chapter 8

Video Summarization

Cuneyt M. Taskiran and Edward J. Delp

Contents

8.1 Introduction

Deriving compact representations of video sequences that are intuitive for users and let them easily and quickly browse large collections of video data is fast becoming one of the most important topics in content-based video processing. Such representations, which we will collectively refer to as *video summaries*, rapidly provide the user with information about the contents of

0-8493-1526-3/2004/$0.00+$1.50

the particular sequence being examined while preserving the essential message. The need for automatic methods for generating video summaries is fueled both from the user and production viewpoints. With the proliferation of personal video recorder devices and hand-held cameras, many users generate many times more video footage than they can digest. On the other hand, in today's fast-paced news coverage, programs such as sports and news must be processed quickly for production or their value quickly decreases. Such time constraints and the increasing number of services being offered place a large burden on production companies to process, edit, and distribute video material as quickly as possible.

In this chapter we review the current state of the art in automatic video summarization. Summarization, be it for a video, audio, or text document, is a challenging and ill-defined task since it requires the processing system to make decisions based on high-level notions such as the semantic content and relative importance of the parts of the documents with respect to each other. Evaluating resulting summaries is also a problem since it is hard to derive quantitative measures of summary quality. We will examine some of the approaches that were proposed to deal with these problems.

8.1.1 Types of Video Summaries

The goal of video summarization is to process video sequences that contain high redundancy and make them more exciting, interesting, valuable, and useful for users. The properties of a video summary depends on the application domain, the characteristics of the sequences to be summarized, and the purpose of the summary. Some of the purposes that a video summary might serve are listed below.

1. Intrigues the viewer to watch the whole video. Movie trailers, prepared by highly skilled editors and high budgets, are the best examples of this type.
2. Lets the user decide if the whole video is worth watching. Summaries of video programs that might be used in personal video recorders are examples of this category. The user may have watched the episode that was recorded or may have already watched similar content so might not want to watch the program after seeing the summary.
3. Helps the user locate specific segments of interest. For example, in distance learning, students can skip parts of a lecture with which they are familiar and instead concentrate on new material.
4. Lets users judge if the video being examined is relevant to their query. In content-based image database applications, it is customary to return the results of queries as thumbnail images, which can be judged at a glance for relevance to the query. The same task is time consuming for video sequences since search results may contain many long sequences containing hundreds of shots. Presenting the summaries of the results would be much more helpful.

5. Enables users of pervasive devices, such as PDAs, palm computers, and cellular phones, to view video sequences, which these devices otherwise would not be able to handle due to their low processing power. Using summaries also results in significant downloading cost savings for such devices. An example of such an application is the system developed by IBM using annotations based on the MPEG-7 standard [1].
6. Gives the viewer all the important information contained in the video. These summaries are intended to replace watching the whole video. Executive summaries of long presentations would be an example of this type.

The above list, while not complete, illustrates the wide range of different types of video summaries one would like to generate. We observe that for most applications video summaries mainly serve two functions: the indicative function, where the summary is used to indicate what topics of information is contained in the original program; and the informative function, where they are used to cover the information in the source video as much as possible subject to the summary length. These summary applications are not independent, and video summarization applications often will be designed to achieve a mixture of them. Naturally there is no single approach, neither manual nor automatic, that will apply to all types of video summaries.

8.1.2 Terminology

Before proceeding further we will introduce the terminology that will be throughout this chapter. We will use the term *video summarization* to denote any method that can be used to derive a compact representation of a video sequence. Once a summary is obtained after processing the source video it has to be presented to the user. We use the term *summary visualization*, or just visualization, to refer to the method that is used to display the summary. There are two main categories of summary visualizations

1. In static visualization methods a number of representative frames, often called keyframes, are selected from the source video sequence and are presented to the user, sometimes accompanied by additional information such as timestamps and closed caption text [2–8]. We refer to such visualizations as *video abstractions*. This type of summary visualization is sometimes referred to as the *storyboard presentation*.
2. Dynamic visualization methods generate a new, much shorter video sequence from the source video [9–14]. We refer to dynamic visualizations as *video skims*.

Unfortunately, there is little agreement in the literature on the terminology used to describe visualizations. For example, Hanjalic and Zhang define a video abstract as a “compact representation of a video sequence” [15], which denotes a general representation and corresponds to our definition of video summary. On the other hand, Lienhart [11] uses the same term to refer to what we defined to be video skims.

For video skims the duration of the summary is generally specified by the user in terms of the summarization ratio (SR), which is defined as the ratio of the duration of the video skim to the duration of the source video. For video abstracts the user may specify the number of keyframes to be displayed.

The chapter is organized as follows: in section 8.2 we examine the numerous approaches proposed to automatically generate summaries from video programs. Methods used for summary visualization are investigated in section 8.3. Evaluation of video summaries is examined in section 8.4. Finally, some concluding remarks are given in section 8.5.

8.2 Approaches to Video Summary Generation

8.2.1 General Approach to Video Summary Generation

Most video content may be broadly categorized into two classes:

1. Event-Based Content. These types of video programs contain easily identifiable story units that form either a sequence of different events or a sequence of events and non-events. Examples of the first kind of programs are talk shows and news programs where one event follows another and their boundaries are well-defined. For talk shows each event contains a different guest while for news programs each event is a different news story. The best example of programs where sequence of events and non-events occur are sports programs. Here, the events may correspond to highlights such as touchdowns, home runs, and goals.
2. Uniformly Informative Content. These are programs which cannot be broken down to a series of events as easily as event-based content. For this type of content, all parts of the program may be equally important for the user. Examples of this type of content are sitcoms, presentation videos, documentaries, soap operas, and home movies.

Note that the distinction given above is not clear cut. For example, for sitcoms one can define events according to the appearance of canned laughter. Movies form another example; most action movies have a clear sequence of action and non-action segments.

For event-based content, since the types of events of interest are well-defined, one can use knowledge-based event detection techniques. In this case, the processing will be domain-specific and a new set of events and event detection rules must be derived for each application domain, which is a disadvantage. However, the summaries produced will be more reliable than those generated using general summarization algorithms. This type of algorithm is examined in section 8.2.5.

If domain-based knowledge of events is not available or if one is dealing with uniformly informative content, like documentaries, one has to resort to more general summarization techniques. The main idea behind these techniques is to get rid of the redundancy in a video sequence by clustering similar parts of the sequence together. This may be done in two ways: In the top-down clustering approach the source video is first divided into segments. One way to achieve this is to perform shot boundary detection on the sequence. Another way is to perform time-constrained clustering to identify segments. Once the segments are identified they are clustered to collect similar segments together. In the bottom-up clustering approach the segment detection step is skipped and frames from the video are clustered directly.

Once the content clusters in the video sequence are identified, each cluster is represented using either keyframes or using portions of the segments belonging to clusters. Representation of clusters for summarization display purposes will be examined in depth in section 8.3. In this section we investigate various approaches that have been proposed to generate video summaries.

8.2.2 Speedup of Playback

A simple way to compactly display a video sequence is to present the complete sequence to the user but increase the playback speed. A technique known as time scale modification can be used to process the audio signal so that the speedup can be made with little distortion [16]. The compression allowed by this approach, however, is limited to a summarization ratio (SR) of 1.5-2.5, depending on the particular program genre [17]. Based on a comprehensive user study, Amir et al. report that for most program genres and for novice users, a SF of 1.7 can be achieved without significant loss in comprehension. However, this SR value is not adequate for most summarization applications, which often require a SR between 10 and 20.

8.2.3 Techniques Based on Frame Clustering

Dividing a video sequence into segments and extracting one or more keyframes from each segment was long recognized as one of the simplest and most compact ways of representing video sequences. For a survey of keyframe extraction techniques the reader is referred to [15]. These techniques generally focus on the image data stream only. Color histograms, because of their robustness, have generally been used as the features for clustering.

One of the earliest work in this area is by Yeung and Yeo [2] using time-constrained clustering of shots. Each shot is then labeled according to the cluster to which it belongs, and three types of events, dialogue, action, and other, are detected based on these labels. Representative frames from each event are then selected for the summary. The Video Manga system by Uchihashi et al. [7] clusters individual video frames using YUV color histograms. Iacob, Lagendijk, and Iacob [18] propose a similar technique. However, in their approach video frames are first divided into rectangles, whose sizes depend on the local structure, and *YUV* histograms are extracted from these rectangles. Ratakonda, Sezan, and Crinon [19] extract keyframes for summaries based on the area under the cumulative action curve within a shot, where the action between two frames is defined to be the absolute histogram difference between color histograms of the frames. They then cluster these keyframes into a hierarchical structure to generate a summary of the program.

Ferman and Tekalp [20] select keyframes from each shot using the fuzzy *c*-clustering algorithm, which is a variation of the *k*-means clustering method, based on alpha-trimmed average histograms extracted from frames. Cluster validity analysis is performed to automatically determine the optimal number of keyframes from each shot to be included in the summary. This summary may then be processed based on user preferences, such as the maximum number of keyframes to view, and cluster merging may be performed if there are too many keyframes in the original summary.

The approaches proposed in [15] and [21] both contain a two-stage clustering structure, which is very similar to the method used in [20] but instead of performing shot detection segments are identified by clustering of video frames. Hanjalic and Zhang use features and cluster validity analysis techniques that are similar to those in [20]. Farin, Effelsberg, and de With [21] propose a two-stage clustering technique based on luminance histograms extracted from each frame in the sequence. First, in an approach similar to time-constrained clustering, segments in the video sequence are located by minimizing segment inhomogeneity, which is defined as the sum of the distances of all frames within a segment to the mean feature vector of the segment. Then, the segments obtained are clustered using the Earth-Mover's distance [22]. Yahiaoui, Merialdo, and Huet [23] first cluster frames based on the $L1$ distance between their color histograms using a procedure similar to *k*-means clustering. Then, a set of clusters is chosen as to maximize the coverage over the source video sequence.

An application domain that poses unique challenges for summarization is home videos. Since home videos generally have no plot or summary and contain very little editing, if any, the editing patterns and high level video structure, which offer a strong cue in the summarization of broadcast programs, are absent. However, home videos are inherently time-stamped during recording. Lienhart [11] proposes a summarization algorithm where shots are clustered at four time resolutions using different thresholds for each level of resolution using frame time stamps. Very long shots, which are common in home videos, are shortened by using the heuristic that during important events audio is clearly audible over a longer period of time than less important content.

8.2.4 Techniques Based on Frame Clustering by Dimensionality Reduction

These techniques perform a bottom-up clustering of the video frames selected at fixed intervals. A high dimensional feature vector is extracted from each frame; this dimensionality is then reduced either by projecting the vectors to a much lower dimensional space [24, 25] or by using local approximations to high dimensional trajectories [6, 8] Finally, clustering of frames is performed in this lower dimensional space.

DeMenthon, Kobla, and Doerman [6] extract a 37-dimensional feature vector from each frame by considering a time coordinate together with the three coordinates of the largest blobs in four intervals for each luminance and chrominance channel. They then apply a curve splitting algorithm to the trajectory of these feature vectors to segment the video sequence. A keyframe is extracted from each segment. Stefanidis et al. [8] propose a similar system; however, they split the three-dimensional trajectories of video objects instead of feature trajectories.

Gong and Liu [24] use singular value decomposition (SVD) to cluster frames evenly spaced in the video sequence. Each frame is initially represented using three-dimensional RGB histograms, which results in 1125-dimensional frame feature vectors. Then, SVD is performed on these vectors to reduce the dimensionality to 150 and clustering is performed in this space. Portions of shots from each cluster are selected for the summary. Cooper and Foote [25] sample the given video sequence at a rate of one frame per second and extract a color feature vector from each extracted frame. The cosine of the angle between feature vectors is taken to be the similarity measure between them and a non-negative similarity matrix is formed between all pairs of frames. Non-negative matrix factorization (NMF) [26], is used to reduce the dimensionality of the similarity matrix. NMF is a linear approximation similar to SVD, the difference being the fact that the basis vectors are non-negative.

8.2.5 Techniques Using Domain Knowledge

As discussed in the beginning of this section, if the application domain of the summarization algorithm is restricted to event-based content, it becomes possible to enhance summarization algorithms by exploiting domain-specific knowledge about events. Summarization of sports video has been the main application for such approaches. Sports programs lend themselves well for automatic summarization for a number of reasons. First, the interesting segments of a program occupy a small portion of the whole content; second, the broadcast value of a program falls off rapidly after the event so the processing must be performed in near real-time; third, compact representations of sports programs have a large potential audience; finally, often there are clear markers, such as cheering crowds, stopped games, and replays, that signify important events.

Summarization of soccer has received a large amount of attention recently (see [27] for a survey of work in soccer program summarization). Li, Pan, and Sezan [28] develop a general model for sports programs where events are defined to be the actions in a program that are replayed by the broadcaster. The replay is often preceded by a close-up shot of the key players or the audience. They apply their approach to soccer videos where they detect close-up shots by determining if the dominant color of the shot is close to that of the soccer field. Ekin and Tekalp [27] divide each keyframe of a soccer program into 9 parts and use features based the color content to classify shots into long, medium, and close-up shots. They also detect shots containing the referee and the penalty box. Goal detection is performed similar to [28] by detecting close-up shots followed by a replay. Cabasson and Divakaran [29] detect audio peaks and a motion activity measure to detect exciting events in soccer programs. Based on the heuristic that the game generally stops after an exciting event, they search the program for sequences of high motion followed by very little motion. If an audio peak is detected near such a sequence it is marked as an event and included in the summary.

Domain knowledge can be very helpful even for uniformly informative content. For example, He et al. [12] have proposed algorithms based on heuristics about slide transitions and speaker pitch information to summarize presentation videos.

8.2.6 Techniques Using on Closed-Captions or Speech Transcripts

For some types of programs a large portion of the informational content is carried in the audio. News programs, presentation videos, documentaries, teleconferences, and instructional videos are some examples of such content. Using the spoken text to generate video summaries becomes a powerful approach for these types of sequences. Content text is readily available for most broadcast programs in the form of closed captions. For sequences, like presentations and instructional programs, where this information is not available, speech recognition may be performed to obtain the speech transcript. Once the text corresponding to a video sequence is available, one can use methods of text summarization to obtain a text summary. The portions of the video corresponding to the selected text may then be concatenated to generate the video skim. Processing text also provides a high level of access to the semantic content of a sequence that is not possible to achieve using the image content only.

Agnihotri et al. [30] search the closed-caption text for cue words to generate summaries for talk shows. Cues such as “please welcome” and “when we come back” in addition to domain knowledge about program structure are used to segment the programs into parts containing individual guests and commercial breaks. Keywords are then used to categorize the conversation with each guest into a number of predetermined classes such as *movie* or *music*. In their ANSES system Pickering, Wong, and Rueger [31] use key entity detection to identify important keywords in closed-caption text. Working under the assumption that story boundaries always fall on shot boundaries, they perform shot detection followed by the merging of similar shots based on the similarity of words they contain. They then detect the nouns in text using a part of speech tagger and use lexical chains [32] to rank the sentences in each story. The highest scoring sentences are then used to summarize each news story.

An example of a technique that uses automatic speech recognition (ASR) is the one proposed by Taskiran et al. [9]. The usage of ASR makes their method applicable to cases where the closed-caption text is not available, such as presentations or instructional videos. In their approach the video is first divided into segments at the pause boundaries. Then, each segment is assigned a score using term frequencies within segments. Using statistical text analysis, dominant word pairs are identified in the program and the scores of segments containing these pairs are increased. The segments with highest scores are selected for the summary.

8.2.7 Approaches Using Multiple Information Streams

Most current summarization techniques focus on processing one data stream, which is generally image data. However, multi-modal data fusion approaches, where data from images, audio, and closed-caption text are combined, offer the possibility to greatly increase the quality of the video summaries produced. In this section we look at a few systems that incorporate features derived from multiple data streams.

The MoCA project [14], one of the earliest systems for video summarization, uses color and action content of shots, among other heuristics, to obtain trailers for feature films. The Informedia project constitutes a pioneer and one of the largest efforts in creating a large video database with search and browse capabilities. It uses integrated speech recognition, image processing, and natural language processing techniques for the analysis of video data [13, 33]. Video segments with significant camera motion, and those showing people or a text caption, are given a higher score. Audio analysis includes detection of names in the speech transcript. Audio and video segments selected for summary are then merged while trying to maintain audio/video synchronicity.

Ma et al. [34] propose a generic user attention model by integrating a set of low-level features extracted from video. This model incorporates features based on camera and object motion, face detection, and audio. An attention value curve is obtained for a given video sequence using the model and portions near the crests of this curve are deemed to be interesting events. Then, heuristic rules are employed, based on pause and shot boundaries, and the SR, to select portions of the video for the summary. Another model-based approach is the computable scene model proposed by Chang and Sundaram [35], which uses the rules of film-making and experimental observations in the psychology of audition. Scenes are classified into four categories using audio and video features.

8.3 Summary Visualization Types

After the video summarization is obtained by using the techniques described in section 8.2 it has to be displayed to the user in an intuitive and compact manner. Depending on the desired summary type and length, information from all detected video clusters may be used. Alternatively, importance scores may be assigned to each cluster using various combinations of visual, audio, textual, and other features extracted from the video sequence, and only portions or keyframes extracted from the clusters with the highest scores may be included in the summary. In this section we review some of the summarization display approaches that have been proposed. These methods mainly fall into two categories: Video abstracts based on keyframes extracted from video and video skims where portions of the source video are concatenated to form a much shorter video clip.

8.3.1 Static Visualizations or Video Abstracts

The simplest static visualization method is to present one frame from each video segment, which may or may not correspond to an actual shot, in a storyboard fashion. The problems with this method are that all shots appear equally important to the user and the representation becomes unpractically large for long videos.

One way to alleviate this problem is to rank the video segments and to display only the representative key-frames belonging to the segments with highest scores. In order to further reflect the relative scores of the segments, the keyframes may be sized according to the score of the segment. This approach has been used in [2] and [7] where keyframes from segments are arranged in a "video poster" using a frame packing algorithm. In their PanoramaExcerpts system Taniguchi, Akutsu, and Tonomura [3] use panoramic icons, which are obtained by merging consecutive frames in a shot, in addition to keyframes. In the Informedia project time ordered keyframes, known as filmstrips, were as video abstracts [36].

Although video abstracts are compact, since they do not preserve the time-evolving nature of video programs, they present fundamental drawbacks. They are somewhat unnatural and hard to grasp for nonexperts, especially if the video is complex. Most techniques just present keyframes to the user without any additional metadata, like keywords, which can make the meaning of keyframes ambiguous. Finally, static summaries are not suitable for instructional, and presentation videos, as well as teleconferences, where most shots contain a talking head, and most of the relevant information is found in the audio stream. These deficiencies are addressed by dynamic visualization methods.

8.3.2 Dynamic Visualizations or Video Skims

In these methods the segments with the highest scores are selected from the source video and concatenated to generate a video skim. While selecting portions of the source video to be included in the video skim, care must be exercised to edit the video on long audio silences, which generally correspond to spoken sentence boundaries. This is due to the experimentally verified fact that users find it very annoying when audio segments in the video skim begin in mid-sentence [9, 12].

8.3.3 Other Types of Visualizations

There are also some video browsing approaches that may be used to visualize video content compactly and hence may be considered a form of video summarization.

As part of the Informedia Project, Wactlar [33] proposes video collages, which are rich representations that display video data along with related keyframes, maps, and chronological information in response to a user query. In their BMOVIES system, Vasconcelos and Lippman [4] use a Bayesian network to classify shots as action, close-up, crowd, or setting based on motion, skin tone, and texture features. The system generates a timeline that displays the evolution of the state of the semantic attributes throughout the sequence. Taskiran et al. [5] cluster keyframes extracted from shots using color, edge, and texture features and present them in a hierarchical fashion using a similarity pyramid. In the CueVideo system, Srinivasan et al. [37] provide a video browser with multiple synchronized views. It allows switching between different views, such as storyboards, salient animations, slide shows with fast or slow audio, and full video while preserving the corresponding point within the video between all different views. Ponceleon and Dieberger [38] propose a grid, which they call the movieDNA, whose cells indicate the presence or absence of a feature of interest in a particular video segment. When the user moves the mouse over a cell, a window shows a representative frame and other metadata about that particular cluster. A system to build a hierarchical representation of video content is discussed in Huang et al. [39] where audio, video, and text content are fused to obtain an index table for broadcast news.

8.4 Evaluation of Video Summaries

Since automatic video summarization is still an emerging field, serious questions remain concerning the appropriate methodology in evaluating the quality of the generated summaries. Many researchers do not include any form of quantitative summary evaluations. Evaluation of the quality of automatically generated video summaries is a complicated task because it is difficult to derive objective quantitative measures for summary quality. In order to be able to measure the effectiveness of a video summarization algorithm, one first needs to define features that characterize a good summary, given the specific application domain being studied. As discussed in section 8.1, summaries for different applications will have different sets of desirable attributes. Hence, the criteria to judge summary quality will be different for different application domains.

Automated text summarization dates back at least to Luhn's work at IBM in the 1950s [40], which makes it the most mature area of media summarization. We will apply the terminology developed for text summary evaluation to evaluation of video summaries. Methods for the evaluation of text summaries can be broadly classified into two categories: *intrinsic* and *extrinsic* evaluation methods [41, 42]. In intrinsic evaluation methods the quality of the generated summaries is judged directly based on the analysis of summary. The criteria used may be user judgment of fluency of the summary, coverage of key ideas of the source material, or similarity to an "ideal" summary prepared by humans. On the other hand, in extrinsic methods the summary is evaluated with respect to its impact on the performance for a specific information retrieval task. To the best of our knowledge, all summary evaluations in video summarization literature up to date have been of the intrinsic type, except [13].

Au: meaning of sentence unclear. Reword.

For event-based content, like a sports program, where interesting events are clearly marked, summaries might be judged on their coverage of events from the source video. Two such evaluations are given in [27] and [20]. Ekin and Tekalp [27] give precision and recall values for goal, referee, and penalty box detection, which are important events in soccer games. In Ferman and Tekalp's study [20], the video summary was examined to determine, for each shot, the number of redundant or missing keyframes in the summary. For example, if the observer thought that an important object in a shot was important but no keyframe contained that object, this resulted in a missing keyframe. Although they serve as a form of quantitative summary quality measure, the event detection precision and recall values given in these studies do not reflect the quality of the summaries from the user's point of view.

For uniformly informative content, where events may be harder to identify, different evaluation techniques have been proposed. He et al. [12] determines the coverage of summaries of key ideas from presentation videos by giving users a multiple choice quiz about the source video before and after watching a video skim extracted from it. The quizzes consist of questions prepared by the presentation speakers that are thought to reflect the key ideas of the presentation. The quality of the video skims was judged by the increase in quiz scores. A similar technique was used by Taskiran et al. [9] in evaluating video skims extracted from documentaries. The quiz method has some serious drawbacks: First, it was found that it does not differentiate between different summarization algorithms adequately [9], so is not useful by itself to judge between different algorithms; second, it is not clear how quiz questions can be prepared in an objective manner, except, perhaps, by authors of presentations who are not usually available; finally, the concept of a “key idea” in a video program is ambiguous and may depend on the viewers.

Another intrinsic evaluation method is to have users rate the skims based on subjective questions, e.g., “Was the summary useful?” and “Was the summary coherent?” Such surveys were used in [11, 12, 13].

In intrinsic evaluation of text summaries generally summaries created by experts are used [41]. Using a similar approach would be much more costly and time consuming for video sequences. Another scheme for evaluation may be to present the segments from the original program to many people and let them select the segments they think should be included in the summary, thereby generating a ground truth. This seems to be a promising approach although agreement among human subjects becomes an issue for this scheme.

Extrinsic evaluation methods offer a more promising alternative to summary evaluation by concentrating on the effect of the summary on some task that is related to the goal of the summary. However, the use of such methods has been very rare in video summarization work. The only extrinsic evaluation method we are aware of is the one used by Christel et al. [13]. In this study video skims extracted from documentaries were judged based on the performance of users on two tasks: factfinding, where users used the video skims to locate video segments that answered specific questions; and gisting, where users matched video skims with representative text phrases and frames extracted from source video.

8.5 Conclusion

In this chapter we have reviewed the current state of the art in automatic generation of summaries for video programs. We saw that compared with early approaches, such as [2, 14], the field has matured and new and more powerful approaches for summary generation and visualization have been proposed. Nevertheless, this field is still a very fast growing one, and there remains many open questions, some of them mentioned in this section.

As we saw in section 8.2, many video summarization algorithms concentrate on gathering information from one data stream, such as images, audio, or closed-captions. Systematic gathering of information from all of these streams and fusing them to generate summaries will greatly enhance the summary quality, as can be seen from the high quality summaries produced by the Informedia Project [13]. Analyzing and fusing data from different information systems while keeping their synchronicity is a challenging task.

Au: OK edit?.

More emphasis needs to be placed on visualization of summaries. We need to go beyond the approach of simple lists of keyframes. Instead, the keyframes should be enhanced by extra information, such as keywords obtained from closed-captions and icons providing at-a-glance access to the important aspects of the summary segments. Such systems have recently begun to appear [38]. In order to develop effective summary visualizations, learning user preferences and access patterns is a must.

The problem of deriving good evaluation schemes for automatically generated video summaries is still a complex and open problem. We feel that valuable clues to this problem can be obtained by studying the numerous approaches proposed in text summary evaluations. Large user studies are needed in this area to decide which family of algorithms performs best for a given program genre.

A factor that can influence evaluation results is the value of the summarization factor used to obtain the video summaries. The evaluation results of the same summarization system can be significantly different when it is used to generate summaries of different lengths [41]. Ideal summarization factors for different program genres need to be investigated.

References

[1]Belle L. Tseng, Ching-Yung Lin, and John R. Smith. Video summarization and personalization for pervasive mobile devices. In *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases 2002*, volume 4676, pages 359–370, San Jose, CA, 23–25 January 2002.

[2]Minerva M. Yeung and Boon-Lock Yeo. Video visualization for compact presentation and fast browsing of pictorial content. *IEEE Transactions on Circuits and Systems for Video Technology*, 7(5): 771–785, October 1997.

[3] Y. Taniguchi, A. Akutsu, and Y. Tonomura. Panorama excerpts: Extracting and packing panoramas for video browsing. In *Proceedings of the ACM Multimedia*, pages 427–436, Seattle, WA, 9–13 November 1997.

[4]Nuno Vasconcelos and Andrew Lippman. Bayesian modeling of video editing and structure: Semantic features for video summarization and browsing. In *Proceedings of IEEE International Conference on Image Processing*, Chicago, IL, 4–7 October 1998.

[5]Cuneyt Taskiran, JauYuen Chen, Charles A. Bouman, and Edward J. Delp. A compressed video database structured for active browsing and search. In *Proceedings of the IEEE International Conference on Image Processing*, Chicago, IL, 4–7 October 1998.

[6]Daniel DeMenthon, Vikrant Kobla, and David Doerman. Video summarization by curve simplification, In *Proceedings of the ACM Multimedia Conference*, pages 211–218, Bristol, England, 12–16 September 1998.

[7]Shingo Uchihashi, Jonathan Foote, Andreas Girgenson, and John Boreczky. Video manga: Generating semantically meaningful video summaries. In *Proceedings of ACM Multimedia'99*, pages 383–392, Orlando, FL, 30 October–5 November 1999.

[8] A. Stefanidis, P. Partsinevelos, P. Agouris, and P. Doucette. Summarizing video datasets in the spatiotemporal domain. In *Proceedings of the International Workshop on Advanced Spatial Data Management (ASDM'2000)*, Greenwich, England, 6–7 September 2000.

[9]Cuneyt M. Taskiran, Arnon Amir, Dulce Ponceleon, and Edward J. Delp. Automated video summarization using speech transcripts. In *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases 2002*, volume 4676, pages 371–382, San Jose, CA, 20–25 January 2002.

[10]JungHwan Oh and Kien A. Hua. An efficient technique for summarizing videos using visual contents. In *Proceedings of the i.e., EE International Conference on Multimedia and Expo (ICME'2000)*, New York, NY, 30 July–2 August 2000.

[11]Rainer Lienhart. Dynamic video summarization of home video. In *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases 2000*, volume 3972, pages 378–389, San Jose, CA, January 2000.

[12]Liwei He, Elizabeth Sanocki, Anoop Gupta, and Jonathan Grudin. Auto-summarization of audio-video presentations. In *Proceedings of the 7th ACM International Multimedia Conference*, pages 289–298, Orlando, FL, 30 October–5 November 1999.

[13]Michael G. Christel, Micheal A. Smith, Roy Taylor, and David B. Winker. Evolving video skims into useful multimedia abstractions. In *Proceedings of the ACM Computer-Human Interface Conference (CHI'98)*, pages 171–178, Los Angeles, CA, 18–23 April 1998.

[14]Silvia Pfeiffer, Rainer Lienhart, Stephan Fischer, and Wolfgang Effelsberg. Abstracting digital movies automatically. *Journal of Visual Communication and Image Representation*, 7(4): 345–353, December 1996.

[15]Alan Hanjalic and HongJiang Zhang. An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, 9 (8): 1280–1289, 1999.

[16]Arnon Amir, Dulce B. Ponceleon, Brian Blanchard, Dragutin Petkovic, Savitha Srinivasan, and G. Cohen. Using audio time scale modification for video browsing. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS-33)*, Maui, Hawaii, 4–7 January 2000.

[17]Barry Arons. SpeechSkimmer: A system for interactively skimming recorded speech. *ACM Transactions on Computer Human Interaction*, 4-(1): 3–38, 1997.

[18] S. M. Iacob, R. L. Lagendijk, and M. E. Iacob. Video abstraction based on asymmetric similarity values. In *Proceedings of SPIE Conference on Multimedia Storage and Archiving Systems IV*, volume 3846, pages 181–191, Boston, MA, September 1999.

[19]Krishna Ratakonda, Ibrahim M. Sezan, and Regis J. Crinon. Hierarchical video summarization. In *Proceedings of SPIE Conference Visual Communications and Image Processing*, volume 3653, pages 1531–1541, San Jose, CA, January 1999.

[20]A. M. Ferman and A. M. Tekalp. Two-stage hierarchical video summary extraction to match low-level user browsing preferences. *IEEE Transactions on Multimedia*, 5 (2): 244–256, 2003.

[21]Dirk Farin, Wolfgang Effelsberg, and Peter H. N. de With. Robust clustering-based video-summarization with integration of domain-knowledge. In *Proceedings of the IEEE International Conference on Multimedia and Expo 2002 (ICME'2002)*, pages 89–92, Lausanne, Switzerland, 26–29 August 2002.

[22]Yossi Rubner, Leonidas Guibas, and Carlo Tomasi. The earth mover's distance, multi-dimensional scaling, and color-based image retrieval. In *Proceedings of the ARPA Image Understanding Workshop*, pages 661–668, New Orleans, LA, May 1997.

[23] I. Yahiaoui, B. Merialdo, and B. Huet. Comparison of multi-episode video summarization algorithms. *EURASIP Journal on Applied Signal Processing*, 3(1): 48–55, 2003.

[24]Yihong Gong and Xin Liu. Video summarization using singular value decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 174–180, 13–15 June 2000.

[25]Matthew Cooper and Jonathan Foote. Summarizing video using non-negative similarity matrix factorization. In *International Workshop on Multimedia Signal Processing*, St. Thomas, US Virgin Islands, 9–11 December 2002.

[26]Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Proceedings of the Conference on Advances in Neural Information Processing Systems*, pages 556–562, Denver, CO, 27 November–2 December 2000.

[27]Ahmet Ekin and A. Murat Tekalp. Automatic soccer video analysis and summarization. In *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases 2003*, volume 5021, pages 339–350, Santa Clara, CA, 20–24 January 2003.

[28]Baoxin Li, Hao Pan, and Ibrahim Sezan. A general framework for sports video summarization with its application to soccer. In *Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing*, pages 169–172, Hong Kong, 6–10 April 2003.

[29]Romain Cabasson and Ajay Divakaran. Automatic extraction of soccer video highlights using a combination of motion and audio features. In *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases 2003*, volume 5021, pages 272–276, Santa Clara, CA, 20–24 January 2003.

[30]Lalitha Agnihotri, Kavitha V. Devera, Thomas McGee, and Nevenka Dimitrova. Summarization of video programs based on closed captions, in *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases 2001*, volume 4315, pages 599–607, San Jose, CA, January 2001.

[31]Marcus J. Pickering, Lawrence Wong, and Stefan M. Rueger. ANSES: Summarization of news video. In *Proceedings of the International Conference on Image and Video Retrieval*, volume LNCS 2728, pages 425–434, Urbana, IL, 24–25 July 2003.

[32]Regina Barzilay and Michael Elhadad. Using lexical chains for text summarization. In *Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97)*, Madrid, Spain, July 1997.

[33]Howard D. Wactlar. Informedia - search and summarization in the video medium. In *Proceedings of the IMAGINA 2000 Conference*, Monaco, 31 January–5 February 2000.

[34]Yu-Fei Ma, Lie Lu, Hong-Jiang Zhang, and Mingjing Li. A user attention model for video summarization. In *Proceedings of ACM Multimedia'02*, pages 533–542, Juans Les Pins, France, 1–6 December 2002.

[35]Shih-Fu Chang and Hari Sundaram. Structural and semantic analysis of video. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME2000)*, New York, NY, 30 July–2 August 2000.

[36]Michael Christel, Alexander G. Hauptman, Adrienne S. Warmack, and Scott A. Crosby. Adjustable filmstrips and skims as abstractions for a digital video library. In *Proceedings of the IEEE Conference on Advances in Digital Libraries*, Baltimore, MD, 19–21 May 1999.

[37] S. Srinivasan, D. Ponceleon, A. Amir, B. Blanchard, and D. Petkovic. Engineering the web for multimedia. In *Proceeding of the Web Engineering Workshop (WEBE), WWW-9*, Amsterdam, the Netherlands, 15–19 May 2000.

[38] Dulce Ponceleon and Andreas Dieberger. Hierachical brushing in a collection of video data. In *Proceedings of the 34th Hawaii International Conference on System Sciences (HICSS'34)*, Maui, HI, 3–6 January 2001.

[39] Qian Huang, Zhu Liu, Aaron Rosenberg, David Gibbon, and Behzad Shahraray. Automated generation of news content hierarchy by integrating audio, video, and text information. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3025–3028, Phoenix, AR, 15–19 March 1999.

[40] P. H. Luhn. Automatic creation of literature abstracts. *IBM Journal*, 2(2): 159–165, 1958.

[41] Hongyan Jing, Regina Barzilay, Kathleen McKeown, and Michael Elhadad. Summarization evaluation methods: Experiments and analysis. In *Proceedings of the AAAI Symposium on Intelligent Summarization*, Palo Alto, CA, 23–25 March 1998.

[42] I. Mani, D. House, G. Klein, L. Hirschman, L. Obrst, T. Firmin, M. Chrzanowski, and B. Sundheim. The TIPSTER SUMMAC text summarization evaluation. *National Institute of Standards and Technology*, October 1998.