(P)$, that is the product of the number of action and dialogue segments in the preview:

$$\eta_5(P) = \frac{\sqrt{N_\text{a} N_\text{d}}}{N} \tag{4.14}$$

where N_a and N_d are defined as follows:

$$N_a = \sum_{p \in P} \text{action}(p)$$

$$N_d = \sum_{p \in P} \text{dialogue}(p) \; .$$

Maximizing (4.14) has the consequence that if a video item has very few dialogue segments, they might all be included in the preview, apparently violating Requirement S.5, *tempo*. In practice this situation is extremely rare because narrative video items usually have large numbers of dialogue segments.

The *structure score* function, $\eta(P)$, introduced in Section 4.2.1, can then be defined as follows:

$$\eta(P) = b_1\eta_1(P) + b_2\eta_2(P) + b_3\eta_3(P) - b_4\eta_4(P) + b_5\eta_5(P) \; , \tag{4.15}$$

where $(b_1, \ldots, b_5)$ are positive normalization factors. In Equation (4.15), $\eta_4(P)$ is subtracted from the other functions because it should be minimized in order to fulfill Requirement S.5.

4.2.8 Modeling temporal order requirements

Temporal order requirements can be taken into consideration by defining an additional function $\omega_1(P)$ to be maximized[1] that depends on the segments positions in

[1] All the functions introduced in this section have to be maximized.

the preview. In Chapter 3 we have presented three alternative requirements for ordering the segments in a video: O.1 dialogue at the beginning O.2 action at the end, and O.3 alternate action and dialogue. For each of these alternatives we give a corresponding definition of $\omega_1(P)$ that we indicate, respectively with $\omega_1'(P)$, $\omega_1''(P)$, and $\omega_1'''(P)$.

For example, to generate a preview having all the dialogue segments together at the beginning of the preview, as stated in Requirement O.1, $\omega_1'(P)$ can be defined as follows:

$$\omega_1'(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N} (N-i)\mathrm{dialogue}(p_i) \; .$$

Alternatively, to generate a preview having all the action segments at the end, as demanded by Requirement O.2, $\omega_1(P)$ can be defined as follows:

$$\omega_1''(P) = \frac{2}{N(N+1)} \sum_{i=1}^{N} i \cdot \mathrm{action}(p_i) \; .$$

To model Requirement O.3, *alternate action and dialogue*, $\omega_1'''(P)$ can be defined as follows:

$$\begin{aligned}\omega_1'''(P) = \frac{1}{2(N-1)} \sum_{i=1}^{N-1} (|\mathrm{action}(p_i) - \mathrm{action}(p_{i+1})| \; + \\ |\mathrm{dialogue}(p_i) - \mathrm{dialogue}(p_{i+1})|) \; .\end{aligned} \tag{4.16}$$

Because a video segment can be classified either as *action* or *dialogue*, in Equation (4.16), two consecutive segments of the same class do not contribute to $\omega_1'''(P)$, while consecutive segments of different classes contribute positively.

To preserve the original order of the segments included in the preview within each scene, as stated in Requirement O.4, it is sufficient to impose:

$$\forall p_i, p_{i+1} \in U_j \; , \; j = 1, \ldots, L \; , \; t_\mathrm{s}(p_i) \leq t_\mathrm{s}(p_{i+1}) \; . \tag{4.17}$$

Similarly, to maintain the original order among all the segments included in the preview, as described in Requirement O.5, we can impose:

$$\forall p_i, p_{i+1} \in P \; , \; t_\mathrm{s}(p_i) \leq t_\mathrm{s}(p_{i+1}) \; . \tag{4.18}$$

To make sure that the main characters in a video are included as early as possible in the preview, as demanded by Requirement O.6, *main characters first*, we define a function $\nu(p_i)$ that returns a positive value in the interval $[0, 1]$ directly proportional to the relative importance of the characters present in segment p_i (see

Chapter 5). To fulfill O.6, we then consider an additional function to maximize, $\omega_2(P)$, defined as follows:

$$\omega_2(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N} (N-i)\nu(p_i) \ .$$

To favor segments that appear toward the end of the video, as stated in Requirement O.7, *time position*, we can consider an additional function to maximize, $\omega_3(P)$ that is directly proportional to each video segment's start time:

$$\omega_3(P) = \sum_{p \in P} \frac{t_s(p)}{\sum_{v \in \mathcal{S}(\boldsymbol{V})} d(v)} \ . \tag{4.19}$$

Although Equation (4.19) appears to be conflicting with Requirement E.2, *non-disclosure of end*, mapped to Constraint (4.9), the two constraints can be both satisfied. A preview should not include segments at the end of the video item to fulfill Constraint (4.9), and it should include more segments near the end than segments near the beginning as result of maximizing Equation (4.19).

The *order score* $\omega(P)$ introduced in Section 4.2.1, can finally be defined as:

$$\omega(P) = c_1\omega_1(P) + c_2\omega_2(P) + c_3\omega_3(P) \ ,$$

where c_1, c_2, and c_3 are positive normalization factors.

4.2.9 Modeling the objective function

To fulfill the priority, structural, and order requirements, the choice of which segments to include in the preview should be such that $\pi(P)$, $\eta(P)$, and $\omega(P)$ are maximized while $\rho(P)$ is minimized. This can be achieved by defining $\text{eval}(P)$ as follows:

$$\text{eval}(P) = e_1\pi(P) - e_2\rho(P) + e_3\eta(P) + e_4\omega(P) \ . \tag{4.20}$$

The coefficients $(e_1, \ldots, e_4)$ are used to weigh the contributions of the different categories of requirements. They can allow personalizing the generation of the preview by changing the relative impact of the different types of requirements on the value of the objective function.

4.3 Problem definition

In view of the definitions and constraints given in the previous sections, the problem of automatically generating a preview P of a given video item $\boldsymbol{V}$ can be formalized by first defining a category of previews that satisfy all the requirements mapped explicitly to constraints.

Definition 4.13 (Set of feasible previews). Given a video item $\boldsymbol{V} = (\mathcal{V}, \mathcal{A})$, minimum and maximum durations D_{min} and D_{max}, the set of *feasible previews* $P^*(\boldsymbol{V}, D_{\text{max}}, D_{\text{min}})$, is the set of previews $P = (p_1, \ldots, p_N)$ that satisfy constraints (4.2), (4.3), (4.4), (4.5), (4.6), (4.7), (4.8), (4.9), (4.12), (4.17), and (4.18). □

Definition 4.14 (Video preview generation problem – VPG). Given a video item $\boldsymbol{V} = (\mathcal{V}, \mathcal{A})$, minimum and maximum durations D_{min} and D_{max}, find the feasible preview $P \in P^*(\boldsymbol{V}, D_{\text{max}}, D_{\text{min}})$ that maximizes $\text{eval}(P)$. □

For ease of reference, Table 4.1 gives an overview of the symbols used in this chapter to define the VPG problem.

Table 4.1. Symbols used in the definition of the VPG problem.

symbol	*description*	*page*
$\mathcal{A}$	audio track	40
$\boldsymbol{A}(p_j)$	priority attributes vector of p_j	46
$a_1(p_j),\ldots,a_7(p_j)$	priority numerical attributes of p_j	46
$\mathrm{action}(p)$	returns 1 if p is an action segment	50
a_i	i-th audio segment	42
C	set of subtitles	46
$C_{\mathrm{a}}(\boldsymbol{V})$	set of commercial advertisements $\boldsymbol{V}$	48
c_k	k-th subtitle	46
$D_{\max}$	maximum preview duration	45
$D_{\min}$	minimum preview duration	45
$d(v)$	duration of video segment v	41
$\mathrm{dialogue}(p)$	returns 1 if p is a dialogue segment	50
$d_{\max}$	maximum segment duration	45
$d_{\min}$	minimum segment duration	45
$\mathrm{eval}(P)$	objective function to maximize	44
L	number of scenes	48
N	number of preview segments	43
n	number of frames	40
N_{a}	number of action segments	50
N_{d}	number of dialogue segments	50
$P=(p_1,\ldots,p_N)$	video preview	43
p_j	j-th preview segment	43
$P^*(\boldsymbol{V},D_{\max},D_{\min})$	set of feasible previews for video item $\boldsymbol{V}$	53
Q	number of audio segments	45
R	frame rate	40
S	number of subtitles	46
$\mathcal{S}(\boldsymbol{V})$	temporal segmentation of $\boldsymbol{V}$	40
$\mathcal{S}_{\mathrm{c}}(\boldsymbol{V})$	scene segmentation of $\boldsymbol{V}$	43
$\mathcal{S}_{\mathrm{h}}(\boldsymbol{V})$	shot segmentation of $\boldsymbol{V}$	42
$t_{\mathrm{e}}(v)$	end time of video segment v	41
$t_{\mathrm{s}}(v)$	start time of video segment v	41
U_i	set of segments of scene i	48
$\boldsymbol{V}$	video item	40
$\mathcal{V}=(f_1,\ldots,f_n)$	sequence of frames of video item $\boldsymbol{V}$	40
v_i	i-th video segment	40
$\boldsymbol{w}$	priority score weighting vector	46

symbol	*description*	*page*
α	discarded fraction of the video item	48
δ_v	time span of video segment v	41
$\eta(P)$	order score of preview P	44
$\nu(p_i)$	relative characters importance of segment p_i	51
$\pi(P)$	priority score of preview P	44
$\pi(p_j)$	priority score of segment p_j	46
$\rho(P)$	redundancy score of preview P	44
$\rho_{\mathrm{c}}(P)$	character redundancy score	47
$\rho_{\mathrm{t}}(P)$	textual redundancy score	47
$\rho_{\mathrm{v}}(P)$	visual redundancy score	47
$\sigma_{\mathrm{c}}(p_i, p_j)$	characters similarity of segments p_i and p_j	47
$\sigma_{\mathrm{t}}(p_i, p_j)$	textual similarity of segments p_i and p_j	47
$\sigma_{\mathrm{v}}(P, \boldsymbol{V})$	visual similarity between $\boldsymbol{V}$ and preview P	48
$\sigma_{\mathrm{v}}(p_i, p_j)$	visual similarity of segments p_i and p_j	47
Ψ_P	tempo distribution of preview P	49
$\Psi_{\boldsymbol{V}}$	tempo distribution of video item $\boldsymbol{V}$	49
$\omega(P)$	structure score of preview P	44

5

Solution approach

After having elicited the requirements that a preview should fulfill in Chapter 3, we have formalized a solution approach in Chapter 4. In this chapter we further specify, specialize and describe the implementation of the elements, constraints and functions that appeared in the formal model of Chapter 4 in an abstract or generic way. Finally we present a solution based on local search and its numerical evaluation.

The rest of this chapter is structured as follows: Section 5.1 provides an overview of our solution approach that is further discussed in Sections 5.2, 5.3, 5.4, 5.5, 5.6, and 5.7. A numerical evaluation of the quality of the generated solutions is reported in Section 5.8, while Section 5.9 describes a last post processing step. Conclusions are presented in Section 5.10.

5.1 Overview

Our approach to solve the video preview generation problem consists of two main steps: a *preparation* step and a *selection* step. Figure 5.1 shows a schematic representation of our approach. In the *preparation* step, the raw audiovisual content is divided into segments that are suitable for being included in a preview. In this step we aim at solving as many top-priority requirements as possible. In the *selection* step an optimal subset of the segments is selected that fulfills as many requirements as possible. Finally, a preview is composed using these segments.

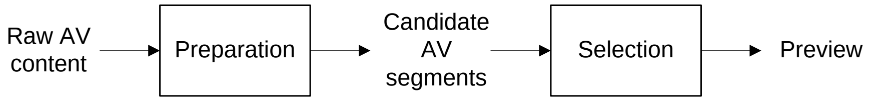


Figure 5.1. Schematic representation of the two main steps of our approach.

The *preparation* main step can be further divided into three intermediate steps: *temporal segmentation*, *micro segmentation*, and *segment compensation*. Figure 5.2 shows all the steps included in our approach.

The *temporal segmentation* step partitions the video frames into visually contiguous segments that are used as basic units in constructing the preview. Because the segmentation is based on shot-cut detection, after this stage there is no guarantee that the duration requirements are fulfilled. The *micro segmentation* step takes care of further splitting segments that are too long for being entirely included in a preview. Finally, the *segment compensation* step repairs the segments that violate one of the continuity requirements.

After these three initial steps we obtain a set of candidate segments suitable for being included in a preview. The *selection* main step is responsible for selecting which segments to include in the preview. It can be further divided into four successive steps: *macro segmentation*, *pre-filtering*, *optimization*, and *post processing*.

The *macro segmentation* step performs a scene segmentation as required by the structural requirements. Before selecting a set of segments that maximizes $\mathrm{eval}(P)$ in the *optimization* step, a *pre-filtering* operation is carried out that eliminates all segments that do not fulfill the exclusion requirements. The last step, the *post processing*, provides the final glue to bind all the selected segments in a visually pleasing video preview.

In the next sections we describe the details of all the steps using the formalism introduced in the previous chapter. Additionally, we provide insights on the implementation of the proposed method for the case of MPEG video. MPEG is the video compression standard used in digital television broadcasts and commercial distribution of videos (DVDs). It has been designed to optimize compression rates and mostly for sequential access. In the following sections we present a fast and efficient implementation of video preview generation for MPEG that does not require full decoding of the video.

5.2 Temporal segmentation

The first step toward generating a video preview consists of segmenting the unstructured video item into meaningful segments. Constraint (4.4) derived from

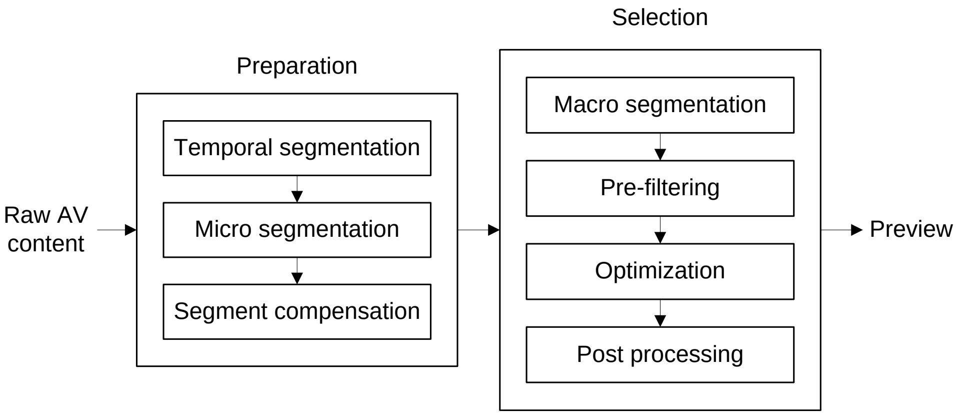


Figure 5.2. Schematic representation of all the steps of our approach.

Requirement C.1 (*visual continuity*) states that the temporal segmentation forming the basis of the preview should be a shot segmentation.

Many methods for shot segmentation have been proposed in the literature with different computational complexity and different performances in terms of precision and recall [Lienhart, 1999]. Because the results of shot segmentation are further refined in successive steps before being used in the creation of a preview, we have implemented a simple histogram-based method based on comparing color histograms of consecutive frames. When the difference between the color histograms exceeds a certain threshold, the beginning of a new shot, a so-called *shot-cut*, is assumed. The threshold is proportional to the frame dimensions or can be dynamically adjusted depending on the average of the histogram difference in a sliding window of a certain number of frames. Although many difference measures have been proposed for comparing color histograms (see for an overview [Lienhart, 1999] or [Barbieri, 2001]), for the purpose of detecting shot-cuts, we found that the simple L_1 distance is in practice good enough. Given two color histograms $H(f_i)$ and $H(f_{i+1})$, of two consecutive frames f_i and f_{i+1}, the L_1 distance is defined as:

$$L_1\left(H(f_i),H(f_{i+1})\right) = \sum_{k=1}^{B} |H_k(f_i) - H_k(f_{i+1})| \ ,$$

where $H_k(f_i)$ represents the k-th bin of the color histogram of frame f_i. A shot-cut is detected each time $L_1\left(H(f_i),H(f_{i+1})\right)$ exceeds a fixed threshold.

In case of video material compressed in MPEG format, considerable improvements in speed and memory usage can be achieved by considering only I-frames (e.g. for a typical GOP size of 12, at a frame rate of 25 frames per second, this corresponds to sub-sampling the video at approximately 2 frames per second). Additionally, to further increase speed and reduce memory requirements, we extract from the video stream, with only partial decoding, rescaled versions (64 times smaller) of the I-frames called DC-images. They are obtained by considering only the DC coefficients of the bi-dimensional discrete cosine transform of the 8x8 pixels blocks of a full-size frame (see also [Barbieri, 2001]). The performances of shot-cut detection using the color histogram of the DC-images are the same as using full-size frames with the considerable advantage of requiring much less computational resources.

Figure 5.3 shows a graph of the L_1 histogram differences between consecutive I-frames of a video segment of 48 seconds. The fixed threshold and the detected shot-cuts are also shown.

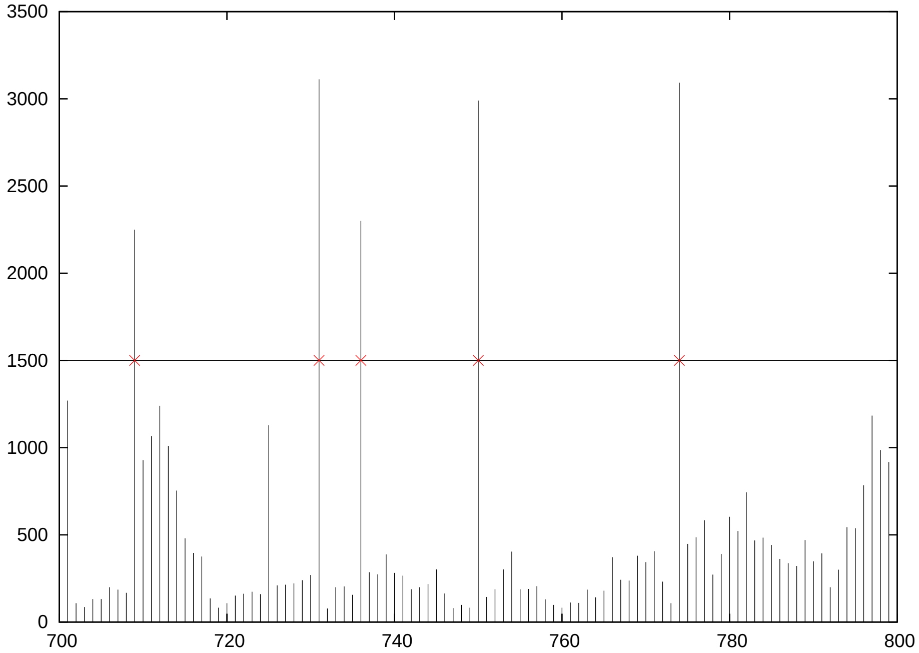


Figure 5.3. L_1 histogram difference between consecutive I-frames of a 48 seconds long video segment. The threshold is fixed at 1500.

Table 5.1 reports the performances of the shot-cut detector on a test set of 5 video programmes belonging to the TRECVID 2001 dataset [Nist, 2007]. The total duration of the videos is about 2 hours. The average recall (percentage of shot-cuts detected) is 90% with an average precision (percentage of correctly detected shot-cuts) of 92%.

The temporal segmentation procedure delivers a list of shot boundaries that

Table 5.1. Precision and recall of the shot-cut detector for 5 videos from TRECVID 2001. The *GT* (*ground truth*) column represents the number of shot-cuts manually annotated. The column *Detected* represents the number of detected shot-cuts and the column *Correct* represents the number of shot-cuts correctly detected. *Precision* is the ratio between the number of shot-cuts correctly detected and the number of shot-cuts detected. *Recall* is the ratio between the number of shot-cuts correctly detected and the number of shot-cuts manually annotated. The average precision is 92% and the average recall is 90%.

Video	**GT**	**Detected**	**Correct**	**Precision**	**Recall**
NASA 25th Anniv. Seg. 5	65	62	57	91.9%	87.7%
NASA 25th Anniv. Seg. 9	103	106	91	85.8%	88.3%
Challenge at Glen Canyon	242	271	227	83.8%	93.8%
The Great Web of Water	531	476	446	93.7%	84.0%
Senses and Sensitivity 3	308	300	294	98.0%	95.5%

define the segments that will be used as elementary units for building the preview.

The shot segmentation procedure does not guarantee that each shot has the proper duration. There could be shots with a duration that exceeds the maximum duration requirements, or shots could have been detected that are too short for being included in the preview out of their context. These two issues are solved using respectively a *micro segmentation* and a *segment compensation* procedure. These procedures are described in the next two sections.

5.3 Micro segmentation

Requirement D.3 (*maximum shot duration*), translated into the right side of constraint (4.3), states that each segment should not be longer than a certain duration. In this step, segments exceeding the maximum duration after the shot segmentation are further divided into sub-segments using a micro segmentation procedure.

For each segment v for which its duration $d(v)$ is greater than $d_{\max}$, the procedure generates a subdivision into sub-segments with durations not greater than $d_{\max}$ and not smaller than $d_{\min}$. If an exact subdivision with these constraints does not exist, then the remaining parts of the segment that are shorter than $d_{\min}$ can be aggregated with adjacent segments or simply discarded in the further processing. The micro-segmentation step can be easily formalized as an integer linear programming problem and solved with standard methods (e.g. simplex method). The problem is an instance of the *bin packing problem*, a *knapsack* variant [Martello and Toth, 1990]. It can be formalized as follows:

$$\max \sum_{i=1}^{N} x_i$$

$$\sum_{i=1}^{N} x_i \leq L$$

$$d_{\min} \leq x_i \leq d_{\max}\ ,\ i = 1,\ldots,N$$

where L is the video segment's total duration, N is the number of sub-segments, given by $N = \lceil L/d_{\max} \rceil$, and x_i is the duration of the i-th sub-segment with $i = 1,\ldots,N$.

Additionally, the micro-segmentation procedure can be guided by clues extracted from the content such as:

- A change in audio category or sentence boundary (audio clue).
- Appearance and disappearance of subtitles (visual clue).
- Appearance and disappearance of detected faces (visual clue).
- A change in camera motion or object motion (motion clue).

Section 5.7.1 describes how these clues can be obtained from the audiovisual content by means of content analysis.

Note that visual changes in content are not listed among the content clues. The reason is that the segments to split result from the shot segmentation step and are therefore visually uniform.

5.4 Segment compensation

The segment compensation step addresses the *minimum duration* Requirement D.2 and the continuity requirements C.2 (*speech continuity*) and C.3 (*subtitles continuity*). The idea is to fulfill these requirements in an early stage, before optimization of the objective function, so that the optimization procedure can be simpler and more efficient.

Although shots are defined by film makers as elementary units, some of them are not understandable if taken alone, because they are meant to be displayed in their natural sequence. These shots are characterized by a too short duration that violates Requirement D.2 (*minimum duration*), formalized in the left side of Constraint (4.3). This requirement can be simply fulfilled by discarding the shots that are too short. Alternatively, a shot that is too short can be merged with its predecessors or successors (forming a *shot sentence*) until the minimum duration is reached and within the limit of $d_{\max}$.

The requirements C.2 (*speech continuity*) and C.3 (*subtitles continuity*), modeled respectively with constraints (4.5) and (4.6) can be very restrictive and they could eliminate too many candidate segments. In the segment compensation step, we aim at reducing the segments with broken sentences by merging them with subsequent or preceding segments.

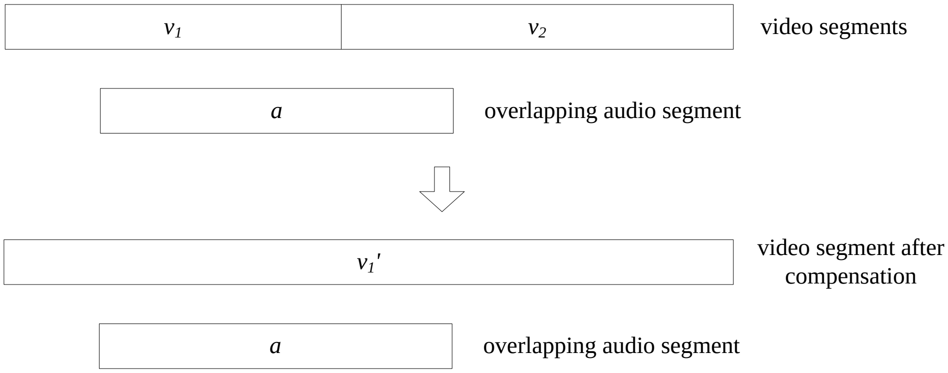

v_1
v_2
video segments
a
overlapping audio segment
v_1'
video segment after compensation
a
overlapping audio segment

Figure 5.4. Two consecutive video segments can be merged to avoid breaking a speech segment.

For each shot in which a spoken sentence or subtitle starts but not ends, the subsequent shots are considered. If the subsequent shots together with the original one allow fulfilling C.2 and C.3 and do not violate D.3 (*maximum segment duration*) then they are merged. If not, they are discarded. However, discarding segments for this reason might lead to too few candidate segments for the preview. To prevent this, it is possible to keep as candidates the segments that violate requirements (*speech continuity*) or C.3 (*subtitles continuity*), and add a penalty term to the objective function for each speech segment which is not completely included in the preview (see Section 5.7).

Another option to use in case too many segment boundaries overlap with spoken sentences or subtitles, is to abandon at this stage the shot segmentation and adopt another temporal segmentation based on speech and subtitles. Segments representing uninterrupted speech or subtitles can be used as starting candidates for the video preview. In this case, the segmentation compensation step will check which audio segments overlap with shot boundaries, and will re-segment the video to match the audio segments. This video re-segmentation is carried out only if the resulting video segments fulfill the duration requirements D.2 (*minimum segment duration*) and D.3 (*maximum segment duration*) as stated by Constraint (4.3).

Yet another possibility would be to relax the constraint that video must be associated with the original audio. Audio could be more freely inserted in some cases, like e.g. music, and more tightly synchronized in other cases, for example a close-up of an actor speaking requires clear lip-synchronized audio. Because of the inherent complexity of achieving such a high level of content understanding, we decided to investigate this as part of future work.

To justify the need for the segment compensation step, a measurement on seven feature films and one documentary has shown that approximately 30% of the video segments obtained after shot segmentation and micro segmentation are associated with subtitles with an overlapping duration of less than 1 second. This means that 30% of the shots, if selected without their surrounding segments, will have an associated subtitle that is not displayed long enough to be read by a viewer. Of course, if the rendering of the subtitles can be controlled (e.g. in case of closed captions, teletext, DVDs), then subtitles shorter than 1 second can just be skipped while rendering the preview.

5.5 Macro segmentation

To ensure a uniform coverage of the whole video item (Requirement S.2) a *macro-segmentation* can be performed to divide the video content into logically or visually consistent units. The next steps of the video preview generation procedure can take care of uniformly including segments from each scene.

In this step we apply known methods proposed in literature. Many algorithms have been published for *logical story unit* detection or *scene boundary* detection. We have tested and implemented various shot clustering techniques [Karoutchi, 2003] and found the *time-constrained clustering* [Boreczky et al., 2000] the most effective for our purposes.

A clustering procedure is applied to cluster shots that are visually similar with the assumption that visually similar shots belong to the same scene. At the end of the procedure, each cluster corresponds to a scene. It is necessary however to take time into consideration because shots that are visually similar but far apart in time usually belong to different scenes.

Time-constrained shot clustering imposes a time-window parameter T_w that prevents two shots that are far apart in time to be clustered together because they can potentially represent different contents or occur at different scenes, even if they are visually similar. The clustering algorithm (e.g. K-means) considers only shots that satisfy the time constraint. The addition of temporal constraints causes minimal changes to the clustering algorithm but significantly reduces the computational complexity.

The result of the macro segmentation procedure is a list of scene boundaries that is used in the next steps.

5.6 Pre-filtering

Requirement P.5, *avoid silences* can be easily fulfilled by removing from the set of segments generated in the previous steps the ones that correspond to silence. Alternatively, P.5 can be taken into consideration in the priority score used for the local search (see Section 5.7.1).

Requirement E.1 (*exclude advertisements*) can be easily fulfilled by detecting advertisements and removing them from the list of candidate segments to form the preview.

To automatically detect advertisements in the video items we apply an algorithm similar to the one described in [Dimitrova et al., 2001; Dimitrova et al., 2002; Schaffer et al., 2002].

In the visual domain we detect monochrome frames (e.g. black frames) and changes in image format (e.g. from wide-screen to 4:3) in the following way. We represent the luminance component of a video frame with a bi-dimensional array $Y(x,y)$ and we compute the variation of the luminance along rows, γ, as follows:

$$\gamma = \frac{1}{h(w-1)} \sum_{j=1}^{h} \sum_{i=1}^{w-1} |Y(i,j) - Y(i+1,j)| \ ,$$

where w and h are the width and height of the frame respectively.

Each time γ is below a certain threshold (fixed heuristically to 0.004), we detect a monochrome frame.

The computation of the luminance variation can be restricted to the top and bottom parts of a frame to detect the presence of monochromatic bars that indicate that a wide-screen frame has been resized to fit a different aspect ratio (4:3):

$$\begin{aligned} \gamma_{\mathrm{lb}} \quad = \quad & \frac{1}{\alpha h(w-1)} \sum_{j=1}^{\alpha h} \sum_{i=1}^{w-1} |Y(i,j) - Y(i+1,j)| + \\ & \frac{1}{\alpha h(w-1)} \sum_{j=(1-\alpha)h}^{h} \sum_{i=1}^{w-1} |Y(i,j) - Y(i+1,j)| \ , \end{aligned}$$

where α is a fraction of the frame's height and it is typically set to 0.007.

Every time γ_{lb} is below a threshold, we detect the presence of the so-called *letterbox*. A change in image format is detected each time the *letterbox* appears or disappears. Figure 5.5 shows some examples of frames with letterbox.

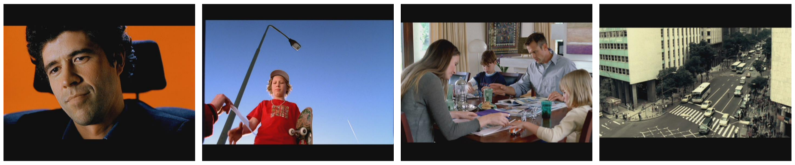


Figure 5.5. Examples of video frames with letterbox.

In the audio domain we detect silences each time the audio energy measured as running average over a 120 milliseconds window drops below a certain threshold.

Monochrome frames, changes in image format and silences are called *separa-*

tors because they occur at the boundary between video programmes and advertisements or between consecutive advertisements.

Visual and audio clues are used to trigger a finite state machine that judges, based on a set of rules (e.g. advertisements usually appear in blocks), whether a video segment is an advertisement. In short, every segment delimited by separators and shorter than a maximum length D is marked as advertisement. In practice the value of D is set to 200 seconds to tolerate missing separators. Figure 5.6 shows the pseudo-code of the advertisement detection algorithm.

The method can achieve high recall rates (e.g. above 90%) and good precision with European and American broadcast material. The 10% missed commercial blocks are usually located at the programme boundaries where broadcasters do not always insert separators.

Table 5.2 reports the performances of the advertisement detector on a test set of 11 video programmes recorded from various European channels. The total duration of the recordings is about 30 hours. The average recall (percentage of advertisements detected) is 95% with an average precision (percentage of correctly detected advertisements) of 75%.

Table 5.2. Precision and recall of the advertisement detector for 11 video programmes recorded from various European broadcasters. The *GT* (*ground truth*) column represents the number of I-frames manually annotated as being advertisements. The column *Detected* represents the number of I-frames detected as being advertisements and the column *Correct* represents the number of I-frames correctly detected as advertisements. *Precision* is the ratio between the number of I-frames correctly detected as advertisements and the number of I-frames detected as advertisements. *Recall* is the ratio between the number of I-frames correctly detected as advertisements and the number of I-frames manually annotated as advertisements. The average precision is 75% and the average recall is 95%.

Recording	**GT**	**Detected**	**Correct**	**Precision**	**Recall**
A Space Odyssey	1500	1844	1465	79%	98%
Dog Day Afternoon	1046	1103	986	89%	94%
Far and Away	1066	1429	1050	74%	99%
Presumed Innocent	1906	2640	1853	70%	97%
Red October	1210	2112	1146	54%	95%
Showgirls	1471	1433	1338	93%	91%
South Park	268	789	265	34%	99%
Star Wars Episode I	1489	1820	1396	77%	94%
The McKenzie Break	294	566	280	50%	95%
The Peacemaker	1622	1543	1500	97%	93%
The Scarlet Letter	1213	1375	1192	87%	98%

```
algorithm ADVERTISEMENT DETECTION
Given video item $\boldsymbol{V} = (\mathcal{V}, \mathcal{A})$, $\mathcal{V} = (f_1, \ldots, f_n)$, frame rate $R$,
maximum advertisement block duration $D$;
begin
    Initialize set of advertisements of video item $\boldsymbol{V}$: $C_{\mathrm{a}}(\boldsymbol{V}) := \emptyset$;
    Initialize state variable: $\sigma := 0$;
    for $i : 1$ to $n$ do
    begin
        $s := i$;
        if $(\sigma == 0)$ then
            if separator$(f_i)$ then
                Potential start of advertisement detected at frame $f_i$
                Switch to state 1: potential start: $\sigma := 1$;
            end
        else if $(\sigma == 1)$ then
            if (separator$(f_i) \wedge (i - s)R < D$) then
                Potential end of advertisement detected at frame $f_i$
                $e := i$;
                Switch to state 2: advertisement detected: $\sigma := 2$;
            else if $(i - s)R \geq D$ then
                Switch to state 0: no advertisement: $\sigma := 0$;
            end
        else if $(\sigma == 2)$ then
            if (separator$(f_i) \wedge (i - s)R < D$) then
                Potential end of advertisement detected at frame $f_i$
                $e := i$;
            else if $(i - s)R \geq D$ then
                $C_{\mathrm{a}}(\boldsymbol{V}) := C_{\mathrm{a}}(\boldsymbol{V}) \cup (f_s, f_n)$;
                Switch to state 0: no advertisement: $\sigma := 0$;
            end
        end
    end
    if $(\sigma == 1) \vee (\sigma == 2)$ then
        $C_{\mathrm{a}}(\boldsymbol{V}) := C_{\mathrm{a}}(\boldsymbol{V}) \cup (f_s, f_n)$;
    end
end
```

Figure 5.6. Algorithm for advertisement detection.

To further ensure that the video preview will not include the start or end of commercial blocks, we can discard extra segments at each detected commercial block boundary in the pre-filtering step. For example, removing 3 minutes of video programme before and after each detected beginning and end of a commercial block can increase the recall rate up to 100% at the expense of reducing precision to 65%. Because our aim is to avoid including commercial advertisements in previews, we can accept this trade-off between high recall and relatively low precision.

Requirement E.2 (*non-disclosure of end*), can be fulfilled by discarding a fraction of the segments at the end of the video item. A statistically sound percentage can be found by identifying for a large set of programmes at which point the end is disclosed (5% in our experiments).

5.7 Optimization

At this stage, most of the top-priority requirements listed in Chapter 3 are already fulfilled. We have available a set of segments that are suitable candidates for being included in the preview. The optimization procedure selects the set of segments that best fulfill the remaining requirements and constraints by maximizing Equation (4.20) representing the objective function $\mathrm{eval}(P)$ defined in Section 4.2.9.

Before presenting the procedure to maximize $\mathrm{eval}(P)$, we describe the structure and the elements that compose the four functions that contribute to $\mathrm{eval}(P)$: *priority score*, *redundancy score, structure score,* and *order score*.

5.7.1 The priority score

The priority score of preview P is defined as the sum of the priority scores of the segments included in the preview (see Section 4.2.4). Each segment p_j has associated a vector $\boldsymbol{A}(p_j)$ of 7 numerical attributes, called *priority attributes*, that are computed from the audiovisual content and correspond to the 7 priority requirements presented in Section 3.2.3. In the next sections we describe how these numerical attributes are computed and how they relate to the priority requirements.

Requirement P.1: fast understanding

Dark scenes and images with low sharpness are difficult to comprehend out of their original context and require more time to be understood by the viewers. On the contrary, bright and sharp scenes can be understood more quickly by viewers and should therefore be preferred for being included in a video preview.

The first priority attribute, related to *fast understanding*, is defined as a linear combination of the *sharpness* and the *brightness* of a segment.

The *sharpness* of a video frame is measured by counting the number of sharp edges. Assuming that each video frame is defined by three bi-dimensional arrays, the luminance (Y) and chrominance $(C_\mathrm{r}, C_\mathrm{b})$ components, each frame f can be written as:

$$f = [Y(x,y)C_\mathrm{r}(x,y)C_\mathrm{b}(x,y)] \ ,$$

where x and y are the column and row index respectively. Because chrominance components can be sub-sampled (in MPEG is mostly 4:2:0 or 4:2:2 formats), luminance and chrominance are treated separately in the extraction algorithm.

For each pixel of the frame, a spatial discrete derivative matrix is computed: $[\nabla Y(x,y) \nabla C_\mathrm{r}(x,y) \nabla C_\mathrm{b}(x,y)]$. A sharp edge is locally detected along the luminance component whenever the norm of the spatial gradient is above a certain threshold θ_Y. The local measure of sharpness for the luminance component is given by:

$$S_\mathrm{Y}(x,y) = \begin{cases} 1 & \text{if } \|\nabla Y(x,y)\| > \theta_\mathrm{Y} \\ 0 & \text{otherwise.} \end{cases}$$

On the other hand, a sharp color edge is detected if the sum of the norms of the spatial gradients along the chrominance components is above a certain threshold $\theta_{\mathrm{C_h}}$. The local measure of sharpness for the chrominance components is then given by:

$$S_\mathrm{Ch}(x,y) = \begin{cases} 1 & \text{if } \|\nabla C_\mathrm{r}(x,y)\| + \|\nabla C_\mathrm{b}(x,y)\| > \theta_{\mathrm{C_h}} \\ 0 & \text{otherwise.} \end{cases}$$

In the previous equations we assume that the norm of $\nabla F(x,y)$ is computed as:

$$\|\nabla F(x,y)\| = |F(x,y) - F(x+1,y)| + |F(x,y) - F(x,y+1)| \ ,$$

where F being Y, C_r, or C_b.

The sharpness S_f of a video frame f is defined as a weighted sum of the number of pixels that involve edges in the luminance and chrominance components:

$$S_f = \sum_{x,y} S_\mathrm{Y}(x,y) + A \sum_{x,y} S_\mathrm{Ch}(x,y) \ .$$

The factor A consists of two factors, $A = A_1 A_2$. The first factor, A_1, is determined by the consideration that the perception of sharpness is more related to luminance than chrominance, so $A_1 = 0.5$. The second factor, A_2 , depends on the chroma format. For 4:2:0 format C_r and C_b are horizontally and vertically sub-sampled with respect to luminance so that the number of pixels for the luminance component is 4 times bigger. Therefore $A = 4 \times 0.5 = 2$. Analogous weights can be defined for the 4:2:2 and 4:4:4 formats.

Figure 5.7 shows some examples of video frames with high and low sharpness values.

The sharpness of the j-th preview segment $p_j = (f_1, \ldots, f_N)$ is defined as the average of the sharpness of its video frames:

$$S(p_j) = \frac{1}{N}\sum_{i=1}^{N} S_{f_i} \ .$$

To obtain a priority attribute in the range [0,1], the sharpness of each segment is normalized with respect to the maximum sharpness value in the whole video item.

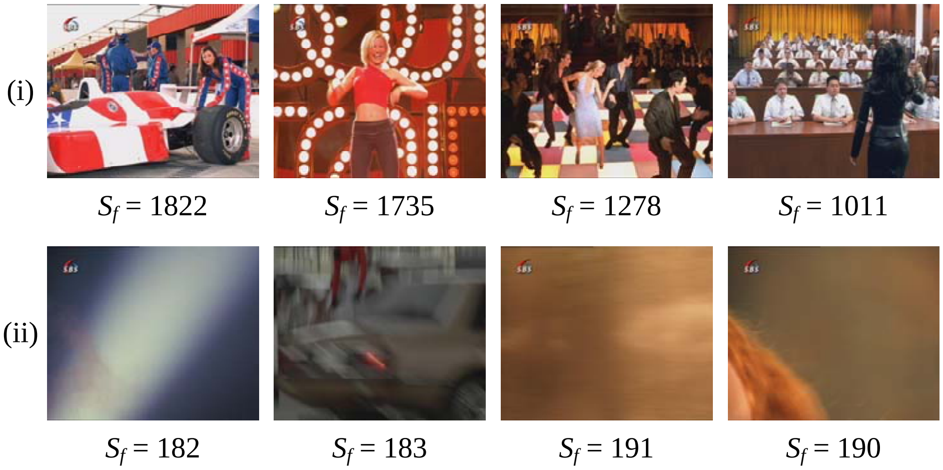


Figure 5.7. Examples of video frames with high (i) and low (ii) sharpness values.

Similarly to sharpness, the overall brightness of a scene influences the time viewers need to comprehend it. Viewers can more easily and more quickly understand a bright scene instead of a dark scene [Matlin and Foley, 1997; Pfeiffer et al., 1996], therefore bright scenes should be preferred to dark scenes to fulfill the *fast understanding* requirement. Figure 5.8 shows some examples of high and low brightness video frames.

We define the overall brightness B_f of a video frame f as the sum of its luminance values across the frame normalized with respect to the maximum luminance value (e.g. $Y_{\max} = 255$):

$$B_f = \frac{\sum_{x,y} Y(x,y)}{Y_{\max}} \ .$$

The total brightness of the j-th preview segment $p_j = (f_1, \ldots, f_N)$ is the average of the overall brightness of its frames:

$$B(p_j) = \frac{1}{N}\sum_{i=1}^{N} B_{f_i} \ .$$

Finally, the first priority attribute for a the j-th preview segment p_j is defined as:

$$a_1(p_j) = \frac{S(p_j) + B(p_j)}{2} \; .$$

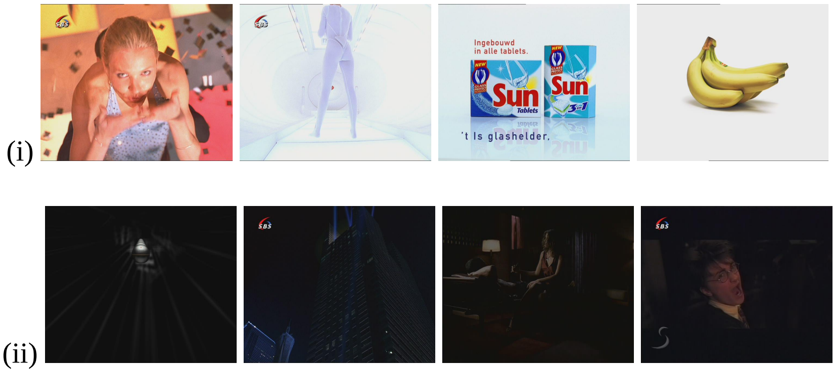


Figure 5.8. Examples of high (i) and low (ii) brightness video frames.

Requirement P.2: people and main characters

Segments representing people should have higher priority according to Requirement P.2, *people* (see Section 3.2.3). To fulfill this requirement we define the second priority attribute to be proportional to the amount of time a person is shown during a video segment and to the actual number of persons shown.

To this extent we apply a face detector to each video frame and we track the detected faces across all the frames of a video segment. We have implemented a skin-color-based face detector inspired by [Abdel-Mottaleb and Elgammal, 1999] and tested also an implementation of the algorithm described in [Viola and Jones, 2001]. The skin-based face detector tends to recognize many more faces than the Viola-Jones detector, but it has a rather high false detection rate and it can miss faces that appear in front of skin-colored backgrounds. The Viola-Jones face detector is much more precise but it fails to detect many faces especially if they are not frontally looking at the camera. The performances of the face detector are however overall satisfactory for prioritizing segments that include persons. Figure 5.9 shows three examples of detected faces in video frames.

Each detected face in a video frame contributes to the priority score depending on its size and position. The bigger the face, the higher its importance. We define the *face score* $\Phi(f)$ for a video frame f as:

$$\Phi(f) = \frac{1}{M_f} \sum_{k=1}^{M_f} \frac{\mathrm{area}(F_k)}{\mathrm{area}(f)} \cdot \frac{w_{P_k}}{8} \; ,$$

where M_f is the number of faces detected in frame f, $\mathrm{area}(F_k)$ is the area of the

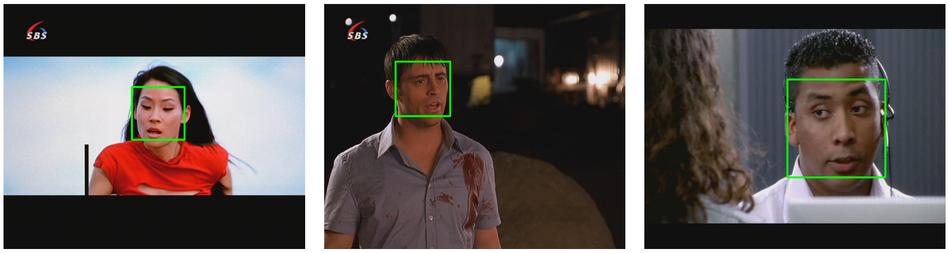
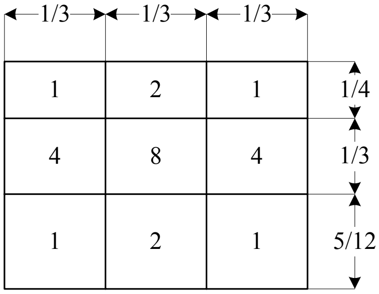


Figure 5.9. Examples of detected faces.

k-th detected face, area(f) is the number of pixels of the video frame, and w_k is the weight for the position of the center of the k-th face according to Figure 5.10. Faces shown in the middle of the frame are weighted higher than faces appearing near the frame's borders [Ma et al., 2002].



1	2	1
4	8	4
1	2	1

Figure 5.10. Position weights applied to detected faces.

The second priority attribute of a segment, $a_2(p_j)$, is defined to be proportional to the amount of time a face appears in a segment:

$$a_2(p_j) = \frac{1}{N}\sum_{i=1}^{N}\Phi(f_i) \ .$$

A straightforward approach for ensuring that the main characters of a given video item are included in a preview, would be to recognize which faces belong to the main actors. Unfortunately, face recognition technology is not sufficiently mature to allow reliable automatic person identification in generic video (e.g. TV and film content). Furthermore traditional person identification systems require “a priori” knowledge of the persons present in the video. For each person, a face model has to be computed and stored in a database to allow for recognition. For generic video (e.g. TV content or movies) creating and maintaining the database is a very expensive and difficult task.

Instead of trying to recognize the persons in each video frame, we propose to use clusters of face features to assess the relative importance of the persons without necessarily recognizing them. This approach is applicable without the need for any “a priori” knowledge (e.g. database of persons’ face models). Instead of trying to classify each detected face using a database of known persons, the face features are clustered into groups of similar ones. Faces belonging to the same person will have similar features and will be clustered together. For the clustering, known algorithms such as K-means [Duda et al., 2002], GLA (Generalized-Lloyd Algorithm) [Gersho A., 1992] or SOM (Self Organizing Maps) [Kohonen, 1997; Haykin, 1999; Duda et al., 2002] can be used.

After processing all the video content, the largest clusters correspond to the most important persons in the video. The video summarization system can then preferably include in the summary segments with faces that belong to the main clusters. Even without an “a priori” knowledge of the persons present in the video, the system will include the most important ones.

Usually, the lead actor/actress are given a lot of screen time and are present throughout the duration of the movie. Also, since, they are important to the movie, we get to see their close-up shots much more often than those of any other supporting characters who might appear only in a few scenes in the movie. Therefore even non-perfect face recognition technology should allow prioritizing faces according to their relative importance in a video item.

The priority attribute $a_2(p_j)$ can be modified to include an additional term that takes into consideration the relative importance of a face:

$$a_2(p_j) = \frac{1}{2N}\sum_{i=1}^{N}\Phi(f_i) + \frac{1}{2N}\sum_{i=1}^{N}K(f_i)\ ,$$

where $K(f_i)$ is the *character score* of frame f_i. The character score is the average of the relative importance of each face detected in the frame. The relative importance of a face F_k is proportional to the size of the face features cluster to which it belongs:

$$K(f) = \frac{1}{M_f}\sum_{k=1}^{M_f}\text{cluster}(F_k)\ .$$

Requirement P.3: action

Perception of action in video is influenced by three factors [Zettl, 2001; Adams et al., 2002; Hanjalic, 2003; Hanjalic and Xu, 2005; Hanjalic, 2006]: amount of object motion, audio loudness and editing rhythm. These three aspects are included in priority attribute $a_3(p_j)$ which estimates how much action is present in segment p_j. We define it as:

$$a_3(p_j) = \frac{M_\mathrm{a}(p_j) + C_\mathrm{d}(p_j) + A_\mathrm{l}(p_j)}{3} \tag{5.1}$$

where $M_\mathrm{a}(p_j)$, $C_\mathrm{d}(p_j)$, and $A_\mathrm{l}(p_j)$ are the *motion activity*, the *cut density*, and the *audio loudness* of segment p_j, respectively. Not knowing the relative influence of these three factors on the perception of action (similar to [Adams et al., 2002] and [Hanjalic, 2003]), we simply compute their average after normalizing each of them to the interval $[0,1]$. The three factors are computed from the audiovisual content as follows.

The motion activity is defined for each video frame as the standard deviation of the motion vectors after motion estimation [Peker et al., 2001]. In case of MPEG compressed content, the motion activity can be computed using the MPEG motion vectors. The motion activity $M_\mathrm{a}(p_j)$ of a video segment p_j is the average motion activity of its frames normalized to the $[0,1]$ interval.

The amount of action perceived in a video segment depends on the actual motion present as well as on the duration of the segment with respect to its neighboring segments. This aspect, usually referred to as *editing rhythm* or *film pace*, can be modeled by means of what we call the *cut density*.

The cut density gives an indication of the frequency at which shot-cuts occur in a video sequence. For each segment v_i, it is defined with respect to the duration of the neighboring shots as follows.

First we compute a running average of the segments' durations in a window of length $2r+1$ shots (r can be set to 2 for a window length of 5 shots):

$$\bar{D} = \frac{1}{2r+1} \sum_{i=-r}^{r} d(v_i) \ .$$

The reason for using a running average is that only a sequence of short shots contributes to the perception of action, not isolated short segments between long ones.

The average segment duration is then fit to a 1-10 scale in the following way [Peker et al., 2001]:

$$D_\mathrm{a} = \max(1, \min(\lfloor \bar{D} \rfloor, 10)) \ ,$$

where D_a is the clipped average segment duration. Figure 5.11 shows the relationship between the average segment duration $\bar{D}$ and the clipped average segment duration D_a.

The cut density $C_\mathrm{d}(v_i)$ for segment v_i is defined as the inverse of the clipped average segment duration:

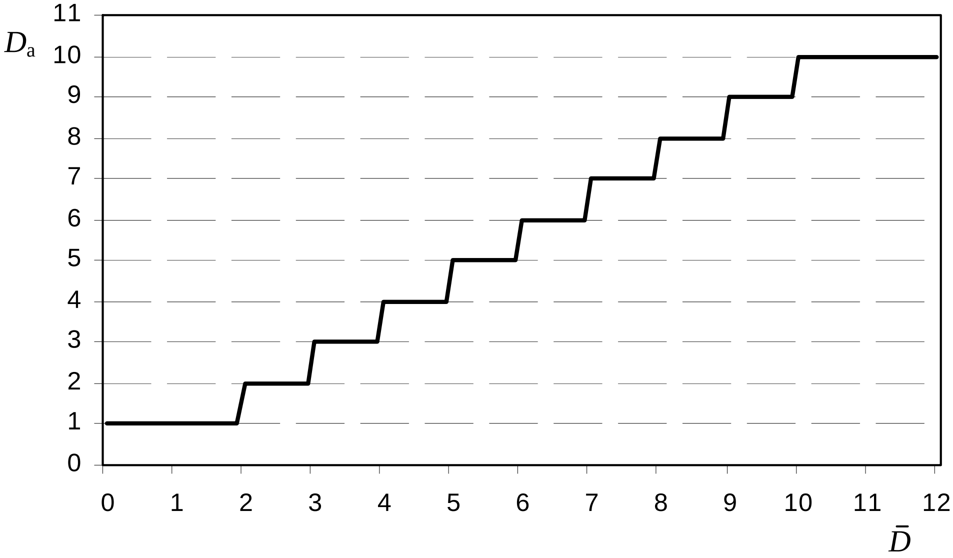


Figure 5.11. Relationship between the average segment duration $\bar{D}$ and the clipped average segment duration D_a.

$$C_\mathrm{d}(v_i) = \frac{1}{D_\mathrm{a}} \; .$$

The audio loudness $A_\mathrm{l}(p_j)$ is simply the average audio energy of the audio segment associated with p_j, normalized to the $[0,1]$ interval.

Requirement P.4: dialogues and speech

To favor segments that represent dialogues or speech, the priority attribute $a_4(p_j)$ is defined as being proportional to the duration of the portions of speech segments that overlap with p_j:

$$a_4(p_j) = \frac{\sum_{a \in S_{p_j}} d\left(\delta_a \cap \delta_{p_j}\right)}{d(p_j)} \; , \tag{5.2}$$

where S_{p_j} is the set of speech segments overlapping with p_j:

$$S_{p_j} = \left\{a | a \in \mathcal{A}_\mathrm{s} \wedge \delta_a \cap \delta_{p_j} \neq \emptyset \right\} \; ,$$

δ_a and δ_{p_j} are the time span intervals of a and p_j, and $\mathcal{A}_\mathrm{s}$ is the set of speech segments of the video item.

Speech segments are located after applying an audio classifier [McKinney and

Breebaart, 2003] to the audio track of the video item. The classifier provides for each audio sample, the probability that the audio sample belongs to a known class such as: speech, music, noise, silence. Intervals of audio samples with speech probability higher than a certain threshold (e.g. 0.6) are considered speech segments. The rest is classified as non-speech. The classification performance of the speech classifier [McKinney and Breebaart, 2003] is around 90% and is sufficient for our purposes.

Audio used in film is usually very rich and includes sounds from multiple sources, intermixed in a way that gives the maximum impact to the audience. Pure speech is rarely used. The speech sound of dialogues is often mixed with background music, noises and other effects. The performances of a speech classifier can be therefore heavily influenced by the presence of background sounds. To overcome this limitation, we use subtitles when available.

Subtitles can be available in the form of time stamped text elements synchronized with the video or can be superimposed to the video frames. In the first case, the time span information of each subtitle is directly available and they can be treated in the same way as speech segments in Equation (5.2). In the case subtitles are superimposed to the video frames, we apply a *text recognition* method [Agnihotri and Dimitrova, 1999; Agnihotri et al., 2002] to extract from each frame the bounding boxes of superimposed captions. Figure 5.12 shows three examples of detected superimposed video text captions. Each bounding box is tracked through consecutive frames to locate the appearing and disappearing time instants of the caption. After this procedure, superimposed captions are treated in the same way as speech segments in Equation (5.2).

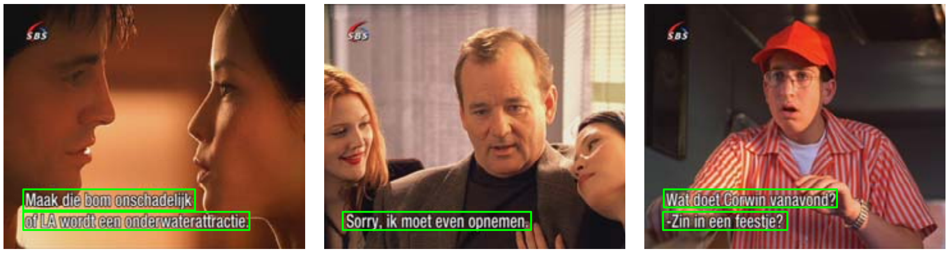

SBS
Maak die bom onschadelijk
of LA wordt een onderwaterattractie.
SBS
Sorry, ik moet even opnemen.
SBS
Wat doet Corwin vanavond?
-Zin in een feestje?

Figure 5.12. Examples of detected superimposed text captions.

The cases in which speech segments or subtitles overlap only partially to the preview segments should be penalized as already mentioned in Section 5.4. For this purpose we subtract from eval(P) a penalty score term $\varepsilon(P)$:

$$\text{eval}(P) = e_1\pi(P) - e_2\rho(P) + e_3\eta(P) + e_4\omega(P) - e_5\varepsilon(P) \ . \tag{5.3}$$

The penalty score term is proportional to the number of speech segments that are partially overlapping with preview segments:

$$\varepsilon(P) = \sum_{j=1}^{N} \frac{\left|\left\{a \in S_{p_j} \ : \ (t_\mathrm{s} \in a \ \wedge \ t_\mathrm{e} \notin a) \vee (t_\mathrm{e} \in a \ \wedge \ t_\mathrm{s} \notin a)\right\}\right|}{\left|S_{p_j}\right|} \ .$$

Requirement P.5: silence

To avoid segments containing silence, the priority attribute $a_5(p_j)$ is proportional to the amount of silence overlapping a preview segment and has a negative sign:

$$a_5(p_j) = -\frac{\sum_{a \in Z_{p_j}} d(\delta_a \cap \delta_{p_j})}{d(p_j)} \ ,$$

where Z_{p_j} is the set of *silence segments* overlapping with p_j and $\mathcal{A}_0 \subseteq \mathcal{A}$ is the subset of audio segments of the video item corresponding to silence:

$$Z_{p_j} = \{a | a \in \mathcal{A}_0 \wedge \delta_a \cap \delta_{p_j} \neq \emptyset\} \ .$$

The audio classifier [McKinney and Breebaart, 2003] used for speech detection is used also to estimate the silence probability. Intervals of audio samples with silence probability higher than a certain threshold (e.g. 0.6) are considered *silence segments*. The rest is classified as *non-silence*.

Requirement P.6: highlights and emotional moments

Detecting highlights and emotional moments in a film is a challenging task to automate, considering the low level of content understanding of state-of-the-art video and audio analysis algorithms. Traditional content analysis aims at automatic understanding of multimedia content by extracting various features from the visual and audio domains. Using low-level features, content analysis algorithms try to imitate the human brain by recognizing objects, concepts and events. Unfortunately current algorithms, in fact, can recognize only a small set of semantic concepts with limited precision.

To partially overcome these limitations, we can apply media production domain knowledge following the so-called *computational media aesthetics* approach [Dorai and Venkatesh, 2002; Dorai and Venkatesh, 2003]. The approach is based on the observation that, in professionally created video, filming techniques and editing operations play a fundamental role in shaping the conveyed message. Professional video production uses certain common conventions often referred to as *film grammar* [Phillips, 1999]. For example, to convey the message that two persons are involved in a dialogue, it is common practice to use close-up shots of the two persons alternated with medium shots showing the two persons together in the same location. Other conventions regard for example the usage of camera angles or the selection of certain sounds and music to manipulate the mood of the audience [Phillips, 1999].

To fulfill Requirement P.6 of including emotional scenes, we define the priority attribute $a_6(p_j)$ as directly proportional to the presence of certain content characteristics that, according to *film grammar*, are used to highlight parts of a narrative film production.

According to the cinematographic principle of *contrast and affinity* [Block, 2001; Zettl, 2001], particularly important moments are highlighted by means of perceptible sudden variations in the content. A powerful clue that directors use to indirectly tell the audience that something important is going to happen, is a sudden change in the audio track. In particular, the start of music is usually an indicator for an interesting point [Eisenstein, 1975; Zettl, 2001; Phillips, 1999].

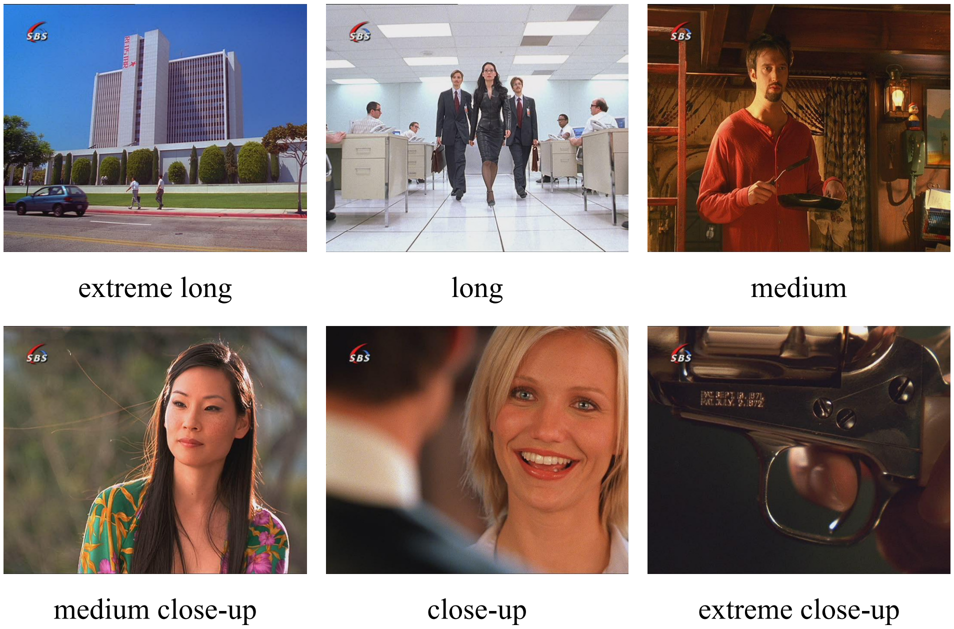
The audio classifier used to detect speech and silence, is also used to estimate the probability that an audio segment contains music [McKinney and Breebaart, 2003].

A sudden change of music genre or pace is also a signal that something interesting might be happening in the film. This clue requires an audio classifier capable of fine distinctions between music genres. The audio classifier at our disposal [McKinney and Breebaart, 2003] was not sufficiently reliable for this purpose, so only the start of music is used as audio clue.

Among the various film grammar rules used by content producers to convey meaning through video, the *field of view* plays a very important role [Mascelli, 1965; Zettl, 2001]. It is determined by the size of a subject in relation with the overall frame, which depends on the distance of the camera from the subject, and the focal length of the lens used. Based on field of view, shots can be classified into different types usually labeled by how big and how near an object appears to the viewers: for example, *extreme long*, *long*, *medium*, *medium close-up*, *full close-up*, and *extreme close-up* (see Figure 5.13).

Of these various shot types, close-up shots are the most interesting for fulfilling Requirement P.6. They show a fairly small part of a scene, such as a character's face, in great detail so that it fills the screen. They focus attention on a person's feelings or reactions, and are used to show people in state of emotional excitement, grief of joy. Close-ups transport the viewers into the scene; eliminate all non-essentials and isolate whatever significant incident should receive narrative emphasis [Block, 2001; Mascelli, 1965; Zettl, 2001; Phillips, 1999]. More specifically they are used to:

- *Underline narrative highlights*, such as important dialogues, player actions or reactions. Whenever dramatic emphasis or increased audience attention is required, the subject should be brought closer to the viewer.



extreme long

long

medium

medium close-up

close-up

extreme close-up

Figure 5.13. Examples of various shot types.

- *Isolate significant subject matter and eliminate all non-essential material from view*. Audience attention thus can be concentrated on an important action, a particular object or a meaningful facial expression.
- *Cue the audience on how they should react*. A reaction close-up of an actor portraying fear, tension, awe, pity or any other action, will stimulate a similar feeling in the viewer.

Close-up shots have already a higher priority for being included in the video preview due to the fact that the face score is proportional to the size of the detected faces. Additionally, one can use a *shot type detector* to classify segments into their type and select preferably close-ups for the preview. An approach to generic shot type detection was presented in [Barbieri et al., 2005] and [Ferrer et al., 2006]. In [Ernst et al., 2005] we presented an approach to shot type detection based on analyzing the depth profile of a shot. Close-up shots are examples of *low-depth of field* shots. These types of shots are characterized by having a small part of the image in focus and the background out of focus. Similarly to close-ups, low-depth of field shots are used in situations where high relative importance is given to a subject. Figure 5.14 shows some examples of low-depth of field shots.

According to film production conventions, reducing the distance between a subject or scene and the audience is a way of signaling the audience that something

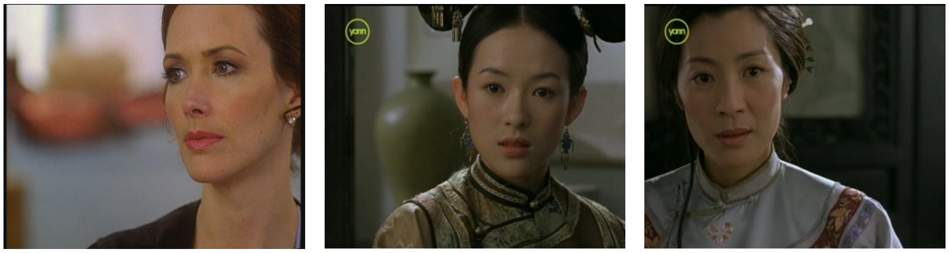


Figure 5.14. Examples of low-depth of field shots. Note the difference in sharpness between foreground and background.

is important. This can be achieved by either moving the camera toward the subject (*camera dolling*) or by using optical zoom. Zoom-in sequences are used to give emphasis to a particular object, situation or character [Zettl, 2001]. They should therefore have high priority in a video preview.

Zoom-in sequences can be detected by applying camera motion or *global motion* estimation techniques. We have implemented a camera motion estimation algorithm known as *luminance projection correlation* method [Uehara et al., 2004]. The priority attribute $a_6(p_j)$ increases if segment p_j is a zoom-in sequence. On the contrary, sequences with panning and tilting decrease the priority score of a video segment. The reason is that video sequences with high camera motion shown out of their context can be difficult to understand. Directors, for example, recommend using a static shot after a camera pan to allow the audience to interpret correctly the scene [Zettl, 2001].

An additional heuristic that can be used to pick interesting scenes from a film, when it is broadcast on a commercial network, is based on detecting advertisements. Broadcasters insert advertisements exactly when the film content creates a peak in the viewers' attention (editing technique known as *cliffhanger*). Because it is in the interest of broadcasters to keep the viewers interested in the rest of the programme, the segment immediately preceding an advertisement contains interesting events and does not disclose key information (e.g. the name of the murderer). If an advertisement is detected, and p_j is preceding it, the priority attribute $a_6(p_j)$ is increased.

Note that this heuristic can be used only with an advertisement detection algorithm that is very precise. In Section 5.6 we mentioned that removing extra content before and after a detected commercial block, could increase recall at the expense of precision. In that situation, the *cliffhanger* heuristic cannot be used.

We define $a_6(p_j)$, the sixth priority attribute, as a linear combination of the factors mentioned above:

- start of music $(+)$,
- zoom-in factor $(+)$,
- panning factor $(-)$,
- presence of a subsequent advertisement $(+)$.

Requirement P.7: story clues

Automatically selecting video segments that can provide clues on the story narrated in a video item requires a high level of semantic understanding of the content. Methods such as automatic speech recognition, speaker recognition, and object detection could be used to try to understand as much as possible of the content. However, the amount of processing involved and the complexity of the implementation would increase dramatically. Additionally, existing algorithms reach various degrees of accuracy and perform rather poorly on unconstrained data.

A simpler and more powerful solution can be devised when textual information such as closed captions or subtitles is available. The basic idea is to analyze the subtitles using traditional information retrieval techniques in order to rank them according to a measure of relative importance. The rank of a subtitle can be associated with the corresponding video segment and added to its priority score.

A simple but effective approach for ranking subtitles within a video item is to look at the frequency of occurrence of each word after having removed all the *stop words*. *Stop words* are frequently-used words in a particular language, such as pronouns, articles but also frequently-used verbs such as auxiliaries. Examples of stop words for the English language are: *about*, *actually*, *because*, *could*, *did*, *either*, *for*, *got*, *have*, *into*, *just*, *know*, *less*, *me*, *not*, *of*, *put*, *rather*, *she*, *that*, *until*, *very*, *was*, *you*.

Stop words lists are easy to create and available for any language. Most existing lists have been created and are meant to be used for indexing and retrieval of textual electronic documents, including web pages. However most subtitles associated with films represent dialogues and standard stop word lists do not include most of the frequently used words in spoken language. To overcome this limitation we extended a standard stop word list with a set of frequently-used spoken terms such as *ah*, *hello*, *hey*, *huh*, *ok*, *really*, *um*, *yes*, that had high frequency of appearance in a set of 225 video items (including mainly American feature films and TV series). The complete stop word list can be found in Appendix A.

If we indicate with $K_M(\boldsymbol{V})$ the set of the M most frequent keywords of video item $\boldsymbol{V}$ (with e.g. $M = 10$), and with $K(p_j)$ the set of keywords of segment p_j, then the priority attribute $a_7(p_j)$ of segment p_j is given by:

$$a_7(p_j) = \frac{\left|K(p_j) \cap K_M(\boldsymbol{V})\right|}{M} \;.$$

Among the subtitles or closed captions that contain one or more important keywords, the ones that end with exclamation marks are usually related to exciting moments in the video. To further align the selection of the preview segments with Requirement P.6, *highlights and emotional moments*, we increase the priority attribute of a segment if its associated subtitles or closed captions contain an exclamation mark.

If additional textual information about the video item is available, such as the title, or an electronic program guide synopsis, it can provide additional clues for calculating the relevance of keywords. The basic idea is to extract from the title and/or the electronic program guide synopsis another set of keywords that is added to the video item's set of keywords. This reflects the idea that segments can be ranked according to their similarity with the electronic program guide summary. This has the advantage of making use of a *good summary* written by an expert in contrast to relying only on the content itself for the analysis.

5.7.2 The redundancy score

The redundancy score defined in Section 4.2.5 is a linear combination of three factors: the *visual redundancy*, the *textual redundancy*, and the *character redundancy*.

The *visual redundancy* score is defined in terms of visual similarity between video segments. Many methods for measuring visual similarity have been proposed in literature (see e.g. [Adjeroh et al., 1998; Bimbo, 1999; Lew, 2001; Qian et al., 2000; Huang et al., 1999; Effelsberg et al., 2000; Pass and Zabih, 1996; Pass and Zabih, 1999; JTC1/SC29/WG11, 2000; Manjunath et al., 2002; Dimitrova et al., 1999]) and in principle any of them could be applied to estimate the visual similarity between two video segments. We have applied a simple method that compares the color histograms of the key-frames using the L_1 norm [Barbieri, 2001]:

$$\sigma_v(p_i, p_j) = 1 - \frac{1}{2}\sum_{k=1}^{B}\left|H_k(p_i) - H_k(p_j)\right| \ ,$$

where $H_k(p_i)$ represents the k-th bin of the color histogram of the key-frame of segment p_i and B is the number of bins of the histogram that we assume is normalized with respect to the number of pixels in the video frames. The visual similarity of two video segments is a real number in the interval $[0,1]$.

The *textual redundancy* score is defined in terms of textual similarity. The textual similarity $\sigma_t(p_i, p_j)$ between two video segments p_i and p_j can be measured by extracting the keywords from the closed captions (or the speech transcript) associated with the video segments, and by counting the number of times these keywords are repeated:

$$\sigma_{\mathrm{t}}(p_i, p_j) = \frac{\left|K(p_i) \cap K(p_j)\right|}{\min\left(|K(p_i)|, \left|K(p_j)\right|\right)}$$

where $K(p_i)$ is the set of keywords associated with the i-th preview segment. $\sigma_{\mathrm{t}}(p_i, p_j)$ is a real number in the range $[0,1]$. In case p_i or p_j do not have associated any keyword, $\sigma_{\mathrm{t}}(p_i, p_j)$ is zero.

To complete the definition of the redundancy score, the character similarity $\sigma_{\mathrm{c}}(p_i, p_j)$ between two segments p_i and p_j included in a preview can be simply measured by counting the number of characters shown both in p_i and p_j:

$$\sigma_{\mathrm{c}}(p_i, p_j) = \frac{\left|C(p_i) \cap C(p_j)\right|}{\min\left(|C(p_i)|, \left|C(p_j)\right|\right)}$$

where $C(p_i)$ represents the set of characters who appear in segment p_i. In case p_i or p_j do not have any character, $\sigma_{\mathrm{c}}(p_i, p_j)$ is zero.

The normalization factors r_1, r_2, and r_3 in the definition of the redundancy score $\rho(P)$ can be all set to $\frac{1}{3}$ so that $\rho(P) \in [0,1]$.

5.7.3 The structure score

The structure score defined in the formal model in Section 4.2.7, is defined as linear combination of five terms, each corresponding to a structural requirement (see Section 3.2.6). In the next sections we provide a specific description of how to compute the five terms.

Requirement S.1: style

The similarity between P and $\boldsymbol{V}$ is represented by $\eta_1 = \sigma_v(P, \boldsymbol{V})$. It is computed as distance between the cumulative histograms of P and $\boldsymbol{V}$. The cumulative histograms are obtained by summing up the corresponding bins of the color histograms of the frames composing the video item and the preview. The distance between the two cumulative histograms is defined as the histogram intersection [Furht, Smoliar & Zhang, 1995].

Requirements S.2 and S.3: uniform coverage and distance between selected segments

Structural requirements S.2 and S.3 are mapped, respectively, to the two terms $\eta_2(P)$ and $\eta_3(P)$ that appear in the structure score $\eta(P)$ (see Section 4.2.7). Equations (4.10) and (4.11) specify already how to compute them.

Requirement S.4: respect scene boundaries

To fulfill Requirement S.4, we subtract from $\eta(P)$ a penalty term, $\eta'(P)$, proportional to the number of segments violating Constraint (4.12) defined as follows:

$$\eta'(P) = \frac{2r}{N} \; ,$$

where r is the number of pairs of segments violating Constraint (4.12), and N is the number of preview segments. Because $\eta'(P)$ is subtracted from $\eta(P)$, the more segments violating scene boundaries, the lower the value of the structure score.

Requirement S.5: tempo

To obtain a preview with a *tempo* that reflects that of the whole video, as stated by Requirement S.5, we have defined in Equation (4.13), Section 4.2.7, η_4 as distance between the original video item's tempo distribution $\boldsymbol{\Psi_V}$ and the preview tempo distribution Ψ_P.

Directors set film tempos during editing by adjusting the duration of the shots. Short shots induce in the audience a perception of action and fast pace. On the contrary, long shots induce in the audience a perception of calm and slow pace. Perceived film tempo is also influenced by the amount of action (actual motion) present in the video scenes and the audio loudness.

The tempo of a video item can be measured using a linear combination of cut density, motion activity, and audio loudness. In fact it is already computed for priority attribute $a_3(p_j)$ (see Equation (5.1)). An alternative definition of a tempo function that depends on shots duration and motion can be found in [Adams, Dorai & Venkatesh, 2002; Dorai & Venkatesh, 2002]. Adams et al. [Adams, Dorai & Venkatesh, 2002] propose a tempo function in which the amount of action and the segments duration are smoothed over large windows of segments. This is because their aim is to locate directly interesting segments within a video item. Because our purpose is instead to have a good representation of the overall pace of a video item, we do not smooth the values of a_3 across segments, but we use it directly to construct a histogram.

The tempo distribution $\boldsymbol{\Psi_V}$ of the video item $\boldsymbol{V}$ is modeled using a histogram that counts how many video segments fall within predefined ranges of the priority attribute a_3. Similarly we compute the tempo distribution Ψ_P of the preview P.

The distance between the two tempo distributions is defined as the L_1 norm between the two histograms $\boldsymbol{\Psi_V}$ and Ψ_P normalized with respect to the total number of video segments.

Requirement S.6: balance action and dialogue

To balance action and dialogue we have defined in Section 4.2.7 two classes of segments: *action* segments and *dialogue* segments. *Action* segments are characterized by having a high activity, while *dialogue* segments have a slower pace and contain spoken dialogues.

Video segments with a value of the priority attribute a_3 (includes motion activity, cut density and audio loudness) higher than a certain threshold θ_a are considered *action* segments. θ_a is fixed empirically to 70% of the maximum value of a_3 across all video segments.

We classify segments as *dialogue* when their action level is low and there is speech: in practice when $a_3 < \theta_d$ and $a_4 > \theta_s$. θ_d is fixed empirically to 40% of the maximum value of a_3 across all video segments. a_4 represents the speech duration relative to the duration of the segment. The threshold θ_s is set empirically to 60%. Each segment whose audio is classified as speech for at least 60% of the time and that has a low action level is considered dialogue.

5.7.4 The order score function

The details of the order score function are already described with a sufficient level of detail in Section 4.2.8.

5.7.5 Local search

From the previous steps (see Figure 5.2 on page 59): *temporal segmentation*, *micro segmentation*, *segment compensation*, *macro segmentation*, and *pre-filtering*, we obtain a set of candidate segments with associated numerical descriptors among which a selection has to be made that optimizes the evaluation function $\mathrm{eval}(P)$ taking into consideration the requirements that a preview should fulfill as presented in Chapters 3, 4, and 5. The structure of $\mathrm{eval}(P)$ has been already presented in the previous sections. Here we propose local search as basic method for its optimization.

Local search denotes a class of algorithms used to find approximate solutions to combinatorial optimization problems [Aarts and Lenstra, 1997]. An instance of a combinatorial optimization problem is specified by a *solution set S*, also called *solution space*, and an *objective function* $f : S \to \mathbb{R}$ that associates to each solution a numerical value. The problem (*maximization* version) is to find a solution with the highest value of the objective function.

A key aspect of local search algorithms is the definition of *neighborhood function* $N : S \to 2^S$ which defines for each solution a set of alternative solutions that are in some sense *near* to it. Local search algorithms begin with an initial solution and then try to find better solutions by exploring the neighborhoods.

A solution is called *locally optimal* for a given neighborhood function if there is no other solution in the neighborhood with higher value of the objective function. A solution is called *globally optimal* if no other solution in the entire solution space exists that has a higher value of the objective function.

A simple local search algorithm is *iterative improvement*, or *hill climbing* shown in Figure 5.15. The algorithm begins with an initial solution and then tries to improve it by searching its neighborhood for a solution with higher value of the objective function. If such a solution is found, it replaces the current solution, and the search continues. Otherwise the algorithm stops and returns the current solution that is by definition only locally optimal. To avoid poor local optima, the algorithm can be restarted many times from different initial solutions. However, if the initial solutions are poorly chosen and there exist many poor local optima, the number of restarts might be very high.

```
INITIALIZE $s$;
repeat
    GENERATE $s' \in N(s)$;
    if $f(s') \geq f(s)$ then
        $s := s'$;
    end;
until $f(s) \leq f(s')$ FOR ALL $s' \in N(s)$
```

Figure 5.15. Iterative improvement algorithm for a maximization problem.

The simulated annealing algorithm [Kirkpatrick et al., 1983; Černý, 1985] overcomes these limitations by replacing the criterion of accepting a new solution only when it improves the current one, with a stochastic criterion that accepts also worse solutions. While in the iterative improvement algorithm neighboring solutions are accepted only if they improve the objective function value, in simulated annealing also solutions that do not improve the objective function are accepted, although with a probability that is gradually decreased during the execution of the algorithm. The probability of accepting worse solutions is controlled by a set of parameters determined by a *cooling schedule*.

Simulated annealing is inspired by thermodynamics, specifically by the way cooling metals anneal. At high temperatures, the atoms of a metal have high energies and can have a certain degree of freedom in restructuring themselves. As the temperature is reduced, the energy of the atoms decreases along with their mobility. If the cooling process is carried out too quickly, the solid will present many irregularities and defects in its crystal structure. But if the temperature is reduced at a sufficiently slow rate, the atoms will form a more consistent and stable crystal structure, corresponding to a state of minimal energy, allowing the metal to be much more durable.

Simulated annealing emulates this process: physical states correspond to solutions, the energy of the physical system corresponds to the objective function value of a solution (for minimization problems, the value of a solution is called cost and has to be minimal just as the energy of a crystalline structure), and a control parameter t is introduced that corresponds to the temperature.

In simulated annealing, given a solution s, a neighboring solution s' is accepted if it is a better solution or, in case it is worse, the probability of acceptance depends on how worse the neighboring solution is and on the current value of the control parameter t:

$$P(s'|s) = \begin{cases} 1 & \text{if } f(s') \geq f(s), \\ \exp\left(\frac{f(s')-f(s)}{t}\right) & \text{if } f(s') < f(s). \end{cases}$$

Figure 5.16 shows the pseudo-code of the simulated annealing algorithm for a maximization problem.

```
INITIALIZE $s,t$;
repeat
    repeat
        GENERATE $s' \in N(s)$;
        if $f(s') \geq f(s)$ then $s := s'$;
        else if $\exp\left(\frac{f(s')-f(s)}{t}\right) < \text{random}[0,1)$ then $s := s'$;
    until EQUILIBRIUM CRITERION
    UPDATE CONTROL PARAMETER $t$
until STOP CRITERION
```

Figure 5.16. Simulated annealing algorithm for a maximization problem.

As shown in Figure 5.16 there are two loops that are performed during the simulated annealing algorithm. At the beginning, the control parameter has a high value that makes all generated solutions acceptable. At each iteration of the outer loop, the control parameter is decreased and the chances of accepting a deteriorating solution decreases as well. When the control parameter reaches values sufficiently close to zero, the chance of accepting a deteriorating solution is nearly zero as well, and the algorithm behaves equivalently to iterative improvement.

In theory, if the control parameter is decreased sufficiently slowly, the algorithm can provide the optimal solution. In practice, the time required for this to happen can be higher than performing an exhaustive search. For this reason a rather fast cooling schedule is usually adopted by practitioners of simulated annealing with the drawback that the solution obtained is only an approximate solution. In our case we have adopted one of the simplest but most effective cooling schedules: the *geometric schedule*. After the *i*-th iteration of the outer loop in Figure 5.16, the control parameter is decremented according to the following formula:

$$t_{i+1} = \alpha \cdot t_i \ ,$$

where $0 < \alpha < 1$. In our case we have chosen empirically $\alpha = 0.9$. At each iteration, the control parameter t decreases to 90% of its previous value. To simplify the calculations, the objective function is normalized so that it returns a value between 0 and 1.

In Chapter 4, the objective function was defined in Equation (4.20) as:

$$\mathrm{eval}(P) = e_1\pi(P) - e_2\rho(P) + e_3\eta(P) + e_4\omega(P) \ .$$

In Section 5.7.1, Equation (5.3), an additional penalty term $\varepsilon(P)$ was added to take into account Constraints (4.5) and (4.6) (*speech* and *subtitles continuity*):

$$\mathrm{eval}(P) = e_1\pi(P) - e_2\rho(P) + e_3\eta(P) + e_4\omega(P) - e_5\varepsilon(P) \ .$$

In the case when all requirements have the same weight, the coefficients $(e_1, \ldots, e_5)$ can be chosen to be all equal to $\frac{1}{3}$. In this case the objective function returns a value between $-\frac{2}{3}$ and 1: in the best case, $\pi(P)$, $\eta(P)$, and $\omega(P)$ are all 1 while $\rho(P)$ and $\varepsilon(P)$ are 0; in the worse case $\pi(P)$, $\eta(P)$, and $\omega(P)$ are all 0 while $\rho(P)$ and $\varepsilon(P)$ are 1. To obtain a value of the objective function between 0 and 1, we simply add a constant $\frac{2}{3}$ and we multiply the result for $\frac{3}{5}$:

$$\mathrm{eval}'(P) = \frac{3}{5}\left(\frac{\pi(P) - \rho(P) + \eta(P) + \omega(P) + \varepsilon(P) + 2}{3}\right) \ .$$

In this way, because the objective function $\mathrm{eval}'(P)$ returns always a value between 0 and 1, we can simply set the control parameter to be equal to 1 at the first iteration: $t_1 = 1$.

For each value of the control parameter, the simulated annealing algorithm runs through a number of cycles until the so called *equilibrium criterion* is met. The equilibrium criterion we have adopted consists in allowing a maximum number of iterations 100 times bigger than the number of candidate video segments available before the optimization step. Additionally, we allow a maximum number of successful improvements that is 10 times bigger than the number of candidate segments. This equilibrium criterion was chosen heuristically after measuring execution times for typical video items having a couple of thousand of candidate video segments. Depending on the usage scenario (e.g. whether the algorithm is running on a consumer device with limited processing power, memory and time available or whether the algorithm is used in an off-line system with plenty of computational resources and time available), the equilibrium criterion can be changed to allow a higher number of iterations or successful improvements so that a smaller or larger part of the solution space is explored at each iteration.

Once the equilibrium criterion for a given value of the control parameter is met, the control parameter is lowered. This is repeated until the *stop criterion* is met. The *stop criterion* we have adopted consists in allowing iterations until a minimum temperature t_{min} is reached. The value of the t_{min} was fixed to 10^{-4} using trial and error. In our experiments, lower values of t_{min} do not result in significant improvements of the objective function.

The last elements required by the simulated annealing are an initial solution and a neighborhood function that, given a solution, generates a new one from a set of solutions somehow *near* the given one. In our case a solution is a sequence of video segments selected among the candidates obtained from the *pre-filtering* step. The neighborhood of a solution is the set of previews that can be obtained from a given preview by making a small change in its video segments. A small change in the selection of the segments or in their order should result in a small change of the value of the objective function $\text{eval}(P)$.

As initial solution, we select randomly one segment from each scene until the maximum allowed preview duration is met. We then define the neighborhood of a preview P as the set of previews that can be obtained by replacing one of the segments with another randomly chosen segment from the same scene if its duration does not cause the preview to exceed the maximum duration. The assumption is that within a scene, many segments are similar to each other or are related in terms of spoken content, characters, etc. By exchanging a segment with another one from its scene, the preview does not change drastically. When replacing a segment causes the preview to exceed its maximum duration, the segment is discarded and a new one is selected from one of the scenes.

Our definition of neighborhood function might seem very restrictive because if a given preview does not include segments from a given scene, the neighborhood function might not introduce them at all. In practice, this situation rarely happens given the typical durations of scenes (around 12 minutes), segments (between 3 and 5 seconds) and previews (e.g. between 60 and 120 seconds). If the goal would have been to generate very short previews using only very few segments, our neighborhood function would have not been a good choice.

5.8 Results

To evaluate the quality of the solutions generated with the algorithm described in the previous sections, it would be interesting to compare them with, for example, a preview generated in a complete random way. Unfortunately, computing such a lower bound in quality for a randomly generated solution is not straightforward. In our model we have not defined penalty functions for the constraints that are implicitly solved by the *temporal segmentation*, *micro segmentation*, *segment compensation*, and *macro segmentation* steps preceding the *pre-filtering* step. Therefore a totally random preview would not fulfill any of the requirements that are taken care in the *preparation* step, and this would not be reflected in $\mathrm{eval}(P)$. A more useful lower bound can be found by randomly selecting segments *after* the *preparation* step.

After the *pre-filtering* step, we obtain a set of candidate segments that satisfy most of the constraints that were explicitly defined. After this step, either a constraint is already satisfied, or it has been translated into a penalty function and included in the calculation of $\mathrm{eval}(P)$. Strictly speaking, a solution obtained after optimization might still not be feasible according to Definition 4.13 of feasible previews given in Section 4.3. In particular, the only constraints that might not be solved even after optimization are the relations (4.5) and (4.6) that are mapped to the penalty term $\varepsilon(P)$. The result of the optimization procedure is a local optima and as such it is only an approximate solution that might or might not be feasible in the strict sense defined in Chapter 4. Ultimately the quality of the generated solutions is measured by means of a user study as reported in Chapter 6.

In this section we are interested in comparing our local search method with other methods of selecting segments after all the preceding steps have been already carried out. Given a set of candidate segments $C_\mathrm{s} = (v_1, \ldots, v_N)$, a maximum preview duration D_max and a minimum preview segment duration d_min, we compare the quality of the solutions generated using local search with two different algorithms: *random selection*, and *subsample selection*.

The *random selection* algorithm is shown in Figure 5.17. It selects segments randomly from the pool of candidates until the duration of the preview is just above its maximum allowed value D_max.

```
algorithm RANDOM SELECTION
Given candidate segments $C_\mathrm{s} = (v_1, \ldots, v_N)$, maximum preview duration $D_\mathrm{max}$;
begin
    Initialize preview $P := \emptyset$;
    while $\sum_{v \in P} d(v) < D_\mathrm{max}$ do
    begin
        repeat
            $i := 1 + (N-1)\mathrm{random}[0,1]$;
            $v_i \in C_\mathrm{s}$;
        until $v_i \cap P = \emptyset$;
        $P := P \cup v_i$;
    end
    if $\sum_{v \in P} d(v) > D_\mathrm{max}$ then $P := P \setminus v_i$;
end
```

Figure 5.17. *Random selection* algorithm for generating a preview.

The *subsample selection* algorithm selects, from the pool of candidates, video segments that are approximately uniformly distributed in time throughout the whole video item. Figure 5.18 shows its pseudo-code.

```
algorithm SUBSAMPLE SELECTION
Given candidate segments $C_s = (v_1, \ldots, v_N)$, maximum preview duration $D_{\max}$,
minimum preview segment duration $d_{\min}$;
begin
    Initialize preview: $P := \emptyset$;
    Calculate subsample step: $s := \lfloor (N/D_{\max}) d_{\min} \rfloor$;
    Initialize segment index: $i := 1$;
    while $(i < N) \wedge (\sum_{v \in P} d(v) < D_{\max})$ do
    begin
        $P := P \cup v_i$;
        $i := i + s$;
    end
    if $\sum_{v \in P} d(v) > D_{\max}$ then $P := P \setminus v_i$;
end
```

Figure 5.18. *Subsample selection* algorithm for generating a preview.

Note that in theory, these last two algorithms do not guarantee to generate previews with duration close to $D_{\max}$ for all possible values of $d_{\min}$ and for all possible sets of candidate segments. In practice, for typical video items, $D_{\max}$, $d_{\min}$ and the sets of candidate segments are such ($D_{\max} =$ 60–90 s, $d_{\min} =$ 4–6 s) that the *random selection* and *subsample selection* algorithms are capable of generating previews with durations very near the desired values.

A set of 30 content items of different genres was used for the evaluation. Table 5.3 shows the content items' titles, genres and durations.

We compare the quality of the solutions generated by the three algorithms for previews of 90 seconds. For the *random selection* and *local search* algorithms we performed 10 runs per instance. In eval(P) we kept all the weights equal. In Table 5.4 we report the mean and standard deviation of the objective function for the three algorithms. The results are plotted in Figure 5.19.

As expected, the *random selection* algorithm shows the poorest performance. The *subsample selection* generates consistently better solutions than the *random selection*. For all the content items, the local search algorithm performs much better than the *subsample selection* and the *random selection* algorithms.

Table 5.3. Video items used in the evaluation of the algorithms.

Title	**Genre**	**Duration**
1. 007 The World is not Enough	action, adventure, thriller	128 minutes
2. 2001: A Space Odyssey	adventure, sci-fi	141 minutes
3. Chain Reaction	action, thriller, drama	106 minutes
4. Charlie's Angels	action, comedy, adventure	98 minutes
5. Deadly Past	action, crime, drama	90 minutes
6. Dog Day Afternoon	crime, drama, thriller	124 minutes
7. Far and Away	adventure, drama	140 minutes
8. Forrest Gump	comedy, drama, romance	142 minutes
9. Friends, Season 5, Ep. 17	comedy, romance	20 minutes
10. From Dusk Till Dawn	action, comedy, horror	108 minutes
11. Gladiator	action, adventure, drama	155 minutes
12. Harry Potter and the Chambers of Secrets	adventure, fantasy, mystery	161 minutes
13. If Someone Had Known	drama	80 minutes
14. Jurassic Park	action, adventure, sci-fi	127 minutes
15. Master and Commander	action, adventure, drama	138 minutes
16. Mission Impossible	action, adventure, thriller	110 minutes
17. Presumed Innocent	crime, drama, thriller	127 minutes
18. The Hunt for Red October	action, adventure, thriller	134 minutes
19. Scary Movie 2	comedy, horror	83 minutes
20. Seduced by a Thief	crime, thriller	90 minutes
21. Showgirls	drama	128 minutes
22. Star Wars Episode I	action, fantasy, sci-fi	133 minutes
23. Terminator 2	action, sci-fi, thriller	137 minutes
24. The Grifters	crime, drama, thriller	119 minutes
25. The Nanny, Season 1, Ep. 0	comedy	24 minutes
26. The Peacemaker	action, thriller	124 minutes
27. The Scarlet Letter	drama, romance	135 minutes
28. The Silence of the Lambs	crime, thriller	118 minutes
29. Top Gun	action, drama, romance	110 minutes
30. Young Americans	crime, drama	104 minutes

Table 5.4. Means (and between brackets standard deviations) of the objective function for the three algorithms for previews of 90 seconds.

Content item	random	subsample	local search
1. 007 The World is not Enough	0.025 (0.002)	0.039	0.47 (0.09)
2. 2001: A Space Odyssey	0.028 (0.002)	0.038	0.41 (0.08)
3. Chain Reaction	0.028 (0.002)	0.034	0.42 (0.08)
4. Charlie's Angels	0.032 (0.002)	0.043	0.56 (0.13)
5. Deadly Past	0.028 (0.002)	0.036	0.46 (0.11)
6. Dog Day Afternoon	0.020 (0.002)	0.029	0.32 (0.07)
7. Far and Away	0.033 (0.002)	0.048	0.55 (0.08)
8. Forrest Gump	0.031 (0.002)	0.043	0.45 (0.05)
9. Friends	0.024 (0.002)	0.026	0.43 (0.06)
10. From Dusk Till Dawn	0.029 (0.002)	0.038	0.43 (0.07)
11. Gladiator	0.029 (0.003)	0.040	0.45 (0.08)
12. Harry Potter	0.024 (0.003)	0.037	0.48 (0.11)
13. If Someone Had Known	0.022 (0.002)	0.028	0.38 (0.08)
14. Jurassic Park	0.025 (0.002)	0.032	0.42 (0.05)
15. Master and Commander	0.031 (0.003)	0.043	0.63 (0.10)
16. Mission Impossible	0.029 (0.002)	0.038	0.52 (0.11)
17. Presumed Innocent	0.022 (0.002)	0.029	0.35 (0.07)
18. The Hunt for Red October	0.030 (0.002)	0.042	0.49 (0.09)
19. Scary Movie 2	0.020 (0.002)	0.027	0.32 (0.06)
20. Seduced by a Thief	0.029 (0.002)	0.037	0.47 (0.11)
21. Showgirls	0.030 (0.002)	0.039	0.49 (0.11)
22. Star Wars Episode I	0.030 (0.002)	0.037	0.48 (0.11)
23. Terminator 2	0.029 (0.002)	0.039	0.58 (0.10)
24. The Grifters	0.032 (0.002)	0.045	0.52 (0.08)
25. The Nanny	0.028 (0.002)	0.038	0.47 (0.09)
26. The Peacemaker	0.021 (0.002)	0.029	0.35 (0.08)
27. The Scarlet Letter	0.031 (0.002)	0.041	0.52 (0.09)
28. The Silence of the Lambs	0.030 (0.002)	0.040	0.50 (0.12)
29. Top Gun	0.022 (0.002)	0.031	0.36 (0.08)
30. Young Americans	0.021 (0.003)	0.031	0.34 (0.07)

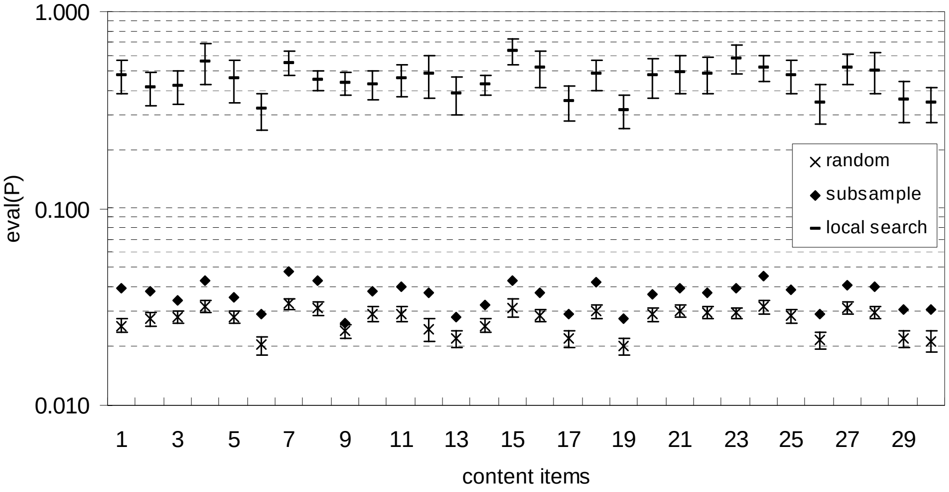


Figure 5.19. Mean scores of eval(P) for the three algorithms evaluated. Error bars show ± standard deviation. Note that the scale of the vertical axis is logarithmic and therefore the error bars are asymmetric.

5.9 Post processing

If the video segments selected by the optimization procedure are simply put one after the other and thus presented out of their original context, users can have misperceptions of action, pace and, of course, misunderstand the actual story line.

A solution is to separate explicitly the preview segments with *fade-out* and *fade-in* transitions. Fades provide the necessary *glue* to preserve continuity between segments [Phillips, 1999]. Furthermore, they give extra useful time to the users to think about what they saw in a video segment before the video switches to a new segment from another context. Without fades, the changes can be too fast and there is no guarantee that continuity is preserved.

5.10 Conclusions

Our numerical evaluation has shown that the local search approach generates previews that are, according to our model, better than previews generated by randomly or uniformly selecting segments. This does not necessarily mean that the degree of fulfillment of the requirements achieved with our method is sufficient and satisfactory for users. To validate our solution, there is no other way than assessing the user satisfaction by means of a user study. In the next chapter we validate our approach by means of a user panel in which previews generated using our algorithm are compared to human-made previews.

6

Evaluation

In this chapter we validate our *optimization-based* approach by means of a user study. The algorithm is evaluated in terms of the intrinsic quality of the generated previews.

This chapter is structured as follows. In Section 6.1 we present the hypothesis upon which the user study is based. The method we adopted is discussed in Sections 6.2. Participants, test material and set-up are described in Sections 6.3, 6.4 and 6.5. The results of the test are discussed in Section 6.6. Our conclusions are presented in Section 6.7.

6.1 Hypothesis

To evaluate the performance of our *optimization-based* approach, the algorithm needs to be tested against *control methods* for generating a video preview. A simple algorithm that can be used for benchmarking is the *subsample selection*[1] algorithm. It generates a video preview by selecting segments almost equally distributed throughout the video with almost no content evaluation and the consideration of only a few user requirements (see Section 5.8 of the previous chapter for more details). We expect previews generated using the *subsample* technique would

[1] For simplicity, in the rest of this chapter we will refer to this algorithm as the *subsample* algorithm.

be of considerably lower quality than previews generated using our *optimization-based* approach.

As an additional control method for our evaluation we have chosen *manually made* previews. In order to have unbiased realistic samples of previews made by humans, we have involved an expert in the domain of film and video editing who was not involved with the development of our algorithms. Given a video item, the expert was asked to make a preview of a given duration aiming at providing an overview of the storyline and a fair impression of the atmosphere without giving away too many plot clues. These *manually made* previews certainly represent an upper-bound for the overall quality of previews.

On a one-dimensional scale, the hypothesis on the quality of the previews generated by our *optimization-based* approach and the two control methods is visualized in Figure 6.1 with low quality on the left of the scale. Note that, although in Figure 6.1, the three methods are represented as equally distributed in terms of quality, we have a less restrictive hypothesis. We only assume that in terms of quality of results, the *subsample* method will be worse than the *optimization-based* approach and *manually made* previews will have the highest possible quality.

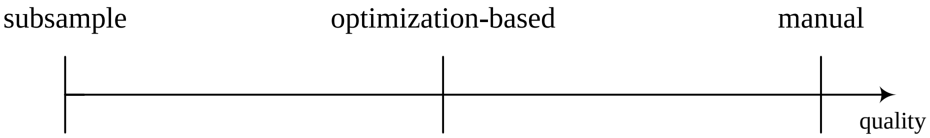


Figure 6.1. Hypothesis of one-dimensional quality scale of the previews generated by the three methods considered in the test.

The question we aim at answering with this user study is whether our *optimization-based* approach actually generates better video previews than the *subsample* method. Additionally we would like to know how much higher the quality of *manually made* previews is with respect to the results of our *optimization-based* algorithm. The general hypothesis we want to verify in this test is:

H0. Our *optimization-based* approach provides a *higher quality overview* of a video item than the *subsample* method.

What we mean with *higher quality overview* needs to be further specified in order to properly design the test. In light of the requirements described in Chapter 3, we can break down the generic hypothesis **H0** into four more specific ones:

H1. The average segment duration in previews generated by our *optimization-based* approach is better than in previews generated by the *subsample* method and worse than in *manually made* previews.

H2. Audio transitions between segments of previews generated by our *optimization-based* approach are better than in previews generated by the *subsample* method, and worse than in *manually made* previews.

H3. Visual transitions between segments of previews generated by our *optimization-based* approach are better than in previews generated by the *subsample* method, and worse than in *manually made* previews.

H4. Previews generated by our *optimization-based* approach are more informative than previews generated by the *subsample* method, and less informative than *manually made* previews.

H5. Previews generated by our *optimization-based* approach give a better idea of the atmosphere of a video item than previews generated by the *subsample* method, and a worse idea of the atmosphere than *manually made* previews.

H6. Previews generated by our *optimization-based* approach are considered more useful for choosing whether to watch a video than previews generated by the *subsample* method, and less useful than *manually made* previews.

6.2 Method

A factorial within-subject design with two independent variables was applied. As discussed in the previous section, three methods to generate video previews were compared (*manual*, *optimized*, and *subsample*) defining an independent variable, *algorithm*. The second independent variable *content* refers to the video item used. Each subject judged all preview versions. The advantage of using this design instead of a between-subject design (in which separate groups of participants watch only one of the preview versions) is that a smaller number of participants is necessary. The disadvantage is that, because subjects see all preview versions, in evaluating a preview, a subject is influenced by having seen other versions.

The goal of the study is to obtain a ranking of the three methods of generating a preview that reflects the users' perceived quality. There are at least three methods that can be used to obtain such ranking by means of a user study: *ranking*, *paired comparison*, and *rating*.

The first method, *ranking*, consists of presenting the users the three previews and asking them to sort them according to the perceived quality. This method requires users a considerable effort in terms of memory because at the moment of ranking, the subjects have to remember all the three previews in order to compare them. Therefore this method is not suitable for the comparison of video content.

In the second method, *paired comparison*, two previews at a time are shown to a user who has just to choose the best of them. By repeating the comparison between pairs of previews obtained with different methods, a ranking of all the methods can be obtained. This method takes advantage of the fact that, in principle, choosing between only two alternatives should not be too difficult for subjects. This apparent advantage, however, is also the cause of the main disadvantages. In choosing between one of two alternatives users make complex evaluations of the positive and negative aspects of the two alternatives. There might be aspects for which one of the two alternatives is better than the other one and aspects for which it is the other way around. The outcome of this evaluation is reflected in their choice but it is extremely difficult to analyze the reasons it. Another disadvantage of using the *paired comparison* method is caused by the fact that the two previews to compare are shown sequentially. In choosing for the best one, the second shown preview is easier to remember and fresher in the mind, making in fact the comparison unfair. An additional disadvantage is that obtaining a ranking of three methods using paired comparison requires inherently more time than direct *ranking* or *rating*. If we consider three methods, A, B, and C, ranking requires showing three previews: A, B, and C. Paired comparison, instead, requires showing three pairs of previews: A-B, A-C, and B-C. Furthermore, to prevent a user from seeing twice the same preview, a between-subject design should be used requiring many more users than a within-subject design.

The third method, *rating* or *scaling*, consists of showing users one preview at a time and asking them to rate directly, on a given scale, different aspects of a preview. The main advantage of *rating* with respect to *ranking* and *paired comparison*, is that users do not need to recall and compare positive and negative aspects of multiple alternatives. They see only one preview at a time and they judge it. The obvious disadvantage is that the quality scale is initially unknown to the users. The ratings are valid only after a *calibration* phase that might be difficult to estimate.

For this study we chose the method of direct rating various aspects of the previews. To mitigate the *calibration* problem, we gave the subjects the possibility of calibrating their scale with two clear examples of a very bad and a very good preview at the beginning of the test.

One of the observations from previous studies [Visser, 2005] is that users find it difficult to compare different previews of the same video item. As result, subjects might rate equally most of the previews. Therefore, we ask users to rate different aspects of the preview, aiming at measuring the perceived differences between previews even if users might find it too difficult to express their overall preference for one or another preview.

6.3 Participants

Forty subjects (20 female, 20 male) participated voluntarily during their normal working time. They were recruited from the staff and student population of the

High Tech Campus in Eindhoven, The Netherlands. None of them had been involved in the development of the algorithms for video preview generation.

An estimate of how many participants would be necessary based on considerations of statistical power was not possible because the distribution of the population with respect to the hypothesis to test was unknown. Additionally there are no similar studies from which we can learn. For these reasons, we started with a pilot study with a small number (10) of participants and, based on intermediate analysis of the collected data, we decided to continue the test until we reached the number of 40 participants.

The average age of the participants was 28 years (median: 27, min: 22, max: 42, standard deviation: 4.4). Of these, 24 participants (60%) were Dutch native speakers and did the test in Dutch. 16 participants (40%) did the test in English although no one was an English native speaker.

All subjects were interested in movies. 23 participants (57.5%) watch more than one film per week, 15 (37.5%) watch between 1 and 4 films per month and only 2 subjects (5%) watch less than one film per month.

6.4 Test material

Table 6.1 reports the video items used in the test. They were chosen among the popular genres: *action*, *thriller*, *comedy*, and *drama*.

Table 6.1. Video items used in the user study.

	Title	**Genre**	**Duration**
P1	007 The World is not Enough	adventure, action, thriller	128 minutes
P2	Friends, Season 5, Episode 17, “The One with Rachel’s Inadvertent Kiss”	romance, comedy	20 minutes
P3	Master and Commander	adventure, action, drama, war	138 minutes
P4	The Nanny, Season 1, Episode 0	comedy	24 minutes
E1	Harry Potter and the Chambers of Secrets	adventure, fantasy, mystery	161 minutes
E2	Forrest Gump	comedy, drama, romance	142 minutes

The source of each video was the official published DVD. Each video item was analyzed and rendered with its original English soundtrack and aspect ratio and no subtitles were displayed along with the previews.

Three different previews (*random*, *subsample* and *optimized*) were created for each of the first four video items (*007, Friends, Master and Commander* and *The Nanny*). Each preview was 90 seconds long. The *subsample* previews were made using $d_{\text{min}} = 5$ seconds.

The video items *Harry Potter* and *Forrest Gump* were used as examples to allow the subjects to calibrate their judgment. For *Harry Potter* only one *optimized* preview was created and for *Forrest Gump* a preview was generated using the *random selection* algorithm described in Figure 5.17 of the previous chapter. These two previews were meant to represent respectively an example of good quality preview and an example of bad quality preview.

People usually remember elements of the previews and of the video items they see and use them in their evaluation during the test. The order in which the different previews and the different video items are shown can therefore influence the outcome of the test. To minimize this influence, the presentation order should be as *balanced* as possible. With three algorithms and four video items in a within-subject design, we obtain twelve *runs* per participant. These runs should satisfy the following constraints. For each participant:

- The same algorithm should not be shown twice in two sequential runs;
- The same video item should not be shown twice in two sequential runs.

Over the 40 participants:

- Each algorithm should be in each of the 12 run slots equally often;
- Each video item should be in each of the 12 run slots equally often;
- Each algorithm is preceded by each other algorithm the same number of times;
- Each video item is preceded by each other video item the same number of times.

We obtain a combinatorial problem of which we need one feasible solution. The problem was solved using simulated annealing by E. Stienstra [Barbieri and Stinstra, 2006].

6.5 Procedure

The participants were given a short oral introduction about the idea of automatic video preview generation and on the purpose of the test. They were told that the video previews were automatically generated by algorithms and that the aim of the experiment was a comparison of three automatic video preview generation algorithms. They were not aware that one of the algorithms was *manually made*.

Participants could choose between two equivalent versions of the test: a Dutch[2] and an English version.

After three questions about gender, age and film watching behavior (see the screen shots of the user interface in Appendix B), before starting evaluating the previews, the participants were shown two examples of video previews (*Harry Potter* and *Forrest Gump*). These examples were meant to help the participants calibrate their quality scale.

After the examples, the participants were shown the 12 previews in balanced order. The participants were not aware of which algorithm was responsible for the creation of a particular preview. Each preview was numbered from 1 to 12 and no information about the videos was given in advance.

It was possible to pause and resume playing a preview but not watching it more than once.

After each preview (including the two calibration examples) participants had to answer seven multiple choice questions (between brackets the Dutch version of the question):

Q1. The preview you saw is made out of fragments of a movie. How was the average duration of the fragments?
(De video preview bestaat uit een aantal fragmenten uit de film. Hoe is de gemiddelde lengte van de fragmenten?)

Q2. How were the audio transitions between the fragments? (Consider e.g. if speech was abruptly cut)
(Wat vindt u van de geluidsovergangen tussen de fragmenten? (Denk bv aan abrupt afbreken van zinnen))

Q3. How were the transitions between the fragments visually?
(Wat vindt u van de visuele overgangen tussen de fragmenten?)

Q4. Was the preview informative?
(Was de preview informatief?)

Q5. Based on the preview you just saw, do you think you have a good impression of the atmosphere (e.g. pace, type of humour) of the movie?
(Denkt u dat u, op basis van de preview die u net zag, een goede indruk heeft van de sfeer (bv snelheid, soort humor) van de film?)

Q6. Overall, how good is the overview of the movie given by this preview?
(Hoe goed vindt u in het algemeen het overzicht dat de preview geeft van de film?)

[2] A translation from English into Dutch was kindly provided by Hans Weda.

Q7. How useful would this preview be for choosing to watch this movie?
(Hoe bruikbaar zou deze preview zijn om te kiezen of u de film wilt zien?)

In the user interface, each question had a multiple choice scale from 1 to 10. The scale of question **Q1** ranged from 1: *much too short (veel te kort)* (on the left of the scale) to 10: *much too long (veel te lang)* (on the right of the scale). The scale of questions **Q2**, **Q3** and **Q6** ranged from 1: *very bad (heel slecht)* to 10: *very good (heel goed)*. The scale of question **Q4** ranged from 1: *definitely not (absoluut niet)* to 10: *definitely yes (ja, absoluut)*. The scale of question **Q7** ranged from 1: *totally useless (helemaal onbruikbaar)* to 10: *very useful (erg bruikbaar)*. Figures B.1 until B.7 in Appendix B show screen shots of the computer interface used in the test.

Questions **Q1**, **Q2**, **Q3**, **Q4**, **Q5** and **Q7** aimed at directly testing hypothesis **H1** to **H6**. Question **Q6** aimed at directly testing the general hypothesis **H0**.

For each preview, after answering the seven above-mentioned questions, participants could enter detailed comments.

After seeing the last preview, for each of the four video items used in the test, participants were asked whether they had already seen the video item within the last 6 months, more than 6 months ago, only partially or never before. Additionally they were also asked whether they liked that type of content (e.g. the genre). The answer to this question was a multiple choice in a 1-10 scale with 1: *definitely not (absoluut niet)* and 10: *definitely yes (ja, absoluut)*.

At the end of the test, participants could report general remarks and comments about the previews or the test.

All tests were done using a personal computer (see Appendix B for screen shots of the interface used in the test) with head-phones, in a dedicated room during working hours. A photo of the test set-up is shown in Figure 6.2. The experiment leader was available during the test for answering clarifying questions.

Participants were allowed to do the experiment at their own pace and convenience. The test sessions lasted on average 35 minutes (median: 32, min: 25 , max: 80, standard deviation: 9).

6.6 Results

Figure 6.3 shows the mean scores of the three algorithms across all the subjects for all the questions but **Q1**. The higher the number, the better the quality. For question **Q1** the scale is shifted to $[-5, +5]$ and numbers near zero indicate a better quality. The mean scores for **Q1** were -0.4 for *manually made* previews, -1.2 for *optimized*, and -1.6 for *subsample*.

It is evident that for all the questions, the *manually made* previews stand out as better with respect to *optimized* and *subsample*. Apparently also the *optimized*

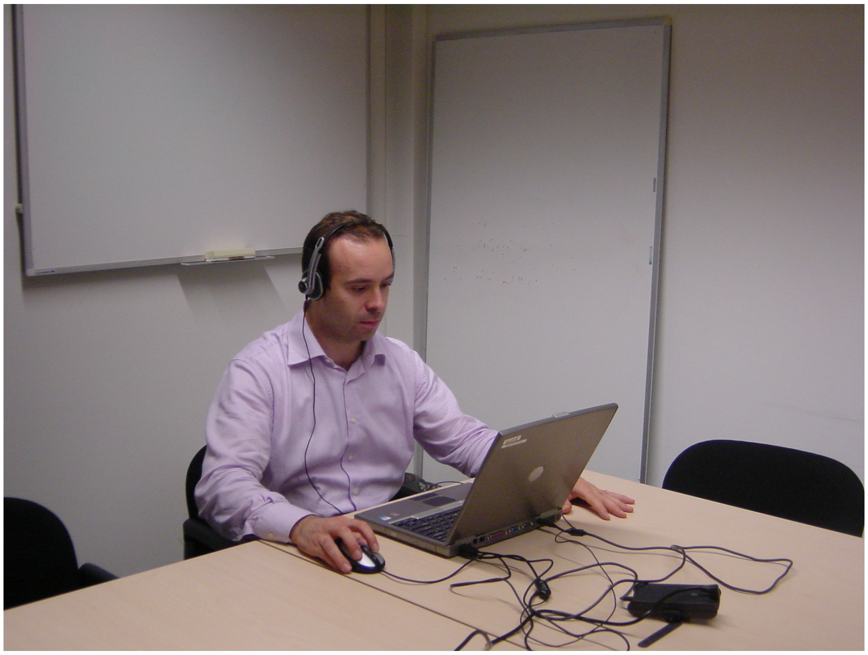


Figure 6.2. User study setup.

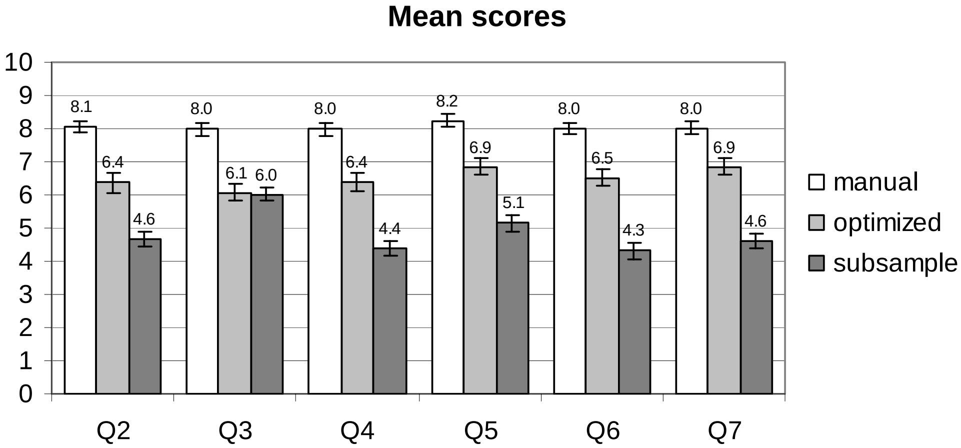


Figure 6.3. Mean scores for questions **Q2**–**Q7**. Error bars show average ± standard error. Question **Q1** has a different scale and its mean scores are shown in Figure 6.4

previews score on average better than *subsample*. The differences in quality between *optimized* and *subsample* are substantial in all questions but **Q3** where the difference is irrelevant.

To verify whether the differences between the three algorithms are statistically significant, for each of the seven questions we conducted an ANOVA (ANalysis Of VAriance) with repeated measures in which *algorithm* and *content* were treated as within-subject independent variables. The scores for the questions were dependent variables. Additionally, *subject* was treated as a random factor. We investigated main effects for *algorithm*, *content*, and *subject*, and the interaction between *algorithm* and *content*.

In the following paragraphs we provide the results of the statistical analysis for each question.

6.6.1 Q1: average segments duration

The mean scores for question **Q1**, *how was the average duration of the fragments*, for the three algorithms and the different video items are plotted in Figure 6.4. The scale goes from -5 to $+5$ where negative numbers mean that users indicated that the segments were too short and positive numbers mean that users found the segments too long. Numbers near zero indicate good average durations of the segments.

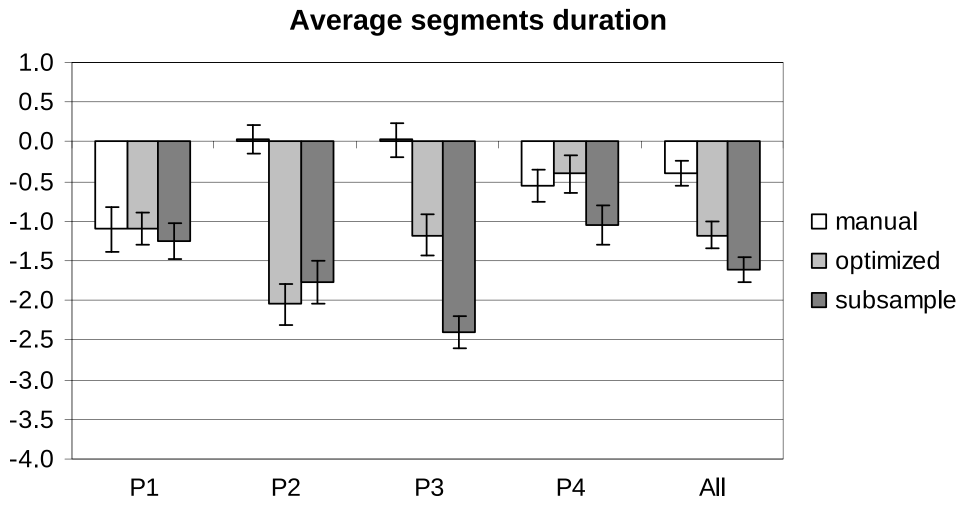


Figure 6.4. Mean scores for question **Q1**, *how was the average duration of the fragments*. In the horizontal axis the four video items. “All” represents the mean across all the video items. Error bars show average ± standard error.

For video item P1, the *manually made* preview scores as good as the *optimization-based* one. This is probably due to a sequence of very short action segments inserted at the end of the preview by the human expert. Many users commented that such sequence suits very well the genre of P1 (action) and enriches the preview. Many other users, instead, found the sequence of very short segments irritating to watch and, for this reason, gave a low score to question **Q1**. For the other video items, the manually made previews score considerably better than the automatically made ones.

From the analysis of variance, a significant main effect for *algorithm* was found ($F = 28.597$, $p < 0.001$). Averaging over *content*, participants indicated that the average segment duration of the *manually made* previews was much closer to zero (not too long, neither too short) than the average segment duration of previews generated by our *optimization-based* approach or using the *subsample* algorithm.

The ANOVA analysis only tells us that there are statistically significant differences among the means of the scores of the three algorithms. It does not tell us which of the means are significantly different from the others, or if the three means are all different from one another. To verify if hypothesis **H1** is valid, it is necessary to perform an additional *post-hoc* test. We performed a Tukey test on the independent variable *algorithm*. The results show that the mean score for *manual* differs significantly from the mean scores for *optimized* and *subsample*. Additionally, the difference between the mean scores for *optimized* and *subsample* are also statistically significant.

Hypothesis **H1** is thus confirmed with a high degree of statistical significance.

Furthermore, a significant main effect for *content* was found ($F = 5.967$, $p = 0.001$): there is a statistically significant difference among the mean scores for the four video items. Predictably, also a significant main effect for *subject* was found ($F = 4.028$, $p < 0.001$): obviously, there are statistically significant differences between the mean scores of different subjects.

6.6.2 Q2: audio transitions

The mean scores for question **Q2**, *how were the audio transitions between the fragments*, for the three algorithms and the different videos are shown in Figure 6.5.

For video item P1, the difference between *manual* and *optimized* is much smaller than for the other video items. Similar to question **Q1**, this is probably due to the presence, in the *manual* preview of P1, of a sequence of very short action segments that was not appreciated by many subjects.

From the analysis of variance, a significant main effect for *algorithm* was found ($F = 79.078$, $p < 0.001$). Averaging over *content*, participants indicated that the audio transitions of the *manual* previews were better than the audio transitions of previews generated by *optimization* and *subsample*.

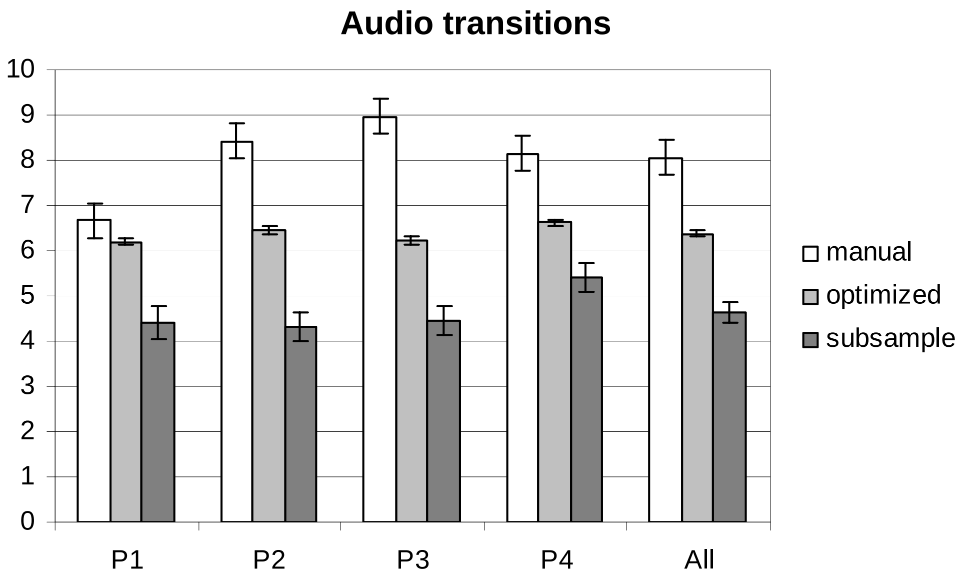


Figure 6.5. Mean scores for question **Q2**, *how were the audio transitions between the fragments*. In the horizontal axis the four video items. “All” represents the mean across all the video items. Error bars show average ± standard error.

We performed a Tukey test on the independent variable *algorithm*. The results show that the mean score for *manual* differs significantly from the mean scores for *optimized* and *subsample*. Additionally, the difference between the mean scores of *optimized* and *subsample* is also statistically significant. Hypothesis **H2** is thus confirmed with a high level of statistical significance.

A significant main effect for *content* was also found ($F = 7.339$, $p < 0.001$): there is a statistically significant difference among the mean scores for the four video items. Predictably, also a significant main effect for *subject* was found ($F = 2.525$, $p < 0.001$): there are statistically significant differences between the mean scores of different subjects.

6.6.3 Q3: visual transitions

The mean scores for question **Q3**, *how were the visual transitions between the fragments*, for the three algorithms and the different video items are shown in Figure 6.6.

Just as in questions **Q1** and **Q2**, for video item P1, the difference between *manual* and *optimized* is much smaller than for the other video items. As for the other previous questions, this is probably due to the presence, in the *manual* preview of P1, of a sequence of very short action segments that many subjects disliked.

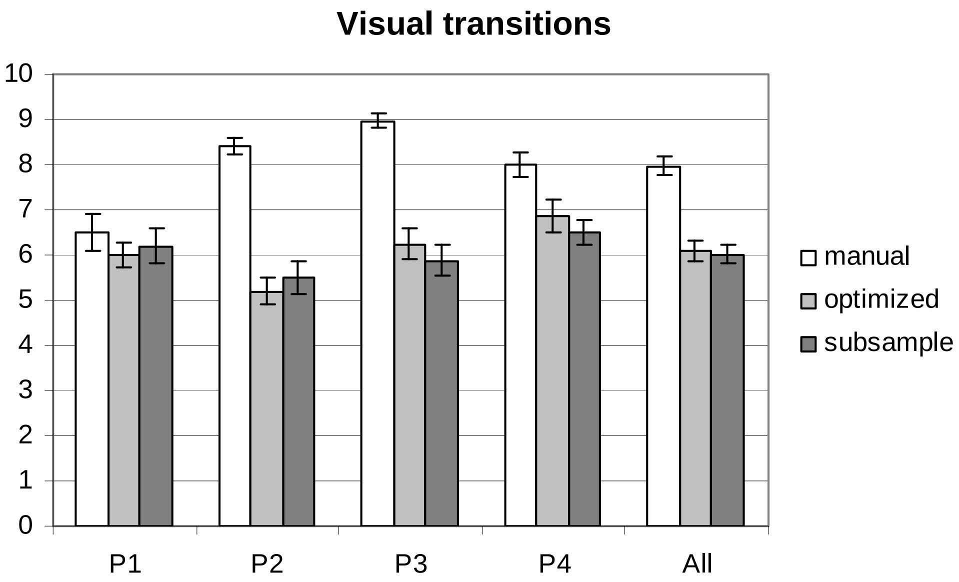


Figure 6.6. Mean scores for question **Q3**, *how were the visual transitions between the fragments*. In the horizontal axis the four video items. "All" represents the mean across all the video items. Error bars show average ± standard error.

From the analysis of variance, a significant main effect for *algorithm* was found ($F = 49.513$, $p < 0.001$). Averaging over *content*, participants indicated that the visual transitions of the *manual* previews were better than the visual transitions of previews generated by *optimization* and *subsample*.

We performed a Tukey test on the independent variable *algorithm*. The results show that the mean score for *manual* differs significantly from the mean scores for *optimized* and *subsample*. Moreover, the difference between the mean scores of *optimized* and *subsample* is not statistically significant. Thus, only the second part of hypothesis **H3** is confirmed with a high level of statistical significance. The quality of visual transitions of previews made using our *optimization-based* approach is not substantially better than of previews made using the *subsample* algorithm. This is not difficult to understand if we consider that both *subsample* and *optimization-based* preview use the same set of candidate video segments resulting from the same segmentation procedure. After *temporal segmentation* and *segment compensation* (see Chapter 5.1), the degree of fulfillment of the *visual continuity* requirements is the same for the *subsample* and the *optimization-based* methods.

A significant main effect for *content* was also found ($F = 8.505$, $p < 0.001$): there is a statistically significant difference among the mean scores for the four video items. Predictably, also a significant main effect for *subject* was found ($F =$

3.384, $p < 0.001$): there are statistically significant differences between the mean scores of different subjects.

6.6.4 Q4: informativeness

The mean scores for question **Q4**, *was the previews informative*, for the three algorithms and the different video items are shown in Figure 6.7.

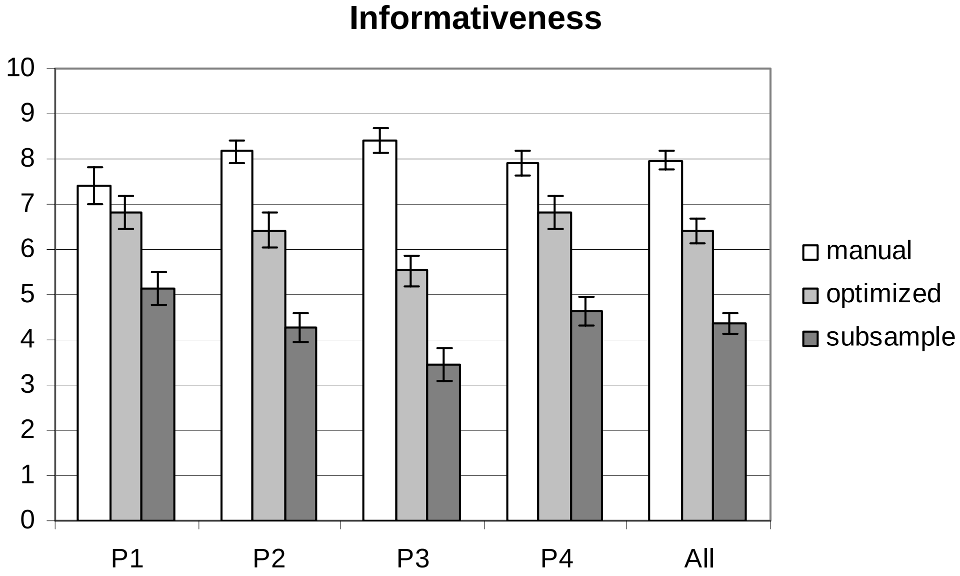


Figure 6.7. Mean scores for question **Q4**, *was the previews informative*. In the horizontal axis the four video items. “All” represents the mean across all the video items. Error bars show average ± standard error.

From the analysis of variance, a significant main effect for *algorithm* was found ($F = 108.887$, $p < 0.001$). Averaging over *content*, participants indicated that the *manual* previews were more informative than the previews generated by *optimized* and *subsample*.

We performed a Tukey test on the independent variable *algorithm*. The results show that the mean score for *manual* differs significantly from the mean scores for *optimized* and *subsample*. Additionally, the difference between the mean scores of *optimized* and *subsample* is also statistically significant. Hypothesis **H4** is thus completely confirmed with a high level of statistical significance.

A significant main effect for *content* was also found ($F = 3.554$, $p = 0.015$): there is a statistically significant difference among the mean scores for the four video items. Predictably, also a significant main effect for *subject* was found ($F = 3.434$, $p < 0.001$): there are statistically significant differences between the mean scores of different subjects.

6.6.5 Q5: atmosphere

The mean scores for question **Q5**, *good impression of the atmosphere*, for the three algorithms and the different video items are shown in Figure 6.8.

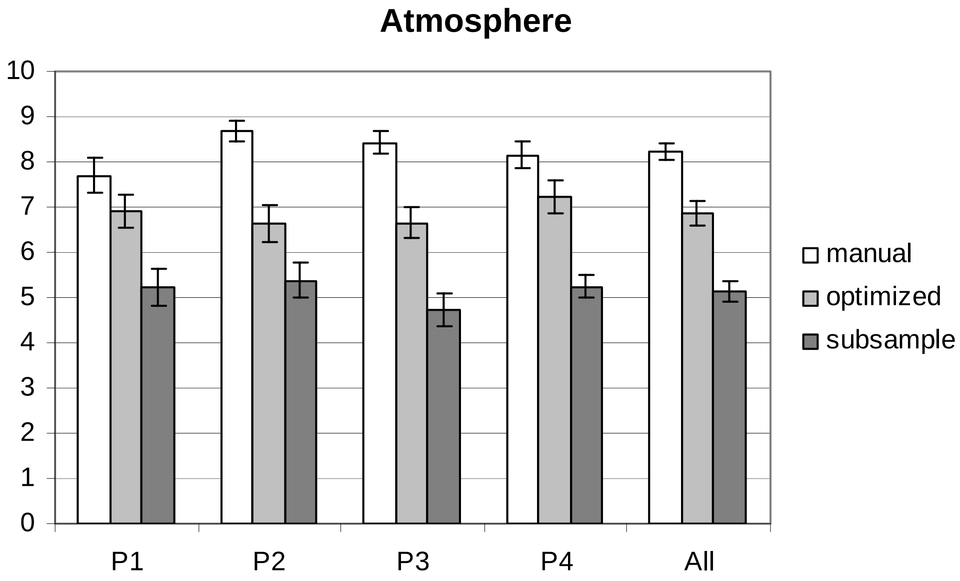


Figure 6.8. Mean scores for question **Q5**, *good impression of the atmosphere*. In the horizontal axis the four video items. “All” represents the mean across all the video items. Error bars show average ± standard error.

From the analysis of variance, a significant main effect for *algorithm* was found ($F = 94.851$, $p < 0.001$). Averaging over *content*, participants indicated that the *manual* previews represented the atmosphere of the original content better than the previews generated by *optimized* and *subsample*.

We performed a Tukey test on the independent variable *algorithm*. The results show that the mean score for *manual* differs significantly from the mean scores for *optimized* and *subsample*. Additionally, the difference between the mean scores of *optimized* and *subsample* is also statistically significant. Hypothesis **H5** is thus completely confirmed with a high level of statistical significance.

Given the very different genres of the content items used, we would expect our method to perform differently for the various genres. Action segments seem also easier to detect and include than the emotional moments typical of drama. Surprisingly, however, no significant main effect for *content* was found ($F = 0.871$, $p = 0.456$): there is no statistically significant difference among the mean scores for the four video items. The differences in representation of the atmosphere for previews generated by the three algorithms are *content-independent*.

Predictably, a significant main effect for *subject* was found ($F = 4.141$, $p < 0.001$). There are statistically significant differences between the mean scores of different subjects.

6.6.6 Q6: overall

The mean scores for question **Q6**, *overall, how good is the overview of the movie given by this preview*, for the three algorithms and the different video items are shown in Figure 6.9.

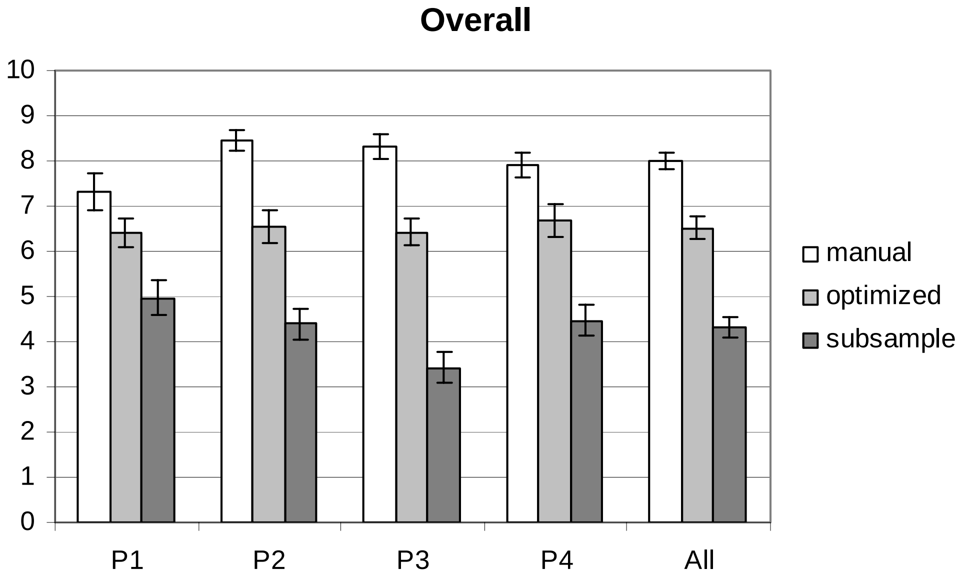


Figure 6.9. Mean scores for question **Q6**, *overall, how good is the overview of the movie given by this preview*. In the horizontal axis the four video items. “All” represents the mean across all the video items. Error bars show average ± standard error.

From the analysis of variance, a significant main effect for *algorithm* was found ($F = 103.245$, $p < 0.001$). Averaging over *content*, participants indicated that the *manual* previews were overall better than the previews generated by *optimized* and *subsample*.

We performed a Tukey test on the independent variable *algorithm*. The results show that the mean score for *manual* differs significantly from the mean scores for *optimized* and *subsample*. Additionally, the difference between the mean scores of *optimized* and *subsample* is also statistically significant. Hypothesis **H0** is thus confirmed completely by our study with a high level of statistical significance.

No significant main effect for *content* was found ($F = 1.199$, $p = 0.31$): there is no statistically significant difference among the mean scores for the four video items. Overall, the quality difference among the algorithms perceived by the subjects does not depend on the content.

Additionally, also a significant main effect for *subject* was found ($F = 2.635$, $p < 0.001$). There are statistically significant differences between the mean scores of different subjects.

6.6.7 Q7: usefulness

The mean scores for question **Q7**, *how useful would this preview be for choosing to watch this movie*, for the three algorithms and the different video items are shown in Figure 6.10.

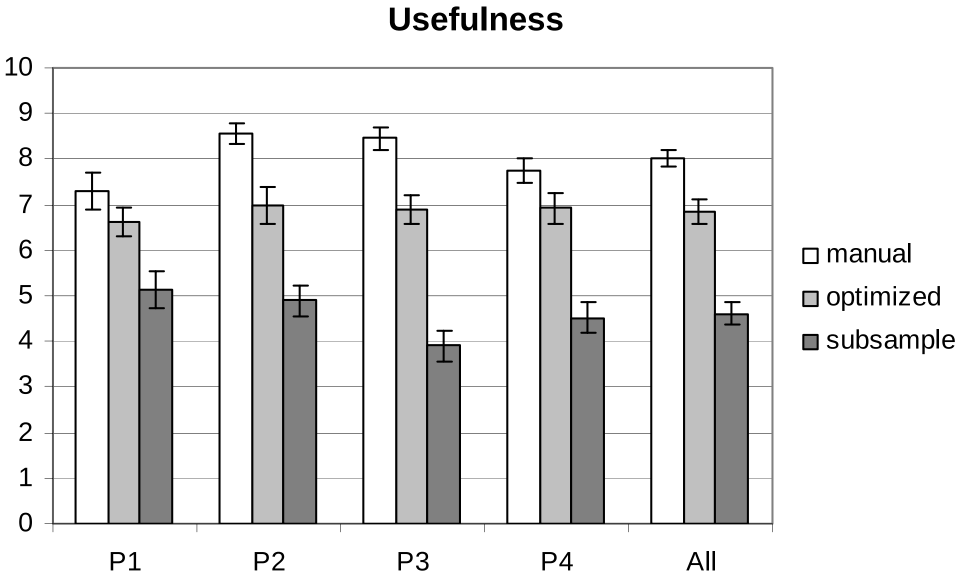


Figure 6.10. Mean scores for question **Q7**, *How useful would this preview be for choosing to watch this movie*. In the horizontal axis the four video items. “All” represents the mean across all the video items. Error bars show average ± standard error.

From the analysis of variance, a significant main effect for *algorithm* was found ($F = 95.183$, $p < 0.001$). Averaging over *content*, participants indicated that the *manual* previews were more useful than the *optimized* and *subsample* previews.

We performed a Tukey test on the independent variable *algorithm*. The results show that the mean score for *manual* differs significantly from the mean scores for *optimized* and *subsample*. Additionally, the difference between the mean scores of *optimized* and *subsample* is also statistically significant. Hypothesis **H6** is thus completely confirmed with a high level of statistical significance.

No significant main effect for *content* was found ($F = 1.7$, $p = 0.167$): there is no statistically significant difference among the mean scores for the four video items. The differences in usefulness perceived by the users among the three algorithms are content-independent.

Predictably, also a significant main effect for *subject* was found ($F = 3.355$, $p < 0.001$). There are statistically significant differences between the mean scores of different subjects.

6.6.8 Other effects

To check whether knowing or liking the content is of influence in the evaluation of the previews, for each of the seven questions we conducted an ANOVA (ANalysis Of VAriance) with repeated measures in which *algorithm* and *content* were treated as within-subject independent variables. Additionally we treated *language*, *gender*, *age*, *filmfan*, *liking*, *knowing* as additional factors.

To simplify the statistical analysis, we mapped the factor *age* to four categories (subjects younger than 25, subjects between 26 and 30, subjects between 31 and 35, and subjects older than 35), the factor *liking* to a four values scale, and the factor *knowing* to two categories (subjects who have never seen the content, and subjects who have seen it in the past at least partially).

No significant main effects were found for any of the seven questions for the variables *language*, *gender*, *age*, *liking*, and *knowing*. An interesting conclusion is that liking the type of content did not influence judging a preview.

We would expect that subjects who know, at least partially, the content, might be stricter in judging the previews. Surprisingly, this is not the case. Knowing already the content is not of influence in judging the previews generated by the three algorithms.

Only one significant main effect for *filmfan* was found for question **Q7** ($F = 5.986$, $p = 0.015$). Subjects who watch more than one film per week scored on average 0.04 points higher than subjects who watch movies less often. We can conclude that film fans tend to find previews slightly more useful than people who do not watch many films.

6.6.9 Comments of the participants

Every participant was invited to write explicit comments regarding each preview or any aspect of the test. Unfortunately the subjects left only 84 comments (2.2 comments per subject on average). We clustered them into 15 categories and we counted the number of comments referring to each algorithm for each category.

Table 6.2 shows all the categories, the number of comments for each category and for each algorithm.

The category with the highest number of comments is related to whether the previews provide enough information on the story line. Not surprisingly, the highest number of negative comments related to this aspect belongs to previews generated using the *subsample* algorithm, while the highest number of positive comments goes to *manually made* previews. Previews generated using the *optimization-based* approach receive an equal number of positive and negative comments related to providing information on the story line.

Many comments refer to the atmosphere conveyed by the preview. *Optimization-based* and *manually made* previews have only positive comments while subjects more often comment that previews generated using the *subsample* method give a bad impression of the atmosphere.

A few subjects comment that *manually made* and *optimization-based* previews give sometimes too much information away.

A few negative comments, that can be summarized as missing logical links between segments, are made referring to previews generated using the *subsample* algorithm.

Table 6.2. Comments categories and number of comments per algorithm sorted by total number of comments.

	Manual	**Optimized**	**Subsample**
Not enough information on the story	0	4	9
Good information on the story	7	4	1
Good impression of the atmosphere	5	6	1
Too short segments	6	0	3
Presence of uninformative scenes	0	0	7
Bad impression of the atmosphere	0	0	6
Gives away too much information	3	2	1
Missing link between segments	0	0	5
Good transitions	2	1	0
Too many action scenes	1	1	1
Bad visual transitions	0	1	1
Bad audio transitions	0	0	2
Not enough action scenes	0	0	2
Preview too short	1	0	0
Not useful for choosing	0	0	1

6.7 Conclusions

From the user study we can conclude that previews generated using our *optimization-based* approach are not as good as manually made previews but have considerably higher quality than previews generated using the *subsample* method.

The differences are statistically significant with a high degree of significance. The amount of perceived difference between the previews depends on the content for most of the quality aspects tested. Furthermore, knowing or liking the content is not of influence in judging the differences in quality of the previews.

Based on the results of the test we can therefore confirm the main hypothesis: our *optimized-based* method provides a *higher quality overview* of a video item than the *subsample* method.

7

Conclusions and suggestions for future work

In the first section of this chapter we highlight the main contribution of this thesis and present our conclusions. Based on these, in the second section, we suggest directions for future work.

7.1 Conclusions

In this thesis we use various audiovisual content analysis techniques for the creation of an algorithmic approach to generating *previews* of narrative-based videos.

We have elicited user needs with respect to *video previews* by analyzing related literature on video summarization and film production and by interviewing end-users, experts and practitioners in the field of video editing and multimedia. To allow fast and convenient content selection, a video preview should take into account more than thirty requirements that can be divided into seven categories: *duration*, *continuity*, *priority*, *uniqueness*, *exclusion*, *structural*, and *temporal order*. *Duration* requirements deal with the durations of the preview and its subparts. *Continuity* requirements consider the smoothness of the flow between preview segments. *Priority* requirements indicate which content should be included in the preview to maximize the amount of conveyed information. *Uniqueness* requirements aim at maximizing the efficiency of the preview by minimizing redundancy. *Exclusion* requirements indicate which content should not be included in the preview. *Structural* requirements are concerned with the structural properties of video, while *temporal order* requirements set the order of the sequences included in the preview.

Based on these requirements, we have introduced a formal model of video summarization specialized for the generation of video previews. The basic idea is to translate the requirements into *score functions*. Each score function is defined to have a non-positive value if a requirement is not met, and to increase depending on the degree of fulfillment of the requirement. A global *objective function* is then defined that combines all the score functions and the problem of generating a preview is translated into the problem of finding the parts of the initial content that maximize the objective function.

Our solution approach is based on two main steps (see Figure 5.1 on page 58): *preparation* and *selection*. In the *preparation* step, the raw audiovisual data is analyzed and segmented into basic elements that are suitable for being included in a preview. The segmentation of the raw data is based on a shot-cut detection algorithm. In the *selection* step various content analysis algorithms are used to perform scene segmentation, advertisements detection and to extract numerical descriptors of the content that, introduced in the objective function, allow to estimate the quality of a video preview. The core part of the *selection* step is the *optimization* step that consists in searching the set of segments that maximizes the objective function in the space of all possible previews. Instead of solving the optimization problem exactly, an approximate solution is found by means of a local search algorithm using *simulated annealing*.

We have performed a numerical evaluation of the quality of the solutions generated by our algorithm with respect to previews generated randomly or by selecting segments uniformly in time. The results on thirty content items have shown that the local search approach outperforms the other methods. However, based on this evaluation, we cannot conclude that the degree of fulfillment of the requirements achieved by our method satisfies the end-user needs completely.

To validate our approach and assess end-user satisfaction, we conducted a user evaluation study in which we compared six aspects of previews generated using our algorithm to human-made previews and to previews generated by subsampling. The results have shown that previews generated using our optimization-based approach are not as good as manually made previews, but have higher quality than previews created using subsample. The differences between the previews are statistically significant.

Finally, this thesis provides answers to the four research questions presented in Section 1.1.1 of the Introduction. The first question is related to how media production knowledge can be used to model narrative video content aiming at automatic summarization. The answer to this question is represented by Chapter 5 in which media production knowledge is exploited in the algorithms for content evaluation. The second question deals with the users' requirements with respect to video previews. This question is directly and thoroughly answered in Chapter 3. The third research question asks which approach to adopt for creating efficient video previews. In Chapter 4 we presented a formal approach, based on the users' requirements and our knowledge of media production, that is suitable for creating very efficient video previews. The fourth research question demands an evaluation of the results from the users' point of view. Our answer is the evaluation study presented in Chapter 6.

7.2 Suggestions for future work

We identify five main directions of research along which the work presented in this thesis could be taken further:

1. Content analysis and understanding
2. Content composition and augmentation
3. Other genres
4. Personalization
5. Validation

In the next sections we describe these five directions and introduce related new research ideas originating from this thesis.

7.2.1 Content analysis and understanding

The basic elements of our summarization technique are visual shots. We also perform audio segmentation in order to discover speech segments and include entire spoken sentences in the preview. Besides this precaution, however, our method does not take further advantage of audio segmentation. To improve the quality of previews, and perhaps to allow the composition of shorter previews, one could think of *decoupling* visual and audio segmentation.

The basic idea would be to combine more freely video segments with speech segments. Speech segments convey rich information about the story line. Once they are detected, they could be associated to images that complement visually what is said and not necessarily to their original visual segments.

The research challenge lies into finding the right visual segments to associate to certain spoken sentences. This requires a semantic understanding of the movie images and its associated dialogues.

To this end, and as an additional suggestion for future work, the textual information contained in the subtitles track of a video item can be processed to discover the narrative structure of a movie. In [Tsoneva et al., 2007] we show how to create summaries that preserve the story line by processing subtitles and movie scripts and using a method of ranking segments that is inspired by web search algorithms.

In a complementary way, video summarization could benefit not only by increasing the level of semantic understanding, but also by better exploiting the syntactic elements used in video. A core part of this thesis is concerned with the application of knowledge of how film is produced to generate video previews (see e.g. Chapter 5). For example, a clue inspired by knowledge of film production is related to the so-called *rack focus* shots. In film production "rack focus" refers to changing the focus during a shot (usually rapidly) from foreground to background or vice versa. This shift in focus is done during a dramatic moment and it is a very powerful indication of a part of content that can convey some important information and therefore should be included in a video preview. An example of rack focus sequence is shown in Figure 7.1.

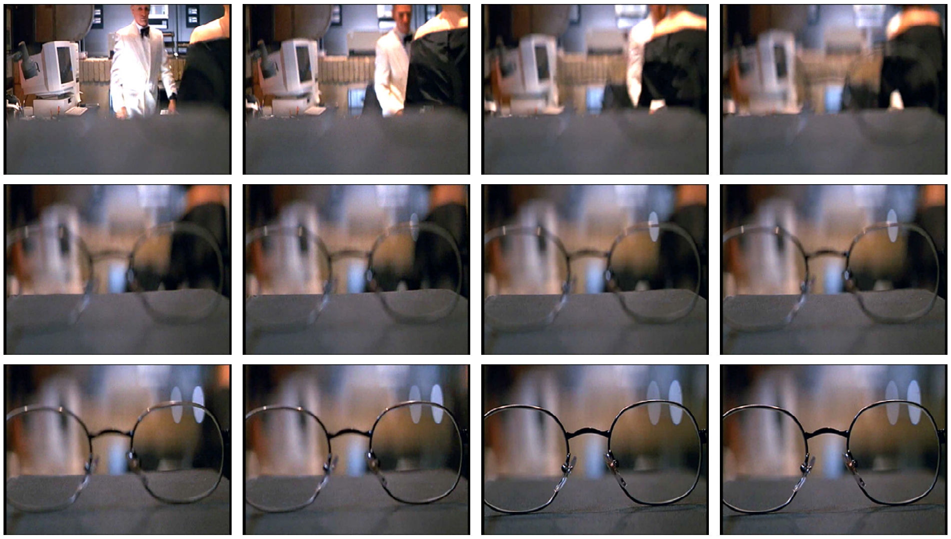


Figure 7.1. Example of a rack-focus shot. From the frame in the top-left corner to the frame in the bottom-right corner, the camera focus changes progressively from the persons in the background to the glasses in the foreground. The shot lasts 4 seconds.

Another relevant film grammar rule is the so-called *Dutch camera angle* (see Figure 7.2). A Dutch angle is a tilted camera angle that is used to emphasize scenes of high emotional content and strong mental stress. Director Joseph Mascelli explains the use of the Dutch angle in cinematography as follows:

"In Hollywood studio parlance a "Dutch" angle is a crazily tilted

> camera angle, in which the vertical axis of the camera is at an angle to the vertical axis of the subject. This results in tilting of the screen image, so that it slopes diagonally, off balance. It is usually used for sequences with weird, violent unstable, impressionistic or other novel effects. A player who has lost his equilibrium, or is drunk, or in highly emotional state, may be shown in a tilted shot. A man-made or natural catastrophe, such as an accident, fire, earthquake may employ tilted camera angles for conveying violence, or topsy-turvy, out-of-this-world effects to the audience. A quiet, statically filmed, slowly paced sequence in an art museum, for instance, could suddenly be thrown into uncontrolled pandemonium by sudden insertion of a tilted shot of a man racing through a doorway crying "Fire!" The remainder of the sequence could employ a series of tilted shots to portray the panic of the trapped museum visitors." [Mascelli, 1965]

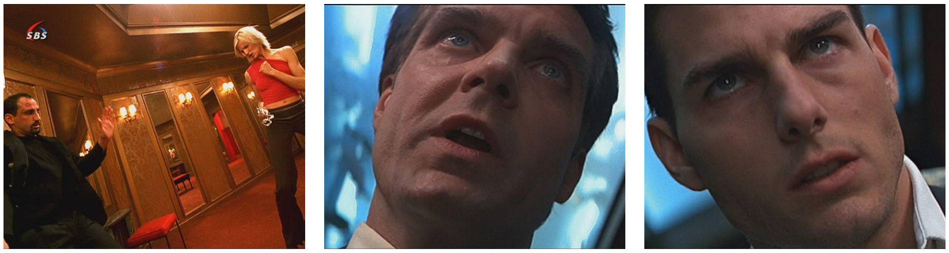


Figure 7.2. Examples of Dutch angle shots.

These type of cinematic effects could be automatically detected and used for further understanding the movie structure, and finding emotional moments.

7.2.2 Content composition and augmentation

One of the main differences between automatically generated video previews and professionally-made trailers, is that trailers often have an additional sound track consisting of music and a voice-over that tells salient facts about the content or a brief summary of the story-line.

Our preview generation method could be extended with the addition of a component that generates a synthetic voice over (using speech synthesis technology) to be mixed to the original audio track. The voice over commentary can be generated from a textual summary associated to the video item (e.g. taken from the electronic program guide), or by processing other textual metadata such as genre, actors names, or year of production.

The challenge is in properly mixing the audio from the original content with the synthetic voice commentary. For example, the synthetic voice over should be in the foreground in action segments with music, but not with segments that have already speech or dialogues. The duration of the synthetic spoken sentences needs to match the selected segments or, in other words, the segment selection step needs to take into consideration the synthetic voice-over properties.

Another way of enhancing a video preview is by influencing its *atmosphere*. *Atmosphere* is a very relevant aspect for characterizing content items [Visser, 2005] and music and sound effects play an important role in determining it. The preview of a content item could be augmented by using, as background music, a representative part of the original audio track. This requires analyzing the audio track for determining representative music parts.

A last, but not least important challenge in preview composition is the automatic creation of summaries that have a logical and consistent story line. One idea for generating summaries around a story line is to use human-made textual synopsis of video items as a base for the content selection.

7.2.3 Other genres

The method presented in this thesis has been developed for *narrative* content, mainly for movies and documentaries. Although movies and documentaries form a considerable part of the video offering available to users, the need for previewing other genres, such as educational and user-created videos, is very high. Recent advances in digital network technology and increased broadband access have created a huge number of new channels for video distribution. However, without adequate means for previewing content, many video productions remain unknown to the majority of the public. Additionally, there is a boom of user-created content that is seldom properly tagged and often of obscure nature. Many websites have recently appeared offering the possibility to upload and freely access thousands of user-created videos.

Research-wise the challenge lies in developing a method that is general enough to be applicable to a large variety of genres. Alternatively, the method should be versatile and easily tunable to accommodate the requirements and characteristics of multiple genres.

7.2.4 Personalization

People like to be addressed individually. In summarization, the notion of personalized summaries can become an important aspect. For example, a movie mainly containing action scenes could also have a poignant love story embedded in it. Persons who particularly like love stories might like previews highlighting these love story elements. Users will require summaries to be personalized so that they can choose the movie they like to watch and not miss out on a movie because the preview did not include sections that might appeal to them more.

Any of the requirements of duration, continuity, priority, uniqueness, exclusion, structural, and temporal order, that were presented in Chapter 3, can be subject to personalization. For example, a user might desire to see more of the introduction segments, which will then affect the structural requirements. The priority requirements can also be based on a user profile: for a person who prefers “dark”, “silent” scenes, we should include those as opposed to “bright”, “dialogue” scenes.

So far the user preferences on summarization have not been fully explored by the research community. An exploration panel of experts and users on issues of multimedia summarization indicated in general that summaries should be personalized [Agnihotri et al., 2003]. For what concerns previews for the purpose of making a selection among a large collection of available content, users indicated in our requirements validation study [Visser, 2005] that previews should include scenes that might be shocking. In this way they can more easily decide not to watch the entire content.

As with any personalization, the problem is twofold: to have an extensive good profile that reflects the user’s needs, and to have an accurate model for performing the computational matching of the user profile to the video analysis features. The challenge here is to ask the “right” questions in order to generate this user profile. One approach is to pose this as a problem of learning from examples where users would be shown many previews and would need to select the one that appeals to them the most. Once the system is trained to the type of previews that a user likes, the different weights contained in the formal model presented in Chapter 4 can be worked out in order to generate personalized previews. However a question arises whether changing the weights is sufficient to influence the segment selection step (see Section 5.1) in order to generate a preview that is really perceived as personalized.

7.2.5 Validation

Another research direction is the evaluation of video preview generation techniques with respect to the task for which they are developed. In our specific case, the challenge lies in assessing the usefulness and effectiveness of automatically generated video previews in selecting programmes in large video archives with respect to, for example, watching trailers, reading electronic program guide descriptions, or following a recommendation.

To perform such an evaluation it would be necessary to embed the video preview generation algorithm into a system with hundreds of different video items. The system should be used in a realistic scenario in which users have to make a selection using one of the tools we want to compare (previews, trailers, electronic program guide, etc.). Questions could be asked before and after users have made a selection to check the level of satisfaction for each tool. Perhaps the main issue in setting-up such an experiment is the creation of a realistic scenario. Although hard-disk recorders are rather popular products, only a few brands and models (e.g. TiVo) have the characteristics that resembles those we considered in our hypothetical scenario: high storage capacity, many content sources, and automatic recording based on a user profile. This poses a challenge in testing video preview generation technology against other methods of content selection.

An interesting aspect that could be tested is whether video previews can help users in discovering content that they would not consider based on textual descriptions or genre metadata. Other questions involve the presentation of previews in user interfaces for video selection and content browsing.

Bibliography

AARTS, E.H.L., AND J.K LENSTRA [1997], *Local Search in Combinatorial Optimization*, Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons, Chichester, England.

ABDEL-MOTTALEB, M., AND A. ELGAMMAL [1999], Face detection in complex environment from color imaging, *Proceedings of the International Conference on Image Processing (ICIP 1999)*.

ADAMI, N., AND R. LEONARDI [2001], Evaluation of different descriptors for identifying similar video shots, *Proceedings of the IEEE Conference on Multimedia and Expo (ICME2001)*, Tokyo, Japan, 948–951.

ADAMS, B., C. DORAI, AND S. VENKATESH [2002], Toward automatic extraction of expressive elements from motion pictures: Tempo, *IEEE Transactions on Multimedia* **4**, 472–481.

ADJEROH, D.A., I. KING, AND M.C. LEE [1998], Video sequence similarity matching, *Proceedings of the International Workshop on Multimedia Information Analysis and Retrieval, MINAR98*, Honk Kong, 80–95.

AGNIHOTRI, L. [2005], *Multimedia Summarization and Personalization of Structured Video*, Ph.D. thesis, Columbia University, New York, USA.

AGNIHOTRI, L., K. DEVARA, T. MCGEE, AND N. DIMITROVA [2001], Summarization of video programs based on closed captioning, *Proceedings of the SPIE Conference on Storage and Retrieval in Media Databases*, San Jose, USA, 599–607.

AGNIHOTRI, L., AND N. DIMITROVA [1999], Text detection for video analysis, *Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries (CBIVL 99)*, 109–113.

AGNIHOTRI, L., N. DIMITROVA, AND J. KENDER [2004], Design and evaluation of a music video summarization system, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan.

AGNIHOTRI, L., N. DIMITROVA, J.R. KENDER, AND J. ZIMMERMAN [2003], Study on requirement specifications for personalized multimedia summarization, *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2003)*, Baltimore, USA, IEEE.

AGNIHOTRI, L., N. DIMITROVA, AND M. SOLETIC [2002], Multilayered video-text extraction method, *Proceedings of IEEE Conference on Multimedia and Expo (ICME 2002)*, Lausanne, Switzerland.

AIZAWA, K., K.-I. ISHIJIMA, AND M. SHIINA [2001], Summarizing wearable video, *Proceedings of the International Conference on Image Processing (ICIP 2001)*, 7–10.

AMERICANA, MAGAZINE [2001], *The Making of a Movie Trailer*, www.americanpopularculture.com.

ANER, A., AND J.R. KENDER [2002], Video summaries through mosaic-based shot and scene clustering, *Proceedings of the European Conference on Computer Vision*, Denmark.

BABAGUCHI, N., Y. KAWAI, AND T. KITAHASHI [2001], Generation of personalized abstract of sports video, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2001)*, Tokyo, Japan, 800–803.

BAILEY, R.W. [1996], *Human Performance Engineering* (Third Edition), Prentice Hall, Upper Saddle River, USA.

BARBIERI, M. [2001], *Advanced Search and Retrieval for Home Multimedia Databases*, Nat.Lab. Report 7170, Philips Research, Eindhoven, The Netherlands.

BARBIERI, M., N. DIMITROVA, AND L. AGNIHOTRI [2005], *Method of video indexing*, WO2006092765, Patent application.

BARBIERI, M., AND E. STINSTRA [2006], *Automatically generated video previews: user study*, Human Information Processing Colloquium, Philips Research, Eindhoven, The Netherlands.

BENINI, S., P. MIGLIORATI, AND R. LEONARDI [2007], Hidden markov models for video skim generation, *Proceedings of the IEEE International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2007)*, Santorini, Greece.

BIMBO, A. DEL [1999], *Visual Information Retrieval*, Morgan Kaufmann Publishers.

BLOCK, B. [2001], *The Visual Story - Seeing the Structure of Film, TV, and New Media*, Focal Press.

BORECZKY, J., A. GIRGENSOHN, G. GOLOVCHINSKY, AND S. UCHIHASHI [2000], An interactive comic book presentation for exploring video, *Proceedings of the ACM International Conference on Computer Human Interaction (CHI 2000)*, The Hague, The Netherlands, 185–192.

CDDB, GRACENOTE [2007], *The CD Database*, www.gracenote.com.

ČERNÝ, V. [1985], A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm, *Journal of Optimization Theory and Applications* **45**, 41–51.

CHEN, H.-W., J.-H. KUO, W.-T. CHU, AND J.-L. WU [2004], Action movies segmentation and summarization based on tempo analysis, *Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval (MIR 2004)*, New York, USA, ACM Press, 251–258.

CHIP, W. [2005], Kryder's law, *Scientific American*.

CHIU, P., A. GIRGENSOHN, W. POLAK, E. RIEFFEL, AND L. WILCOX [2000], A genetic algorithm for video segmentation and summarization, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2000)*, 1329–1332.

CHRISTEL, M.G., A.G. HAUPTMANN, H.D. WACTLAR, AND T.D. NG [2002], Collages as dynamic summaries for news video, *Proceedings of the ACM International Conference on Multimedia*, Juan-les-Pins, France, 561–569.

CONNELL, J., A.W. SENIOR, A. HAMPAPUR, Y.-L. TIAN, L. BROWN, AND S. PANKANTI [2004], Detection and tracking in the ibm peoplevision system, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan.

DAGTAS, S., T. MCGEE, AND M. ABDEL-MOTTALEB [2000], Smartwatch: An automated video event finder, *Proceedings of the ACM International Conference on Multimedia*, Los Angeles, USA.

DEMENTHON, D., V. KOBLA, AND D. DOERMANN [1998], Video summarization by curve simplification, *Proceedings of the sixth ACM International Conference on Multimedia*, Bristol, UK.

DIMITROVA, N., S. JEANNIN, J. NESVADBA, T. MCGEE, L. AGNIHOTRI, AND G. MEKENKAMP [2002], Real time commercial detection using mpeg features, *Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2002)*, Annecy, France.

DIMITROVA, N., J. MARTINO, L. AGNIHOTRI, AND H. ELENBAAS [1999], Color superhistograms for video representation, *Proceedings of the IEEE International Conference on Image Processing (ICIP 1999)*, Kobe, Japan.

DIMITROVA, N., T. MCGEE, AND H. ELENBAAS [1997], Video keyframe extraction and filtering: A keyframe is not a keyframe to everyone, *Proceedings of ACM Conference on Information and Knowledge Management*.

DIMITROVA, N., J. NESVADBA, T. MCGEE, S. JEANNIN, AND G. MEKENKAMP [2001], *Commercial Detection using Features from Empire Encoder*, Technical Report Philips Research USA - TR-2001-017, Philips Research, Briarcliff Manor, USA.

DIMITROVA, N., J. ZIMMERMAN, A. JANEVSKI, L. AGNIHOTRI, N. HAAS, AND R. BOLLE [2003], Content augmentation aspects of personalized entertainment experience, *Proceedings of the Ninth International Conference on User Modeling, TV'03: the 3rd Workshop on Personalization in Future*

TV (UM), Johnstown, USA, 42–51.

DING, W., G. MARCHIONINI, AND D. SOERGEL [1999], Multimodal surrogates for video browsing, *Proceedings of the ACM Conference On Digital Libraries*, Berkeley, USA, 85–93.

DORAI, C., AND S. VENKATESH [2002], *Media Computing - Computational Media Aesthetics*, The Kluwer International Series in Video Computing, Kluwer Academic Publishers.

DORAI, C., AND S. VENKATESH [2003], Bridging the semantic gap with computational media aesthetics, *IEEE Multimedia* **10**, 15–17.

DUDA, R.O., P.E. HART, AND D.G. STORK [2002], *Pattern Classification*, Wiley-Interscience.

DVB [2007], *Digital Video Broadcasting Project (DVB)*, www.dvb.org.

EBADOLLAHI, S., S.-F. CHANG, AND H. WU [2002], Echocardiogram videos: Summarization, temporal segmentation, and browsing, *Proceedings of the IEEE International Conference on Image Processing (ICIP 2002)*, Rochester, USA, 613–616.

EFFELSBERG, W., R. JAIN, AND R. LIENHART [2000], Visualgrep: A systematic method to compare and retrieve video sequencies, *Multimedia Tools and Applications* **10**, 47–72.

EISENSTEIN, S. [1975], *The Film Sense*, Harcourt Brace & Company.

EKIN, A., MURAT TEKALP A., AND R. MEHROTRA [2003], Automatic soccer video analysis and summarization, *Proceedings of the International Conference on Electronic Imaging*, Santa Clara, USA, 339–350.

ERNST, F., H. WEDA, M. BARBIERI, AND S. DE WAELE [2005], *Method and apparatus for determining the shot type of an image*, WO2007036823, Patent application.

EROL, B., D.-S. LEE, AND J. HULL [2003], Multimodal summarization of meeting recordings, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2003)*, Baltimore, USA, 25–28.

FARIN, D., W. EFFELSBERG, AND P.H.N. DE WITH [2002], Robust clustering-based video-summarization with integration of domain-knowledge, *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2002)*, Lausanne, Switzerland, 89–92.

FERRER, M.Z., M. BARBIERI, AND H. WEDA [2006], Automatic classification of field-of-view in video, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2006)*, Toronto, Canada.

FURHT, B., S.W. SMOLIAR, AND H. ZHANG [1995], *Video and Image Processing in Multimedia Systems*, The Kluwer International Series in Engineering and Computer Science - Multimedia Systems and Applications, Kluwer Academic Publishers.

GERSHO A., GRAY R.M. [1992], *Vector Quantization and Signal Conversion*, The Kluwer International Series in Engineering anf Computer Science - Communications and Information Theory, Kluwer Academic Publishers.

GONG, Y., AND X. LIU [2001], Summarizing video by minimizing visual content redundancies, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2001)*, Tokyo, Japan.

GUTTERSWIJK, J. [2004], *How do people choose their TV programme - TV evening programming,* Technical note, Philips Research, Eindhoven, The Netherlands.

HAHN, UDO, AND INDERJEET MANI [2000], The challenges of automatic summarization, *IEEE Computer* **33**, 29–36.

HANJALIC, A. [2003], Multimodal approach to measuring excitement in video, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2003)*, Baltimore, USA, 289–292.

HANJALIC, A. [2006], Extracting moods from pictures and sounds, *IEEE Signal Processing Magazine* **23**, 90–100.

HANJALIC, A., AND L.-Q. XU [2005], Affective video content representation and modeling, *IEEE Transactions on Multimedia* **7**, 143–154.

HANJALIC, A., AND H.-J. ZHANG [1999], An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis, *IEEE Transactions on Circuits and Systems for Video Technology* **9**, 1280–1289.

HAUPTMANN, A., AND M. SMITH [1995], Text, speech, and vision for video segmentation: The informedia project, *AAAI Fall 1995 Symposium on Computational Models for Integrating Language and Vision*.

HAYKIN, S. [1999], *Neural networks,* Prentice-Hall.

HE, L., E. SANOCKI, A. GUPTA, AND J. GRUDIN [1999], Auto-summarization of audio-video presentations, *Proceedings of the ACM International Conference on Multimedia*, Orlando, USA, 489–498.

HERMES, T., AND C. SCHULTZ [2007], Automatic generation of hollywood-like movie trailers, *Image: Journal of Interdisciplinary Image Science* **1**.

HUANG, J., S.R. KUSNAR, M. MITRA, R. ZABIH, AND W.-J. ZHU [1999], Spatial color indexing and applications, *International Journal of Computer Vision* **35**, 245–268.

IMDB [2007], *Internet Movie Database,* www.imdb.com.

IONESCU, B., P. LAMBERT, D. COQUIN, L. OTT, AND V. BUZULOIU [2006], Animation movies trailer computation, *Proceedings of the ACM International Conference on Multimedia*, Santa Barbara, USA, 631–634.

IRANI, M., AND P. ANANDAN [1998], Video indexing based on mosaic representations, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **5**, 905–921.

JASINSCHI, R., N. DIMITROVA, T. MCGEE, L. AGNIHOTRI, AND J. ZIMMERMAN [2001], Video scouting: An architecture and system for the integration of multimedia information in personal TV applications, *Proceedings of the IEEE International Conference on Audio and Speech Signal Processing (ICASSP 2001)*, Salt Lake City, USA, 1405–1408.

JTC1/SC29/WG11, ISO/IEC [2000], *MPEG-7 Visual part of eXperimentation Model*, Standard MPEG, N3321, ISO/IEC JTC1/SC29/WG11, Noordwijkerhout, The Netherlands.

JUNG, B., T. KWAK, J. SONG, AND Y. LEE [2004], Narrative abstraction model for story-oriented video, *Proceedings of the ACM International Conference on Multimedia*, New York, USA, 828–835.

KAROUTCHI, D. [2003], *Video summarization and video posters creation based on clustering algorithm and color feature*, Nat.Lab. Technical Note PR-TN-2003/00550, Philips Research Europe, Eindhoven, The Netherlands.

KIM, J.-G., H.S. CHANG, K. KANG, M. KIM, J. KIM, AND H.-M. KIM [2003], Summarization of news video and its description for content-based access, *International Journal of Imaging Systems and Technology* **13**, 267–274.

KIRKPATRICK, S., C.D. GELATT, AND M.P. VECCHI [1983], Optimization by simulated annealing, *Science* **220**, 671–680.

KOHONEN, T. [1997], *Self-Organizing Maps*, Springer-Verlag.

KOLEKAR, M.H., AND S. SENGUPTA [2006], Event-importance based customized and automatic cricket highlight generation, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2006)*, Toronto, Canada, 1617–1620.

LEE, J.K., J.H. OH, AND S. HWANG [2006], Scenario based dynamic video abstractions using graph matching, *Proceedings of the ACM International Conference on Multimedia*, Hilton, Singapore, 810–819.

LEW, M.S. [2001], *Principles of Visual Information Retrieval*, Springer.

LI, B., AND M.I. SEZAN [2001], Event detection and summarization in sports video, *Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'01)*, Kauai, USA.

LI, D., I.K. SETHI, N. DIMITROVA, AND T. MCGEE [2000], Classification of general audio data for content-based retrieval, *Pattern Recognition Letters* **22**, 533–544.

LI, Y., W. MING, AND C.-C.J. KUO [2001], Semantic video content abstraction based on multiple cues, *Proceedings of the International Conference on Multimedia and Expo (ICME 2001)*, Tokyo, Japan, 804–808.

LIENHART, R. [1999], Comparison of automatic shot boundary detection algorithms, *Proceedings of Storage and Retrieval for Image and Video*

Databases VII, San Jose, USA, 290–301.

LIENHART, R. [2000], Dynamic video summarization of home video, *Proceedings of the SPIE Conference on Storage and Retrieval for Media Databases*, San Jose, USA.

MA, Y.-F., L. LU, H.-J. ZHANG, AND M. LI [2002], A user attention model for video summarization, *Proceedings of the ACM International Conference on Multimedia*, Juan les Pins, France, 533–542.

MANI, I. [2001], Summarization evaluation: An overview, *Proceedings of the Second NTCIR Workshop on Research in Chinese and Japanese Text Retrieval and Text Summarization*, Tokyo, Japan.

MANJUNATH, B.S., P. SALEMBIER, AND T. SIKORA (EDITORS) [2002], *Introduction to MPEG-7*, John Wiley & Sons.

MARTELLO, S., AND P. TOTH [1990], *Knapsack problems: Algorithms and Computer Implementations*, John Wiley & Sons.

MASCELLI, J.V. [1965], *The Five C's of Cinematography - Motion Pictures Filming Techniques*, Silman-James Press, Los Angeles, USA.

MATLIN, M.W., AND H.J. FOLEY [1997], *Sensation and Perception*, Allyn & Bacon.

MCKINNEY, M., AND J. BREEBAART [2003], Features for audio and music classification, *Proceedings of the 4^{th} International Symposium on Music Information Retrieval (ISMIR 2003)*, Washington DC, USA.

MEKENKAMP, G., M. BARBIERI, B. HUET, I. YAHIAOUI, B. MERIALDO, R. LEONARDI, AND M. ROSE [2002], Generating TV summaries for CE-devices, *Proceedings of the 2002 ACM International Conference on Multimedia*, Juan les Pins, France, 83–84.

MERIALDO, B., K.T. LEE, D. LUPARELLO, AND J. ROUDAIRE [1999], Automatic construction of personalized TV news programs, *Proceedings of the 1999 ACM International Conference on Multimedia*, Orlando, USA, 323–331.

MERLINO, A., AND M. MAYBURY [1999], *An Empirical Study of the Optimal Presentation of Multimedia Summaries of Broadcast News*, Chapter in Advances in Automatic Text Summarization, 391–341. MIT Press.

NAM, J. [1999], Video abstract of video, *Proceedings of the IEEE Workshop on Multimedia Signal Processing*, Copenhagen, Denmark.

NIST [2007], *TRECVid*, www-nlpir.nist.gov/projects/trecvid/.

OMOIGUI, N., L. HE, A. GUPTA, J. GRUDIN, AND E. SANOCKI [1999], Time-compression: Systems concerns, usage, and benefits, *Proceedings of the ACM International Conference on Computer Human Interaction (CHI 1999)*, Pittsburgh, USA.

OVER, P., A.F. SMEATON, AND P. KELLY [2007], The TRECVID 2007 BBC rushes summarization evaluation pilot, *Proceedings of the TRECVID Video Summarization Workshop (TVS 2007)*, Augsburg, Germany, 1–15.

PASS, G., AND R. ZABIH [1996], Histogram refinement for content-based image retrieval, *Proceedings of the Third IEEE Workshop on Applications of Computer Vision (WACV'96)*, Sarasota, USA, 96–102.

PASS, G., AND R. ZABIH [1999], Comparing images using joint histograms, *Multimedia Systems* **7**, 234–240.

PAULUSSEN, I., M. BARBIERI, AND G. MEKENKAMP [2003], The Spation project: Embedding content analysis in consumer electronics networks, *Proceedings of the Third International Workshop on Content-Based Multimedia Indexing (CBMI 2003)*, Rennes, France.

PEKER, K.A., A. DIVAKARAN, AND T.V. PAPATHOMAS [2001], Automatic measurement of intensity of motion activity of video segments, *Proceedings of Storage and Retrieval for Media Databases 2001*, SPIE.

PEKER, K.A., I. OTSUKA, AND A. DIVAKARAN [2006], Broadcast video program summarization using face tracks, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2006)*, Toronto, Canada, 1053–1056.

PEKER, K. A., A. DIVAKARAN, AND H. SUN [2001], Constant pace skimming and temporal sub-sampling of video using motion activity, *Proceedings of the IEEE International Conference on Image Processing (ICIP 2001)*, Thessaloniki, Greece, 414–417.

PETKOVIC, M., V. MIHAJLOVIC, AND W. JONKER [2002], Multi-modal extraction of highlights from TV Formula 1 programs, *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2002)*, Lausanne, Switzerland.

PFEIFFER, S., R. LIENHART, S. FISCHER, AND W. EFFELSBERG [1996], Abstracting digital movies automatically, *Journal of Visual Communication and Image Representation* **7**, 345–353.

PHILLIPS, W.H. [1999], *Film – An Introduction*, Bedford St. Martin's, USA.

QIAN, R.J., M.I. SEZAN, AND P.J.L. VAN BEEK [2000], Image retrieval using blob histograms, *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2000)*, New York, USA, 125–128.

RUBIN, M. [1992], *NonLinear: A Guide to Electronic Film and Video Editing*, Playground Productions, Los Angeles, USA.

RUSSELL, D.M. [2000], A design pattern-based video summarization technique: Moving from low-level signals to high-level structure, *Proceedings of the IEEE Hawaii International Conference on System Sciences*, Maui, USA.

SAARELA, J., AND B. MERIALDO [1999], Using content models to build audio-video summaries, *Proceedings of Storage and Retrieval for Image and Video*

Databases VII, San Jose, USA, 338–347.

SALEMBIER, P., AND J. SMITH [2001], MPEG-7 multimedia description schemes, *IEEE Transactions on Circuits and Systems for Video Technology* **11**, 748–759.

SALTON, G., AMIT SINGHAL, MANDAR MITRA, AND CHRIS BUCKLEY [1997], Automatic text structuring and summarization, *Information Processing and Management* **33**, 193–207.

SCHAFFER, D., L. AGNIHOTRI, N. DIMITROVA, T. MCGEE, AND S. JEANNIN [2002], Improving digital video commercial detectors with genetic algorithms, *Proceedings of the Genetic and Evolutionary Computation Conference 2002*, New York, USA.

SILVA, G.C. DE, T. YAMASAKI, AND K. AIZAWA [2005], Evaluation of video summarization for a large number of cameras in ubiquitous home, *Proceedings of the ACM International Conference on Multimedia*, Hilton, Singapore, 820–828.

SMEATON, A.F., P. OVER, AND W. KRAAIJ [2006], Evaluation campaigns and TRECVID, *Proceedings of the Multimedia Information Retrieval Workshop (MIR 2006)*, Santa Barbara, USA, 321–330.

SMYTH, B., AND P. COTTER [2000], A personalized TV listening service for the digital TV age, *Knowledge-Based Systems* **13**, 53–59.

SUNDARAM, H., L. XIE, AND S.-F. CHANG [2002], A utility framework for the automatic generation of audio-visual skims, *Proceedings of the ACM International Conference on Multimedia*, Juan les Pins, France.

SYEDA-MAHMOOD, T., AND D. PONCELEON [2001], Learning video browsing behavior and its application in the generation of video previews, *Proceedings of the ACM International Conference on Multimedia*, Ottawa, Canada, 119–128.

TANIGUCHI, Y., A. AKUTSU, AND Y. TONOMURA [1997], Panoramaexcerpts: extracting and packing panoramas for video browsing, *Proceedings of the ACM International Conference on Multimedia*, Seattle, USA, 427–436.

TOKLU, C., S.-P. LIOU, AND M. DAS [2000], Videoabstract: A hybrid approach to generate semantically meaningful video summaries, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2000)*, 1333–1336.

TSONEVA, T., M. BARBIERI, AND H. WEDA [2007], Automated summarization of narrative video on a semantic level, *Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2007)*, Irvine, USA.

TURETSKY, R., AND N. DIMITROVA [2004], Screenplay alignment for closed-system speaker identification and analysis of feature films, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan, 1659–1662.

UCHIHASHI, S., J. FOOTE, A. GIRGENSOHN, AND J. BORECZKY [1999], Video manga: Generating semantically meaningful video summaries, *Proceedings of the ACM International Conference on Multimedia*, Orlando, USA, 388–392.

UEHARA, K., M. AMANO, Y. ARIKI, AND M. KUMANO [2004], Video shooting navigation system by real-time useful shot discrimination based on video grammar, *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan, 583–586.

VIOLA, P., AND M. JONES [2001], Rapid object detection using a boosted cascade of simple features, *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Kauai, USA, 511–518.

VISSER, A. [2005], *User Study Automatically Generated Movie-in-a-Minute Video, What People Prefer in a Preview*, Technical note, Philips Research Europe.

WACTLAR, H.D, T. KANADE, M.A. SMITH, AND S.M. STEVENS [1996], Intelligent access to digital video: Informedia project, *IEEE Computer* **29**, 46–52.

WANG, T., T. MEI, X.-S. HUA, X.-L. LIU, AND H.-Q. ZHOU [2007], Video collage: A novel presentation of video sequence, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2007)*, Beijing, China, 1479–1482.

WEI, C.-Y., N. DIMITROVA, AND S.-F. CHANG [2004], Color-mood analysis of films based on syntactic and psychological models, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan, 831–834.

XIE, L., L. KENNEDY, S.-F. CHANG, A. DIVAKARAN, H. SUN, AND C.-Y. LIN [2005], Layered dynamic mixture model for pattern discovery in asynchronous multi-modal streams, *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP 2005)*, Philadelphia, USA.

YAHIAOUI, I., B. MERIALDO, AND B. HUET [2001], Generating summaries of multi-episode video, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2001)*, Tokyo, Japan.

YEO, B.-L., AND M.M YEUNG [1997], Video visualization for compact presentation and fast browsing of pictorial content, *IEEE Transactions on Circuits and Systems for Video Technology* **7**, 771–785.

YOKOI, T., AND H. FUJIYOSHI [2006], Generating a time shrunk lecture video by event detection, *Proceedings of the IEEE International Conference on*

Multimedia and Expo (ICME 2006), Toronto, Canada, 641–644.

ZETTL, H. [2001], *Sight Sound Motion - Applied Media Aesthetics* (Third Edition), Wadsworth Publishing Co., Belmont, USA.

ZHONG, D., R. KUMAR R., AND S.-F. CHANG [2001], Real-time personalized sports video filtering and summarization, *Proceedings of the ACM International Conference on Multimedia*, 623–625.

ZHU, X., AND G. PENN [2006], Utterance-level extractive summarization of open-domain spontaneous conversations with rich features, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2006)*, Toronto, Canada, 793–796.

A

Stop word list

The following list of words represents the stop word list used in ranking subtitles as described in Section 5.7.1.

a aah about above according across actually ad adj ain't after afterward again against ah albeit all almost alone along already also although always am among amongst an and another any anybody anyhow anyone anything anyway anywhere apart are aren aren't around as ass at author available away ay

b baby back be became because become becomes becoming been before beforehand begin beginning behind being below beside besides between beyond billion both but buy by

can cannot canst can't caption certain cfrd choose come came coming comes conducted considered contrariwise co copy cos could couldn couldn't

d day described describes designed determine determined did didn didn't different discussed do does doesn doesn't doing don don't dost doth double down dr dual due during

e each eg eight eighty either else elsewhere end ending enough er etc even ever every everybody everyone everything everywhere except excepted excepting exception exclude excluding exclusive f far farther farthest few fifty find first five for former formerly forth forty forward found four free from front fuck fuckin fucking further furthermore furthest

general get give given go god going gonna good got

had halves hardly has hasn hasn’t hast hath have haven haven’t he he’d he’ll hell hello help hence henceforth her here hereabouts hereafter hereby herein here’s hereto hereupon hers herself he’s hey hi him himself hindmost his hither hitherto hm hmm home homepage how however howsoever hundred huh

i id i’d if i’ll im i’m in inasmuch include included including ii indeed indoors information inside insomuch instead int into investigated inward inwards is isn isn’t it its it’s itself i’ve i.e. ie

j join just

kind know knew known knowing knows

l last later latter latterly least less lest let let’s like likely little ll look looked ltd

m made make makes man many may maybe me meantime meanwhile might million miss more moreover most mostly mr mrs ms msie much must my myself

namely need neither never nevertheless new next nine ninety no nobody none nonetheless nope nor not nothing notwithstandig now nowadays nowhere

o obtained of off often oh ohh ok okay on once one one’s only onto ooh or other others otherwise ought our ours ourselves out outside over overall ow own

p page per performance performed perhaps plenty possible present presented presents provide provided provides put quite

rather rd re really recent recently related report required reserved results right ring round

s said sake same sang save saw say see seeing seem seemed seeming seems seen seldom selected selves sent seven seventy several sfrd shalt she she’d she’ll she’s shit should shouldn shouldn’t shown sideways significant since site six sixty slept slew slung slunk smote so some somebody somehow someone something sometime sometimes somewhat somewhere sorry spake spat spoke spoken sprang sprung stave staves still stop studies such supposing

take took taken taking t tell telling told ten test tested text th than that that’ll that’s the thee their them themselves then thence thenceforth there thereabout thereabouts thereafter thereby therefore therein there’ll thereof thereon there’s thereto thereupon these they they they’ll they’re they’ve think thinking thinks thirty this those thou though thousand three thrice through throughout thru thus thy thyself till time together to too toward towards trillion twenty two types uh uhh uk um unable under underneath unless unlike unlikely until up upon upward upwards us use used using

various ve very via

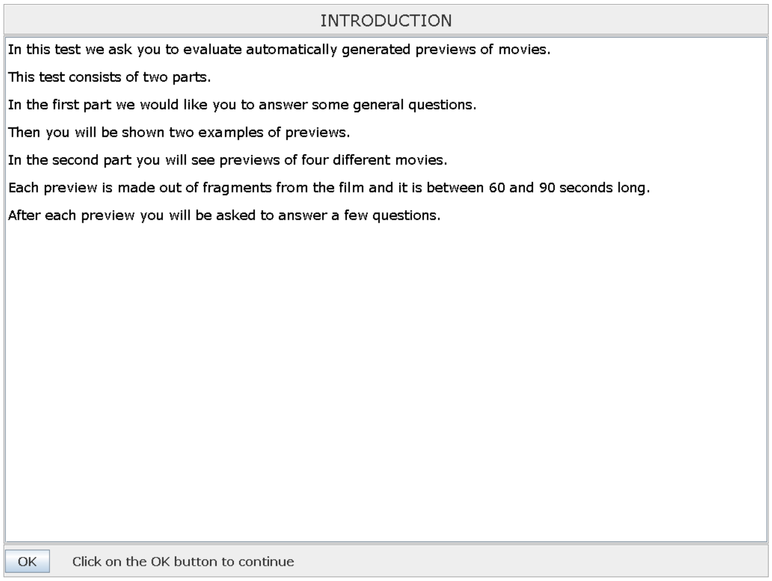
w want was wasn wasn’t way we we’d week welcome well we’ll were we’re weren weren’t we’ve what whatever what’ll what’s whatsoever when whence whenever whensoever where whereabouts whereafter whereas whereat whereby wherefore wherefrom wherein whereinto whereof whereon wheresoever whereto whereunto whereupon wherever wherewith whether whew which whichever whichsoever while whilst whither who whoa who’d whoever whole who’ll whom whomever whomsoever who’s whoo whose whosoever why will wilt with within without won won’t worse worst would wouldn wouldn’t wow

x

y ye yeah year yes yet yippee you you’d you’ll your you’re yours yourself yourselves you’ve

B

User test screen shots



INTRODUCTION

In this test we ask you to evaluate automatically generated previews of movies.

This test consists of two parts.

In the first part we would like you to answer some general questions.

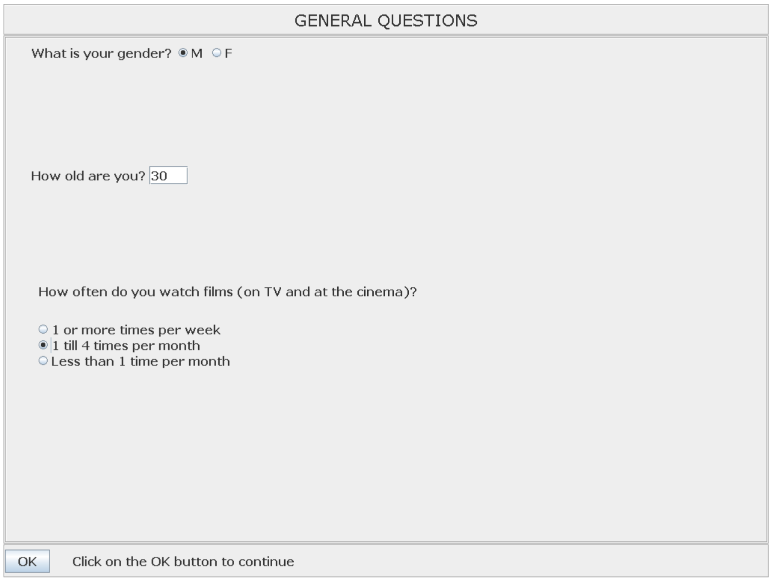
Then you will be shown two examples of previews.

In the second part you will see previews of four different movies.

Each preview is made out of fragments from the film and it is between 60 and 90 seconds long.

After each preview you will be asked to answer a few questions.

Figure B.1. Screen shot of the interface used in the test: introduction screen.



GENERAL QUESTIONS

What is your gender? ● M ○ F

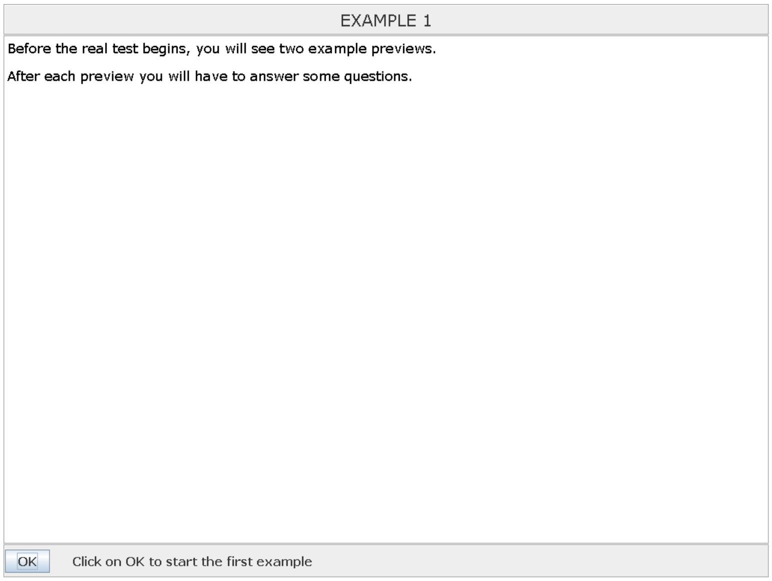
How old are you? 30

How often do you watch films (on TV and at the cinema)?

○ 1 or more times per week
● 1 till 4 times per month
○ Less than 1 time per month

Click on the OK button to continue

Figure B.2. Screen shot of the interface used in the test: general questions.



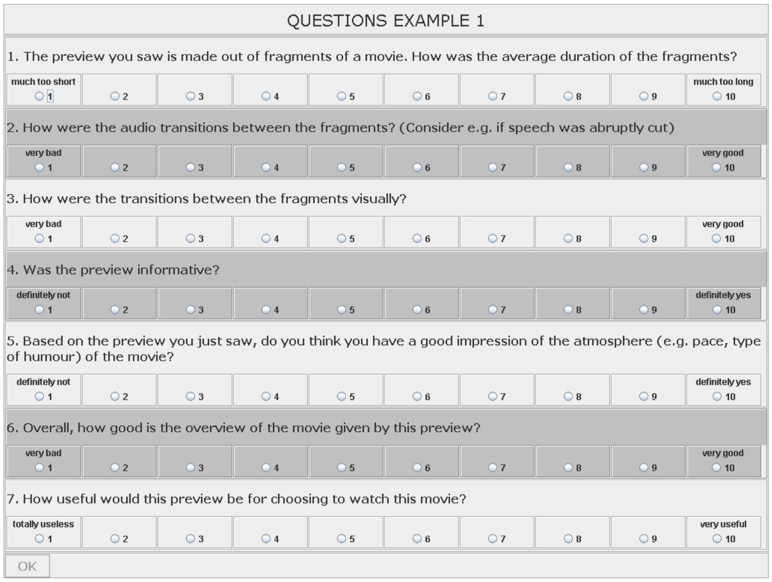
EXAMPLE 1

Before the real test begins, you will see two example previews.

After each preview you will have to answer some questions.

Click on OK to start the first example

Figure B.3. Screen shot of the interface used in the test: intermediate screen shown before the first preview example begins.



QUESTIONS EXAMPLE 1

1. The preview you saw is made out of fragments of a movie. How was the average duration of the fragments?

much too short ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	much too long ○ 10

2. How were the audio transitions between the fragments? (Consider e.g. if speech was abruptly cut)

very bad ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	very good ○ 10

3. How were the transitions between the fragments visually?

very bad ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	very good ○ 10

4. Was the preview informative?

definitely not ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	definitely yes ○ 10

5. Based on the preview you just saw, do you think you have a good impression of the atmosphere (e.g. pace, type of humour) of the movie?

definitely not ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	definitely yes ○ 10

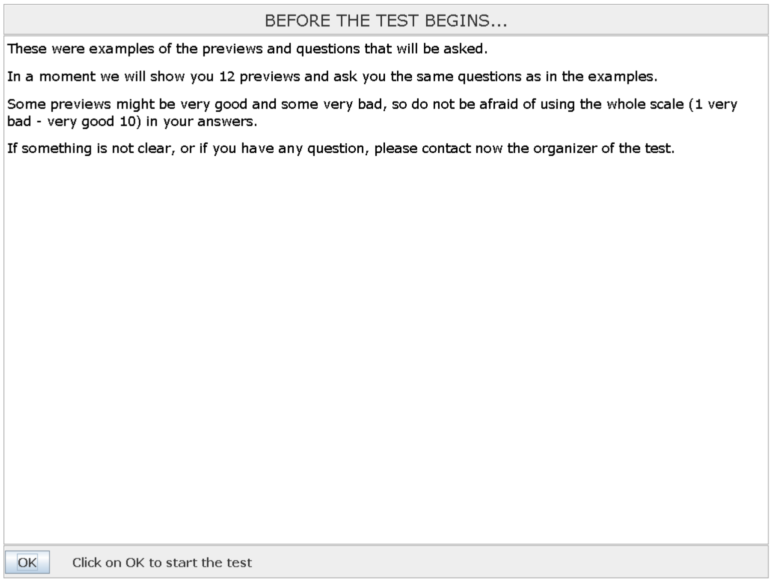
6. Overall, how good is the overview of the movie given by this preview?

very bad ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	very good ○ 10

7. How useful would this preview be for choosing to watch this movie?

totally useless ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	very useful ○ 10

Figure B.4. Screen shot of the interface used in the test: questions related to the first example.



BEFORE THE TEST BEGINS...

These were examples of the previews and questions that will be asked.

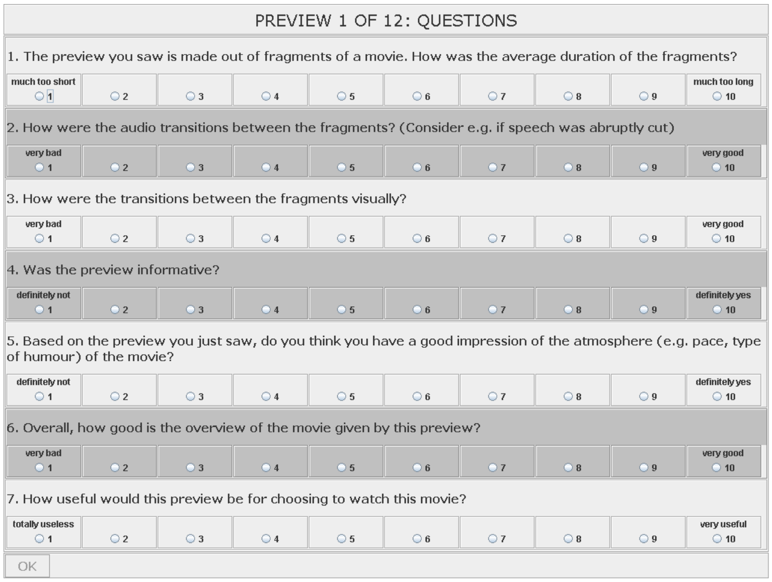
In a moment we will show you 12 previews and ask you the same questions as in the examples.

Some previews might be very good and some very bad, so do not be afraid of using the whole scale (1 very bad - very good 10) in your answers.

If something is not clear, or if you have any question, please contact now the organizer of the test.

Click on OK to start the test

Figure B.5. Screen shot of the interface used in the test: intermediate screen shown before the preview begins.



PREVIEW 1 OF 12: QUESTIONS

1. The preview you saw is made out of fragments of a movie. How was the average duration of the fragments?

much too short ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	much too long ○ 10

2. How were the audio transitions between the fragments? (Consider e.g. if speech was abruptly cut)

very bad ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	very good ○ 10

3. How were the transitions between the fragments visually?

very bad ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	very good ○ 10

4. Was the preview informative?

definitely not ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	definitely yes ○ 10

5. Based on the preview you just saw, do you think you have a good impression of the atmosphere (e.g. pace, type of humour) of the movie?

definitely not ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	definitely yes ○ 10

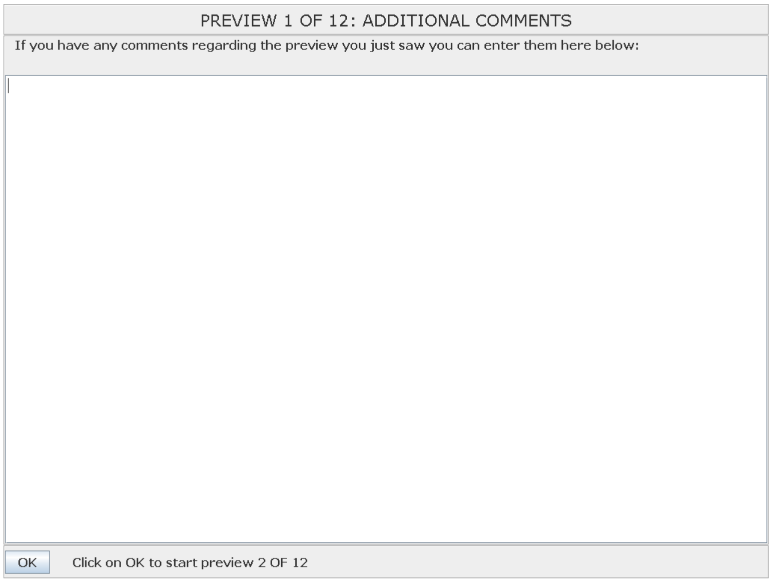
6. Overall, how good is the overview of the movie given by this preview?

very bad ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	very good ○ 10

7. How useful would this preview be for choosing to watch this movie?

totally useless ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	very useful ○ 10

Figure B.6. Screen shot of the interface used in the test: questions related to the preview.

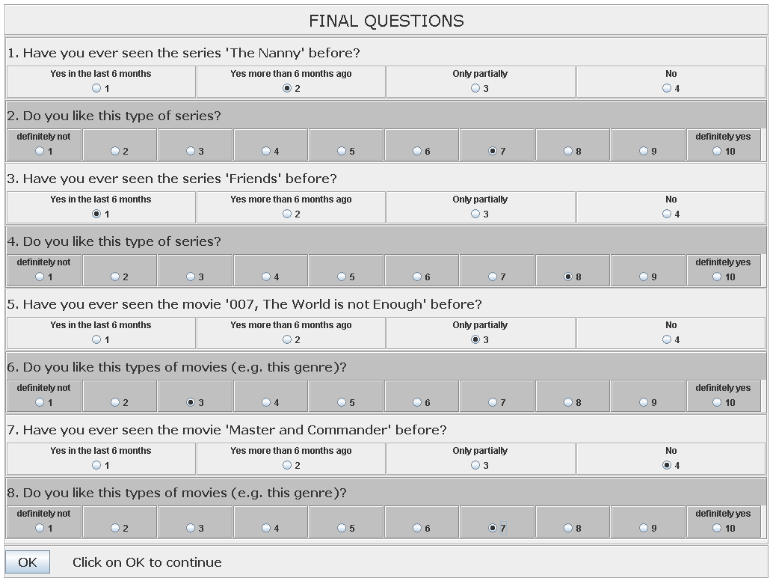


PREVIEW 1 OF 12: ADDITIONAL COMMENTS

If you have any comments regarding the preview you just saw you can enter them here below:

Click on OK to start preview 2 OF 12

Figure B.7. Screen shot of the interface used in the test: additional comments after answering the questions.



FINAL QUESTIONS

1. Have you ever seen the series 'The Nanny' before?

Yes in the last 6 months ○ 1	Yes more than 6 months ago ● 2	Only partially ○ 3	No ○ 4

2. Do you like this type of series?

definitely not ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	● 7	○ 8	○ 9	definitely yes ○ 10

3. Have you ever seen the series 'Friends' before?

Yes in the last 6 months ● 1	Yes more than 6 months ago ○ 2	Only partially ○ 3	No ○ 4

4. Do you like this type of series?

definitely not ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	○ 7	● 8	○ 9	definitely yes ○ 10

5. Have you ever seen the movie '007, The World is not Enough' before?

Yes in the last 6 months ○ 1	Yes more than 6 months ago ○ 2	Only partially ● 3	No ○ 4

6. Do you like this types of movies (e.g. this genre)?

definitely not ○ 1	○ 2	● 3	○ 4	○ 5	○ 6	○ 7	○ 8	○ 9	definitely yes ○ 10

7. Have you ever seen the movie 'Master and Commander' before?

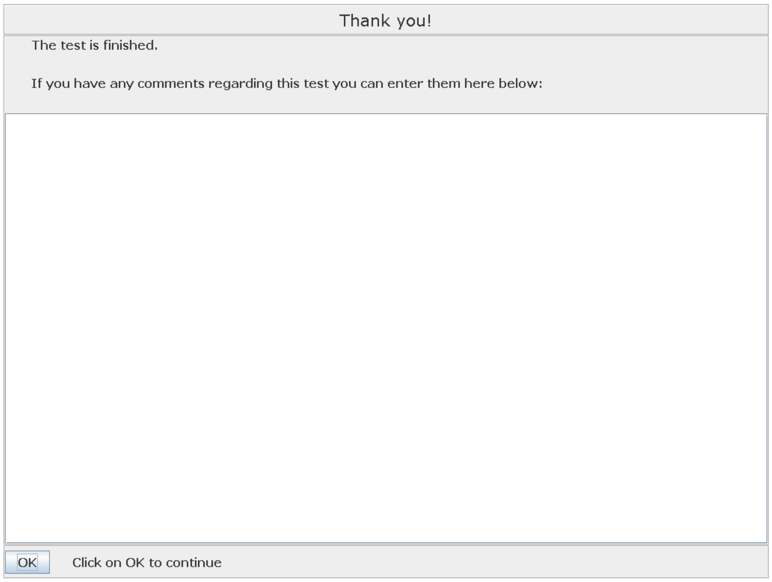
Yes in the last 6 months ○ 1	Yes more than 6 months ago ○ 2	Only partially ○ 3	No ● 4

8. Do you like this types of movies (e.g. this genre)?

definitely not ○ 1	○ 2	○ 3	○ 4	○ 5	○ 6	● 7	○ 8	○ 9	definitely yes ○ 10

Click on OK to continue

Figure B.8. Screen shot of the interface used in the test: final questions.



Thank you!

The test is finished.

If you have any comments regarding this test you can enter them here below:

Click on OK to continue

Figure B.9. Screen shot of the interface used in the test: final comments before the test ends.

Publications

The following publications are related to this thesis.

Papers and book chapters

- T. Tsoneva, M. Barbieri, H. Weda, “Automated summarization of narrative video on a semantic level”, *Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2007),* Irvine, USA, September 2007.
- M. Barbieri, N. Dimitrova, L. Agnihotri, “Movie-in-a-Minute: Automatically Generated Video Previews”, *chapter in the book Intelligent algorithms in ambient and biomedical computing*, edited by W.F.J. Verhaegh, E.H.L. Aarts, J.H.M. Korst, Springer, Philips Research Books Series, 2006, ISBN 1402049536.
- M. Zapata Ferrer, M. Barbieri, H. Weda, “Automatic Classification of Field of View in Video”, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2006)*, Toronto, Canada, July 2006.
- M. Barbieri, H. Weda, N. Dimitrova, “Browsing video recordings using Movie-in-a-minute”, *Proceedings of the IEEE International Conference On Consumer Electronics (ICCE 2006)*, Las Vegas, USA, January 2006.
- M. Barbieri, N. Dimitrova, L. Agnihotri, “Movie-in-a-Minute: Automatically Generated Video Previews”, *Proceedings of the IEEE 5th Pacific-Rim Conference on Multimedia 2004 (PCM2004)*, Tokyo, Japan, December 2004. Published in “Advances in Multimedia Information Processing - PCM 2004: 5th Pacific Rim Conference on Multimedia, Tokyo, Japan, November 30 - December 3, 2004. Proceedings, Part II”, K. Aizawa, Y. Nakamura, S. Satoh (eds.), Lecture Notes in Computer Science, vol. 3332, pp. 9-18, Springer-Verlag Heidelberg.
- M. Barbieri, N. Dimitrova, L. Agnihotri, “Movie-in-a-Minute: Automatically Generated Video Previews”, *Proceedings of the 2nd Philips Symposium on Intelligent Algorithms (SOIA 2004),* Eindhoven, The Netherlands, December 2004.
- N. Dimitrova, L. Agnihotri, M. Barbieri, “Providing Rapid Access to Video Content Through Automatic Summaries”, *chapter in MMIR MultiMedia In-*

formation Retrieval - Metodologie ed esperienze internazionali di content-based retrieval per l'informazione e la documentazione, edited by Roberto Raieli and Perla Innocenti, AIDA, Roma 2004, ISBN 8890114495.

- M. Barbieri, L. Agnihotri, N. Dimitrova, "Video Summarization: Methods and Landscape", *Proceedings of SPIE International Conference on Internet Multimedia Management Systems IV (ITCom 2003)*, Orlando, FL, USA, 7-11 September 2003.
- G. Mekenkamp, M. Barbieri, B. Huet, I. Yahiaoui, B. Merialdo, R. Leonardi, M. Rose, "Generating TV Summaries for CE-devices", *Proceedings of the 2002 ACM International Conference on Multimedia*, Juan les Pins, France, 1-6 December 2002.

Granted patents

- M. Barbieri, "Apparatus for reproducing an information signal stored on a storage medium", US6957387.
- A. Stella, J. Nesvadba, F. Snijder, M. Barbieri, "Estimating signal power in compressed audio", EP1393301.

Patent applications

- Weda H., Barbieri M., "Generation of image data summarising a sequence of video frames", filed in 2007.
- M. Barbieri, H. Weda, "Method and apparatus for automatically generating a summary of a multimedia content item", filed in 2006.
- H. Weda, M. Barbieri, "Method and apparatus for generating a summary", filed in 2006.
- O. Seupel, M. Barbieri, "Data summarization system and method for summarizing a data stream", filed in 2006.
- M. Barbieri, L. Agnihotri, N. Dimitrova, "Method and device for automatic generation of summary of a plurality of images", WO2007099496.
- L. Agnihotri, M. Barbieri, N. Dimitrova, "Automatic generation of trailers containing product placements", WO2006077536.
- M. Barbieri, N. Dimitrova, L. Agnihotri, "Method of video indexing", WO2006092765.
- M. Barbieri, N. Dimitrova, L. Agnihotri, "Summarization of audio and/or visual data", WO2006095292.

- N. Dimitrova, L. Agnihotri, M. Barbieri, “Device for enabling to represent content items through meta summary data, and method thereof”, WO2006134538.
- L. Agnihotri, N. Dimitrova, M. Barbieri, A. Hanjalic, “Synthesis of composite news stories”, WO2006103633.
- S. Gutta, M. Barbieri, “Method and apparatus for pausing a live transmission”, 2005, WO2007036833.
- F. Ernst, H. Weda, M. Barbieri, S. de Waele, “Method and apparatus for determining the shot type of an image”, WO2007036823.
- D. Burazerovic, M. Barbieri, “Processing method and device using scene change detection”, WO2005074297.
- D. Burazerovic, M. Barbieri, “Coding method and corresponding coded signal”, WO2005074296.
- M. Barbieri, D. Burazerovic, “Monochrome frame detection method and corresponding device”, WO2005099273.
- M. Barbieri, D. Burazerovic, “Coding method applied to multimedia data”, WO2005099274.
- M. Barbieri, L. Agnihotri, “Video trailer”, WO2005086471.
- M. Barbieri, L. Agnihotri, “Zoom into video summaries”, WO2005119515.
- M. Barbieri, L. Agnihotri, “Updating video summary”, WO2005119515.
- D. Burazerovic, M. Barbieri, “Method and device for processing coded video data”, WO2006048807.
- M. Barbieri, G. Mekenkamp, “Creating a summarized overview of a video sequence”, WO2006092752.
- F. Snijder, J. Nesvadba, M. Barbieri, “Method and apparatus for similar video content hopping”, WO2004061711.
- M. Barbieri, G. Mekenkamp, “System and method for generating audio-visual summaries for audio-visual program content”, WO2004105035.
- M. Barbieri, G. Mekenkamp, B. Huet, “Method and circuit for creating a multimedia summary of a stream of audiovisual data”, WO2005062610.
- M. Barbieri, “Detecting a content item in a digital video stream”, WO2005009043.
- N. Dimitrova, M. Barbieri, L. Agnihotri, “Method and apparatus to catch up with a running broadcast or stored content”, WO2005103954.
- M. Barbieri, “Video abstracting”, WO2005017899.

- A. Stella, J. Nesvadba, F. Snijder, M. Barbieri, “Silence detection”, WO02093801.
- T. McGee, M. Barbieri, “Method and system for selecting chapter boundaries for digital video recordings”, WO2004042730.
- F. Pessolano, M. Barbieri, “Video abstracting”, WO2004047109.
- M. Barbieri, J. Nesvadba, G. Mekenkamp, M. Ceccarelli, W. Fontijn, R. Tol, “Reproducing apparatus providing a colored slider bar”, WO0221530.
- M. Barbieri, J. Nesvadba, F. Snijder, A. Stella, “Content analysis apparatus”, WO02093928.

Automatic Summarization of Narrative Video

Summary

The amount of digital video content available to users is rapidly increasing. Developments in computer, digital network, and storage technologies all contribute to broaden the offer of digital video. Only users' attention and time remain scarce resources. Users face the problem of choosing the right content to watch among hundreds of potentially interesting offers.

Video and audio have a dynamic nature: they cannot be properly perceived without considering their temporal dimension. This property makes it difficult to get a good idea of what a video item is about without watching it. *Video previews* aim at solving this issue by providing compact representations of video items that can help users making choices in massive content collections. This thesis is concerned with solving the problem of automatic creation of video previews.

To allow fast and convenient content selection, a video preview should take into consideration more than thirty requirements that we have collected by analyzing related literature on video summarization and film production. The list has been completed with additional requirements elicited by interviewing end-users, experts and practitioners in the field of video editing and multimedia. This list represents our collection of user needs with respect to video previews.

The requirements, presented from the point of view of the end-users, can be divided into seven categories: *duration*, *continuity*, *priority*, *uniqueness*, *exclusion*, *structural*, and *temporal order*. *Duration* requirements deal with the durations of the preview and its subparts. *Continuity* requirements request video previews to be as continuous as possible. *Priority* requirements indicate which content should be included in the preview to convey as much information as possible in the shortest time. *Uniqueness* requirements aim at maximizing the efficiency of the preview by minimizing redundancy. *Exclusion* requirements indicate which content should not be included in the preview. *Structural* requirements are concerned with the structural properties of video, while *temporal order* requirements set the order of the sequences included in the preview.

Based on these requirements, we have introduced a formal model of video summarization specialized for the generation of video previews. The basic idea is to translate the requirements into *score functions*. Each score function is defined to have a non-positive value if a requirement is not met, and to increase depending on the degree of fulfillment of the requirement. A global *objective function* is then defined that combines all the score functions and the problem of generating a preview is translated into the problem of finding the parts of the initial content that maximize the objective function.

Our solution approach is based on two main steps: *preparation* and *selection*. In the *preparation* step, the raw audiovisual data is analyzed and segmented into basic elements that are suitable for being included in a preview. The segmentation of the raw data is based on a shot-cut detection algorithm. In the *selection* step various content analysis algorithms are used to perform scene segmentation, advertisements detection and to extract numerical descriptors of the content that, introduced in the objective function, allow to estimate the quality of a video preview. The core part of the *selection* step is the *optimization* step that consists in searching the set of segments that maximizes the objective function in the space of all possible previews. Instead of solving the optimization problem exactly, an approximate solution is found by means of a local search algorithm using *simulated annealing*.

We have performed a numerical evaluation of the quality of the solutions generated by our algorithm with respect to previews generated randomly or by selecting segments uniformly in time. The results on thirty content items have shown that the local search approach outperforms the other methods. However, based on this evaluation, we cannot conclude that the degree of fulfillment of the requirements achieved by our method satisfies the end-user needs completely.

To validate our approach and assess end-user satisfaction, we conducted a user evaluation study in which we compared six aspects of previews generated using our algorithm to human-made previews and to previews generated by subsampling. The results have shown that previews generated using our optimization-based approach are not as good as manually made previews, but have higher quality than previews created using subsample. The differences between the previews are statistically significant.

Acknowledgments

First of all, I express my gratitude to Emile Aarts. The first time we discussed the hypothesis of my enrollment in a Ph.D. was in 2000, when my master studies were yet to come to an end. In 2003, I had already done research for two years when we started meeting regularly to make sure my efforts would converge and eventually lead to a Ph.D. thesis. Thank you Emile, through all these years you gave me excellent guidance, helped me to stay focused and supported me greatly.

A special thanks to Gerhard Mekenkamp, for persuading me to pursue a Ph.D. in addition to my work responsibilities and for giving me freedom and space in his projects during my first years at Philips Research.

I am in debt to Nevenka Dimitrova for her brilliant advice and her dedication in coaching me in spite of the distance and career changes. Nevenka, you not only have given me invaluable suggestions, but you have often provided me with that drive that made me hold on and keep up during hard times. Together with Lalitha Agnihotri you made our collaboration very pleasant and stimulating. I really enjoyed working with you.

I am very grateful to Jan Korst for his rigorous and patient review of the thesis and for meticulously stimulating me in going deeper in the understanding and in providing argumentations.

Thanks to the members of the core committee, Prof.'s Jan Biemond, Lynda Hardman and Arnold Smeulders, for carefully reviewing this thesis and providing excellent feedback.

Many people helped me throughout the research that lead to this thesis. Some of the algorithms could not have been implemented so easily without the software base for MPEG processing and the assistance of Ad Denissen, Igor Paulussen, Erik Niessen and Gerhard Mekenkamp. Peter Jakobs, Peter Sels and Ruud Wijnands developed a video player tailored to my special needs that greatly simplified part of my research. I am thankful to Jeroen Breebaart and Martin McKinney for providing the audio classification software and to Vasanth Philomin for the face detection library.

The analysis of the literature on which Chapter 2 is based was done together with Lalitha Agnihotri and Nevenka Dimitrova. I also thank Lalitha for providing the text detection software.

I am particularly grateful to Jettie Hoonhout for her invaluable cooperation and feedback during the design of the user evaluation studies. I am thankful to Jan Engel for his support in the statistical analysis of the user evaluation results. Thanks to Erwin Stinstra who elegantly solved the combinatorial problem of the user study design allowing me to save precious time and considerable effort. A special thanks to Albertine Visser for her contribution to the user requirements study and for conducting the first user evaluation using the paired-comparison method. Thanks to Robert van Uden for manually creating the previews for the user study and to all the participants who donated some of their time testing and judging the quality of the results of my algorithms.

Special gratitude to Hans Weda for his indispensable advice, the thorough and critical review of the manuscript and the translation of the user study text into Dutch. Thanks to Marco Campanella and Prarthana Shrestha for the useful discussions. Marco, Prarthana and Hans, thanks for your flexibility whenever this thesis took some of your time away. Your enthusiasm and dedication brought new life in our team and helped me speeding up my own work.

Special gratitude to Freddy Snijder for our technical and philosophical discussions during and after work.

I am thankful to Wim Verhaegh for the discussions and his suggestions on the formal model. I am grateful to Gerhard Langelaar, Olaf Seupel and Enno Ehlers for the useful discussions on embedded implementations of content analysis and summarization algorithms.

Thanks to Philips Research and in particular to Eric Persoon, Wouter Leibbrandt and Hans van Gageldonk for supporting the research that lead to this thesis. A very special thanks to all my colleagues at Philips Research. You make Nat.Lab. a stimulating environment where working is a pleasure. I am grateful to you for creating all the conditions for doing scientific research with a pragmatic approach that still values good engineering solutions.

A very special thanks to Andrea Notari, who always stimulated me in pursuing a Ph.D. and helped me persist every time I felt discouraged.

Finally, I want to thank my family for their moral support and for always being there whenever I needed them. Thanks to my sister Maddalena who carefully revised some parts of this thesis.

The last words are for my wife Kathelijne. Not only you have been extremely patient every time this thesis took my time and attention away from you, but you have encouraged me all the way from the very beginning to the end. Now that this manuscript is finished, we can look forward to spending some nice time together.

Curriculum Vitae

Mauro Barbieri was born in 1975 on March 4th, in Cento, Italy. He attended the "ITIS Ugo Bassi" high-school in Cento where he graduated with the final mark of 60/60. From 1995 to 2000 he studied Informatics Engineering at the University of Bologna, Italy. In 2000 he did a six-month internship at the Philips Research Laboratories Eindhoven, The Netherlands, under the supervision of Marco Ceccarelli and Gerhard Mekenkamp. Based on this internship, he wrote his Master's Thesis "Advanced Search and Retrieval for Home Multimedia Databases" and graduated cum laude in 2000 at the University of Bologna. In 2001 he became permanent staff member of the Experience Processing department at the Philips Research Laboratories Eindhoven, where he did his Ph.D. research. Currently he is a Senior Scientist and project leader working on innovative multimedia applications. His research interests include video analysis and summarization, multimedia systems and applications, and human-computer interaction. He has filed over 70 patent applications and his work has been published and presented at several international conferences.