Chapter XIV

Video Abstraction

Jung Hwan Oh, The University of Texas at Arlington, USA

Quan Wen, The University of Texas at Arlington, USA

Sae Hwang, The University of Texas at Arlington, USA

Jeongkyu Lee, The University of Texas at Arlington, USA

ABSTRACT

This chapter introduces Video Abstraction, which is a short representation of an original video, and widely used in video cataloging, indexing, and retrieving. It provides a general view of video abstraction and presents different methods to produce various video abstracts. Also, it discusses a new approach to obtain a video abstract called video digest that uses the closed-caption information available in most videos. The method is efficient in segmenting long videos and producing various lengths of video abstracts automatically. The authors hope that this chapter not only gives newcomers a general and broad view of video abstraction, but also benefits the experienced researchers and professionals by presenting a comprehensive survey on state- of-the-art video abstraction and video digest methods.

INTRODUCTION AND BACKGROUND

The volume of digital video data has been increasing significantly in recent years due to the wide use of multimedia applications in the areas of education, entertainment, business, and medicine. To handle this huge amount of data efficiently, many techniques about video segmentation, indexing, and abstraction have emerged to catalog, index, and retrieve the stored digital videos. The topic of this chapter is *video abstraction*, a short

representation of an original video that helps to enable the fast browsing and retrieving of the represented contents. A general view of video abstraction, its related works, and a new approach to generate it will be presented in this chapter. Digital video data refers to the video picture and audio information stored by the computer using digital format. In this chapter, the terms "digital video," "video," "film," and "movie" all refer to a digital video unless specified with clarifications.

Before discussing the details of video abstraction, we provide readers with a fundamental view on video. Video consists of a collection of video frames, where each frame is a picture image. When a video is being played, each frame is being displayed sequentially with a certain frame rate. The typical frame rates are 30 and 25 frames/second as seen in the various video formats (NTSC, PAL, etc.). An hour of video has 108,000 or 90,000 frames if it has a 30 or 25 frames/second rate, respectively. No matter what kind of video format is used, this is a huge amount of data, and it is inefficient to handle a video by using all the frames it has. To address this problem, video is divided into segments, and more important and interesting segments are selected for a shorter form — a video abstraction. With granularity from small to large, the segmentation results can be *frame*, *shot*, *scene*, and *video*. Shot is a sequence of frames recorded in a single-camera operation, and scene is a collection of consecutive shots that have semantic similarity in object, person, space, and time. Figure 1 illustrates the relationship among them. Video abstraction methods will use these notions of video structure.

There are two types of video abstraction, *video summary* and *video skimming* (Li, Zhang, & Tretter, 2001). Video summary, also called a *still abstract*, is a set of salient images (*key frames*) selected or reconstructed from an original video sequence. Video skimming, also called a *moving abstract*, is a collection of image sequences along with the corresponding audios from an original video sequence. Video skimming is also called a *preview* of an original video, and can be classified into two sub-types: *highlight* and *summary sequence*. A highlight contains the most interesting and attractive parts of a video, while a summary sequence renders the impression of the content of an entire video. Among all types of video abstractions, summary sequence conveys the highest semantic meaning of the content of an original video. We will discuss the details of video summary and video skimming in the next two sections of the chapter. In a later section, we briefly describe the future work, and give our concluding remarks in the last section.

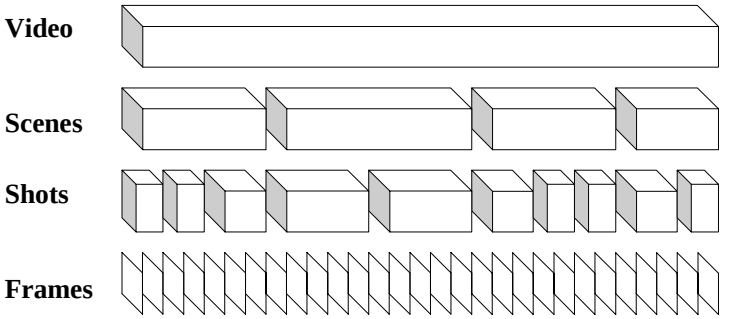
VIDEO SUMMARY

As mentioned in the Introduction, video summary is a set of salient images (key frames) selected or reconstructed from an original video sequence. Therefore, selecting salient images (key frames) from all the frames of an original video is very important to get a video summary. Several different methods using shot boundaries, visually perceptual features, feature spaces, and/or clusters will be discussed in the following subsections.

Shot Boundary-based Key Frame Selection

In the shot boundary-based key frame selection, a video is segmented into a number of shots and one or more key frames are selected from each shot. Together, these selected

Figure 1. Structure of video

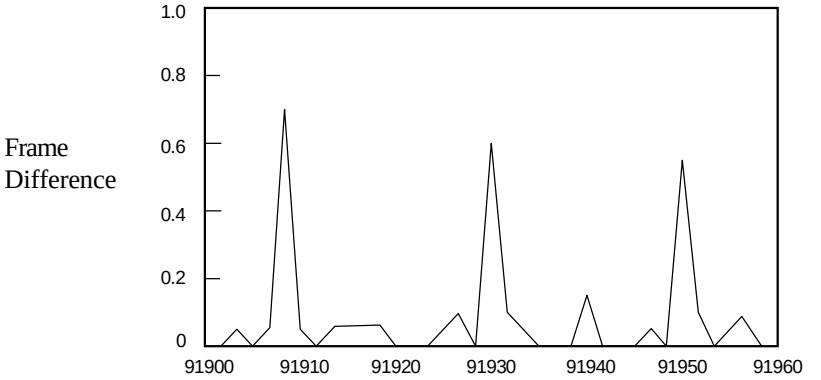


key frames form a video summary. Therefore, the main concern of this approach is how to detect shot boundaries. As mentioned in the previous section, a shot is defined as a collection of frames recorded from a single-camera operation. The principle methodology of shot-boundary detection is to extract one or more features from the frames in a video sequence, and then the difference between two consecutive frames is computed using the features. In case the difference is more than a certain threshold value, a shot boundary is declared.

Many techniques have been developed to detect a shot boundary automatically. These schemes mainly differ in the way the inter-frame difference is computed. The difference can be determined by comparing the corresponding pixels of two consecutive frames (Ardizzone & Cascia, 1997; Gunsel, Ferman, & Tekalp, 1996; Swanberg, Shu, & Jain, 1993). Color or grayscale histograms can be also used (Abdel-Modttaleb & Dimitrova, 1996; Lienhart, Pfeiffer, & Effelsberg, 1996; Truong, Dorai, & Venkatesh, 2000; Yu & Wolf, 1997). Alternatively, a technique based on changes in the edges has also been developed (Zabih, Miller, & Mai, 1995). Other schemes use domain knowledge (Lienhart & Pfeiffer, 1997; Low, Tian, & Zhang, 1996), such as predefined models, objects, regions, etc. Hybrids of the above techniques have also been investigated (Adjeroh & Lee, 1997; Chang, Chen, Meng, Sundaram, & Zhong, 1997; Jiang, Helal, Elmagarmid, & Joshi, 1998; Oh & Hua, 2000; Oh, Hua, & Liang, 2000; Sun, Kankanhalli, Zhu, & Wu, 1998; Taskiran & Delp, 1998; Wactlar, Christel, Gong, & Hauptmann, 1999). Figure 2 shows the frame differences between two consecutive frames computed using the edge change ratio (Zabih et al., 1995) in a certain range (Frame #91900 to Frame #91960) of a video. As seen in the figure, three shot boundaries, between Frame #91906 and Frame #91907, between Frame #91930 and Frame #91931, and between Frame #91950 and Frame #91951 can be detected.

Once shot detection is completed, key frames are selected from each shot. For example, the first, the middle, or the last frame of each shot can be selected as key frames (Hammoud & Mohr, 2000). If a significant change occurs within a shot, more than one key frame can be selected for the shot (Dufaux, 2000).

Figure 2. Example of frame differences by edge change ratio



Perceptual Feature-based Key Frame Selection

In the perceptual feature-based key frame selection, the first frame is selected initially as the most recent key frame, then the following frames are compared using the visually perceptual features. The examples of those features include color, motion, edge, shape, and spatial relationship (Zhang, 1997). If the difference between the current frame and the most recent key frame exceeds a predefined threshold, the current frame is selected as a key frame. We discuss three methods that use different features as follows.

Color-Based Selection

Color is one of the most important features for video frames; it can distinguish an image from others since there is little possibility that two images of totally different objects have very similar colors. Color histogram is a popular method to describe the color feature in a frame due to its simplicity and accuracy. It selects *N* color bins to represent the entire color space of a video and counts how many pixels belong to each color bin of each frame. Zhang (1997) first quantizes the color space into 64 super-cells. Then, a 64-bin color histogram is calculated for each frame where each bin is assigned the normalized count of the number of pixels. The distance ($D_{his}(I,Q)$) between two color histograms, *I* and *Q*, each consisting of *N* bins, is quantified by the following metric:

$$D_{his}(I,Q) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} a_{ij}(I_i - Q_i)(I_j - Q_j) \quad \textbf{(1)}$$

where the matrix a_{ij} represents the similarity between the colors corresponding to bins *i* and *j*, respectively. This matrix needs to be determined from human visual perception studies. If a_{ij} is an identity matrix, this equation measures the Euclidean distance between two color histograms.

After the first key frame is decided manually, the color histograms of consecutive frames are compared with that of the last selected key frame using Equation (1). If the

distance is larger than a predefined threshold, the current frame is decided as the last key frame. The user can change the threshold value to control the amount of key frames. A larger threshold will produce less key frames. On the contrary, a lower threshold will produce more key frames.

Motion-Based Selection

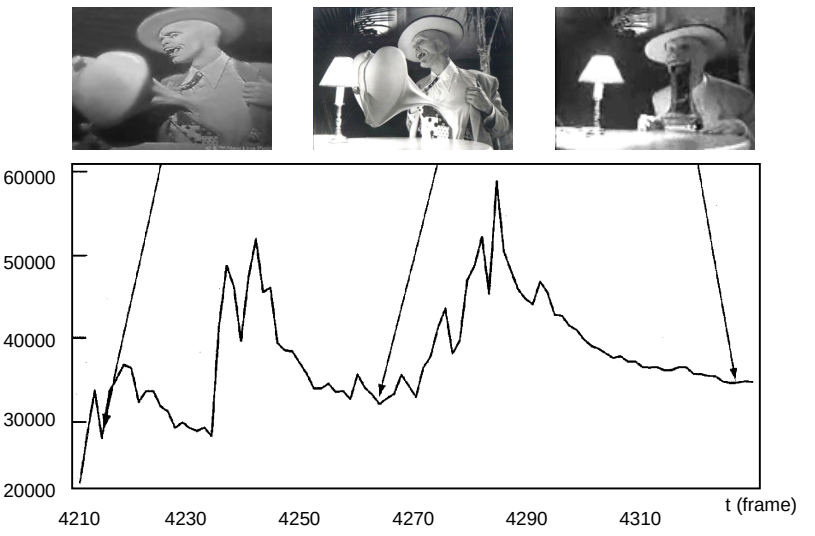
A color histogram is insensitive to camera and object motion (Wolf, 1996; Zhang, 1997). In film production, a director often pans and zooms the camera from one location to another to show the connection between two events. Similarly, several distinct and important gestures by a person will appear in one shot. Therefore, color-based key frame selection may not be enough to render the visual contents of a shot. Wolf (1996) uses motions to identify key frames. In the algorithm for key frame identification, a motion metric, *M(t)* based on optical flow is computed for frame *t* with a size of *r´c* using the following formula:

$$M(t) = \sum_{i=1}^{r} \sum_{j=1}^{c} |o_x(i, j, t)| + |o_y(i, j, t)| \quad \textbf{(2)}$$

where $o_x(i, j, t)$ is the *x* component of optical flow of a pixel positioned *i* and *j* in frame *t* and similarly $o_y(i, j, t)$ for the *y* component. Then, the metric is analyzed as a function of time to select key frames at the minima of motion.

The analysis begins at $t=0$, and identifies two local maxima, m_1 and m_2 using Equation (2) such that the difference between the two values (m_1 and m_2) is larger than a predefined threshold. A frame with a value of the local minimum of *M(t)* between these two local maxima is selected as a key frame. The current m_2 is selected as m_1, and the algorithm continues to find the next m_2 in temporal order. Figure 3 shows the values of

Figure 3. Values of M(t)*s and key frames from a shot in the movie*, The Mask



*M(t)*s for the frames in a shot and a couple of key frames selected by the algorithm. The *M(t)* curve clearly shows the local maxima and minima of motion in the shot.

Object-based Selection

Object-based key frame selection methods can be found in the literature (Ferman, Gunsel, & Tekalp1997; Kim & Huang, 2001). Figure 4 illustrates the integrated scheme for object-based key frame extraction (KFE) (Kim & Huang, 2001), which can be briefly described as follows.

First, it computes the difference of the number of regions between the last key frame and the current frame. When the difference exceeds a certain threshold value, the current frame is considered as a new key frame assuming a different event occurs.

In case the difference is less than a certain threshold value, two 7-dimensional feature vectors (x_k and x_{last}) for the current frame and the last key frame are generated using the seven Hu moments (Nagasaka & Tanaka, 1991; Zhang, 1997), which are known as reasonable shape descriptors. Then, the distance, $D(F_{last}, F_k)$ is computed between x_k and x_{last} by using the city block distance measure (Zhang, 1997). Because the city block distance, which is also called the "Manhattan metric," is the sum of the distances among all variables, it can measure spatial closeness, which helps to decide whether the current frame can be a new key frame. If this difference exceeds a given threshold value, the current frame is selected as a new key frame in the same event.

Feature Vector Space-based Key Frame Selection

The feature vector space-based key frame selection (DeMenthon, Kobla, & Doermann, 1998; Zhao, Qi, Li, Yang, & Zhang, 2000) considers that the frames in a video sequence are characterized by not just one but multiple features. Each frame can be represented by a vector with multiple features, which is a point in multi-dimensional feature space. And the entire feature vectors of the frames in a video sequence can form a curve in the feature space. Key frames are selected based on the property of the curve such as sharp corners or transformations. These perceptually significant points in the curve can be obtained by the multidimensional curve splitting algorithm, which was proposed by Ramer (1972).

Figure 4. Block diagram of integrated system for object-based key frame extraction

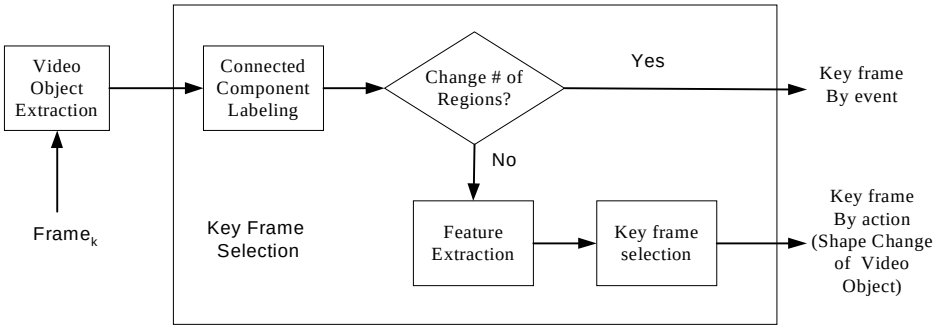
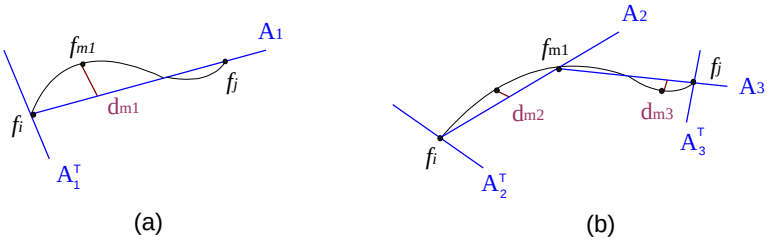


Figure 5. Curve-splitting algorithm

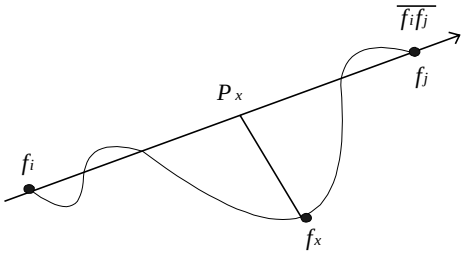


For illustrative purposes, we consider the two dimensional space (Figure 5). f_i and f_j in Figure 5(a) and (b) represent two feature vectors of the first frame and the last frame in a video, respectively. The curve represents the feature vectors of the entire frames in a video. A new Cartesian coordinate system will be built by the axis, A_1 pointing from f_i to f_j and the orthogonal axis, A_1^T. The maximum distance, d_{ml} between the curve and the axis, A_1 is calculated, and compared with the predefined threshold (x). If d_{m1} is larger than x, the curve will be split into two curve pieces $f_i f_{m1}$ and $f_{m1} f_j$, and the same procedure is applied recursively to each of the two curve segments (Figure 5b) until the maximum distance is smaller than the threshold, x.

In the multi-dimensional feature space, we denote a feature vector, f_x of each frame in a video sequence as the following: $f_x = \{ f_{1x}, f_{2x}, \ldots, f_{nx} \}$, where n is the dimension of the feature space. As shown in Figure 6, the distance between the feature point, f_x and the feature line, $f_i f_j$ is calculated by the followings: $\text{Dist}(f_x, f_i f_j) = |f_x - p_x|$, where $p_x = f_i + m(f_j - f_i)$ and $m = (f_x - f_i)(f_j - f_i) / (f_j - f_i)(f_j - f_i)$.

The difference between frames F_i and F_j can be measured as the Euclidean distance in the feature space. The shape and the dimensionality of the feature curve in a video sequence can be formed based on the feature vectors to characterize each individual frame. Therefore, choice of proper features is an important factor for the feature vector space-based key frame selection.

Figure 6. Multi-dimensional feature space



Cluster-based Key Frame Selection

If the number of key frames for each shot is limited to one, it may not represent the content of a shot very well since the complexity of each shot is hardly reflected by one frame. Several key frame-selection techniques based on clustering have been proposed (Hanjalic & Zhang, 1999; Sun et al., 1998; Uchihashi, 1999; Wolf, 1996; Zhuang, Rui, Huang, & Mehrotra, 1998) that select a proper number of key frames from a shot.

In a brief explanation of the cluster-based key frame selection, a given shot, *s* has *N* number of frames, and these *N* number of frames, $\{f_1, f_2, \ldots, f_N\}$ are clustered into *M* number of clusters, $\{C_1, C_2, \ldots, C_M\}$. This clustering is based on the similarity measures among frames, where the similarity of two frames is defined as the similarity of their features, such as color, texture, shape, or a combination of the above. Initially, the first frame, f_1 is selected as the centroid of the first cluster. Then, the similarity values are measured between the next frame f_i and the centroids of existing clusters C_k ($k = 1, 2, \ldots, M$), such that the maximum value and its corresponding cluster, C_j are determined. If this maximum value is less than a certain threshold value, it means frame f_i is not close enough to be added into any existing cluster, then a new cluster is formed for frame f_i. Otherwise, frame f_i is put into the corresponding cluster, C_j. The above process is repeated until the last frame f_N is assigned into a cluster. This is a simple clustering algorithm, but more sophisticated algorithms (e.g., K-mean [Ngo, Pong, & Zhang, 2001]) can be used. After the clusters are constructed, the representative frames are extracted as key frames from the clusters.

Zhuang et al. (1998) use the color histogram of a frame as the feature and select the frame that is closest to the centroid of a cluster as a key frame. They also consider the cluster size such that if the size of a cluster is smaller than a predefined value, those smaller clusters are merged into a larger one using a pruning technique. Sun et al. (1998) perform an iterative partitional-clustering procedure for key frame selection. First, a difference is computed between the first and last frames in each shot. If the difference is less than a threshold value, only the first and last frames are selected as key frames. If the difference exceeds a threshold value and the size of the cluster is smaller than the tolerable maximum size, all frames in the cluster are taken as key frames. Even if the difference is larger than the threshold but the size of the cluster is larger than the tolerable size, the cluster is divided into sub-clusters with the same size, and the partitional-clustering procedure for each sub-cluster is iterated. Hammoud and Mohr (2000) extract multiple key-frames to represent a cluster. First, they select a key frame that is the closest frame to the centroid of a cluster. The similarity between the key frame and each frame in a cluster is calculated. If this similarity is larger than a predefined similarity threshold, the frame is added to a set of key frames. A temporal filter is applied on the set of all selected key frames in order to eliminate the overlapping cases among the constructed clusters of frames.

Other Methods

There are other methods for selecting key frames besides the key frame extraction methods mentioned above. The most intuitive way is to select key frames by sampling at fixed or random distances among frames. The others include face and skin-color detection-based (Dufaux, 2000), statistic-based (Yfantis, 2001), and time-constrained-based (Girgensohn & Boreczky, 2000) methods. Ideas combining several of the above methods are also very common in practice.

VIDEO SKIMMING

As mentioned at the beginning of this chapter, video abstraction is classified into two types: *video summary* and *video skimming*. We have discussed methods in getting video summary in the previous section. In this section, the methods for producing video skimming will be explored. Video skimming consists of a collection of image sequences along with the related audios from an original video. It possesses a higher level of semantic meaning of an original video than the video summary does. We will discuss the video skimming in the following two subsections according to its classification: *highlight* and *summary sequence*.

Highlight

A highlight has the most interesting parts of a video. It is similar to a trailer of a movie, showing the most attractive scenes without revealing the ending of a film. Thus, highlight is used in a film domain frequently. A general method to produce highlights is discussed here. The basic idea of producing a highlight is to extract the most interesting and exciting scenes that contain important people, sounds, and actions, then concatenate them together (Kang, 2001a; Pfeiffer, Lienhart, Fischer, & Effelsberg, 1996). It is illustrated in Figure 7.

Pfeiffer et al. (1996) used visual features to produce a highlight of a feature film and stated that a good cinema trailer must have the following five features: (1) important objects/people, (2) action, (3) mood, (4) dialog, and (5) a disguised ending. These features mean that a highlight should include important objects and people appearing in an original film, many actions to attract viewers, the basic mood of a movie, and dialogs containing important information. Finally, the highlight needs to hide the ending of a movie.

In the *VAbstract* system (Pfeiffer et al., 1996), a scene is considered as the basic entity for a highlight. Therefore, the scene boundary detection is performed first using existing techniques (Kang, 2001b; Sundaram & Chang, 2000; Wang & Chua, 2002; Zabih et al., 1995). Then, it finds the high-contrast scenes to fulfill the trailer Feature 1, the high-motion scenes to fulfill Feature 2, the scenes with basic color composition similar to the average color composition of the whole movie to fulfill Feature 3, the scenes with dialog of various speeches to fulfill Feature 4, and deletes any scene from the last part of an original video to fulfill Feature 5. Finally, all the selected scenes are concatenated together in temporal order to form a movie trailer. Figure 8 shows the abstracting algorithm in the *VAbstract* system.

Figure 7. Diagram of producing a video highlight

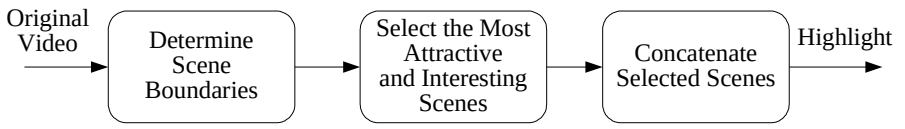
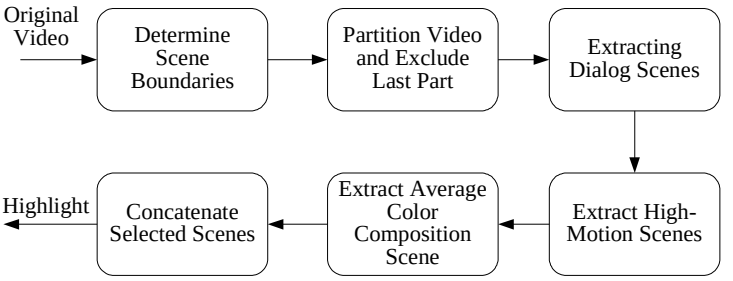


Figure 8. VAbstract abstracting algorithm



We will now discuss the main steps in *VAbstract* system, which are scene boundary detection, extraction of dialog scene, extraction of high-motion scene, and extraction of average color. More details can be found in Pfeiffer et al. (1996).

- *Scene Boundary Detection:* Scene change can be determined by the combination of video- and audio-cut detections. Video-cut detection finds sharp transition, namely *cut* between frames. The results of this video-cut detection are shots. To group the relevant shots into a scene, audio-cut detection is used. A video cut can be detected by using color histogram. If the color histogram difference between two consecutive frames exceeds a threshold, then a cut is determined. The details of the audio-cut detection method can be found in Gerum (1996).
- *Extraction of Dialog Scene:* A heuristic method is used to detect dialog scenes. It is based on the finding that a dialog is characterized by the existence of two "a"s with significantly different fundamental frequencies, which indicates that those two "a"s are spoken by two different people. Therefore, the audio track is first transformed to a short-term frequency spectrum and then normalized to compare with the spectrum of a spoken "a." Because "a" is spoken as a long sound and occurs frequently in most conversations, this heuristic method is easy to implement and effective in practice.
- *Extraction of High-Motion Scene:* Motion in a scene often includes camera motion, object motion, or both. The related motion-detection methods can be found in Section 2.2.2, and in Oh and Sankuratri (2002). A scene with a high degree of motion will be included in the highlight.
- *Extraction of Average Color Scene:* A video's mood is embodied by the colors of each frame. The scenes in the highlight should have the color compositions similar to the entire video. Here, the color composition has physical color properties such as luminance, hue, and saturation. It computes the average color composition of the entire video and finds scenes whose color compositions are similar to the average.

Summary Sequence

Being different from *highlight*, which focuses on the most interesting parts of a video, *summary sequence* renders the impressions of the content of an entire video. It

conveys the highest level of semantic meaning of an original video among all the video abstraction categories. Some representative methods, such as time compression-based, model-based, and text- and speech-recognition- based methods are discussed in the following subsections.

Time Compression-Based Method

The methods to obtain summary sequence are diverse. Text- and speech-recognition- based and model-based methods are two major categories. Also, there are other methods such as the speed-up method to generate a video skimming. Omoigui He, Gupta, Grudin, and Sanocki, (1999) use a time-compression technology to speed up watching a video. This time-compression method consists of two aspects: audio compression and video compression. In this section, we will describe them briefly.

Audio compression can be obtained in a very intuitive way. Suppose we divide the entire audio clip into the equal-sized segments with a length of 100 milliseconds (ms) each. If we delete a 25 ms portion from each segment and concatenate all the remaining 75 ms portions, the total length of the entire audio will be reduced to three-quarters of the original one. The drawback of this simple method is that there is some sound distortion, although the intelligibility of the audio (speech) is un-affected. Other ways to improve the quality of audio time-compression are selective sampling (Neuburg, 1978), sampling with dichotic presentation (Orr, 1971), and Short-Time Fourier Transform (Griffin & Lim, 1984).

As to video compression, it simply drops the frames according to the compression ratio of the audio. If we use the audio time-compression example mentioned above, in which the audio is compressed to three-quarters of an original video, then one frame will be dropped for every four video frames.

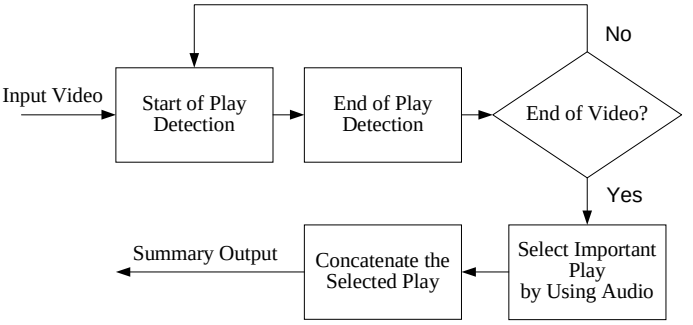
Model-Based Method

Some types of videos have fixed structures that can facilitate the process of extracting important segments. Sports and news programs may fall into this category. A number of video- skimming methods based on the modeling for these types of videos have been reported (Babaguchi, 2000; Li & Sezan, 2002). The basic idea of the model-based method is to use the special structure of the video to select its most important scenes. These special structures include fixed scene structures, dominant locations, and backgrounds. Figure 9 shows the basic idea.

A model-based approach in Li and Sezan (2002) depicts how to model videos based on their domain knowledge. We will describe the details of the method in this section. Li and Sezan summarize the American football broadcast video by defining a set of plays appearing in an entire game. The idea is that first, the start of a play is detected by using field color, field lines, camera motions, team jersey colors, and player line-ups. Second,

Figure 9. Model-based method for summary sequence

Figure 10. Diagram of a football video summarizer



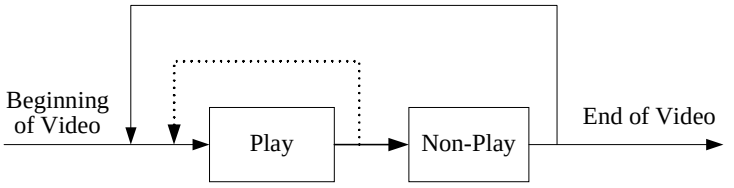
the end of the play is detected by finding camera breaks after the play starts. At last, the waveform of the audio is used to find the most exciting plays, and a summary of the game is constructed by combining them. Figure 10 gives a general view of the whole process.

The method comes out from the observation that generally a football broadcast lasts about three hours but the actual game time is only 60 minutes. There are many unexciting video segments in the broadcast when, for example, the ball is "dead" or there is no play. Therefore, the method defines a "play" as the period between the time when the offensive team has the ball and is about to attempt an advance and the time when the ball is dead. The whole football video is modeled as a series of "plays" interleaved with non-plays. Figure 11 illustrates the structure model of the football video.

Text- and Speech-Recognition-based Method

As mentioned in the previous section, the model-based approach can only be applied to certain types of videos such as sports or news programs that have a certain type of fixed structures. However, most of the other videos do not have these structures. In other words, we need a different approach to apply to general videos that do not have any structure for modeling. To address this problem, the audio information — especially speech in a video — is widely used. This speech information can be obtained from caption data or speech recognition. Caption data usually helps people with hearing problems watch TV programs. Now it is broadly used by the main TV channels and many educational audio and video materials. There are two types of captions, namely, open

Figure 11. Structure model of football video (the inner loop (in dotted line) means that there may have no non-play between plays)



caption and closed caption. Open-caption data is stored and displayed as a part of video frames. Closed-caption data is stored separately from each video frame and displayed as an overlap on video frames. When there is no caption data provided, speech recognition technologies are used to obtain the corresponding text information. Some brands of commercial speech recognition software are available with good performance, such as *Dragon Naturally Speaking* by Scan Soft and *Via Voice* by IBM.

In this section, first, the general idea of existing methods using the text from caption data or speech recognition to get summary sequence will be presented. Then, a new method called *video digest* is discussed.

General Idea of Existing Methods

The general idea of text- and speech-recognition based methods (Agnihoti, 2001; Alexander, 1997; Christel, Smith, Taylor, & Winkler1998; Fujimura, Honda, & Uehara, 2002; He, Sanoki, Gupta, & Grudin, 1999; Li, 2001; Smith & Kanade, 1997; Taskiran, Amir, Ponceleon, & Delp, 2002) is simple, and falls into four steps:

Step 1. Segment a video into a number of shots (or scenes) according to its visual and/or audio (not speech) contents.
Step 2. Obtain text information from the video by capturing caption data or using speech recognition. One (i.e., Signal to Noise Ratio (SNR) technique) of the natural language processing (NLP) techniques is used to get the dominant words or phrases from the text information.
Step 3. Find the shots (or scenes) including the dominant words or phrases obtained in *Step 2*.
Step 4. Concatenate the corresponding shots (or scenes) obtained in *Step 3* together in temporal order.

The main drawbacks of the existing approach are as follows. First, the segmentation results do not always reflect the semantic decomposition of a video, so the generated summary is not always optimal. Therefore, we need a different segmentation technique that considers the semantic of a video. Second, the dominant words or phrases may not be distributed uniformly throughout a video so that the generated summary may miss certain parts of the video. Thus, we need to get different units (e.g., sentences) instead of dominant words or phrases. Third, the existing approach produces only one version (with a fixed length) of a summary from a video due to the lack of its flexibility. However, it is desirable to have several versions (with various lengths) of summaries to satisfy numerous applications with diverse requirements. Fourth, some existing approaches are dependent upon a number of specific symbols (e.g., ">", ">>", or ">>>") in caption or domain knowledge so that they cannot be applied generally. We need a new approach independent of any specific symbol or domain knowledge. To address the four issues above, we introduce a new approach for video-summary sequence as follows.

Top-down Approach for Video Segmentation

The first task for video-summary sequence is to partition a video into a number of segments. The existing methods for video- summary sequence adapt one of the existing shot- boundary detection (SBD) techniques to get the segments that are shots (or

scenes). As mentioned previously, these SBD techniques are bottom-up, in which a sequence of frames is extracted from a video and two consecutive frames are compared to find a boundary. Since these shots are very short, a number of related shots are grouped into a scene. However, it is still an open problem to find optimal scene boundaries by grouping related shots automatically, as mentioned in the literatures (Corridoni, Bimbo, Lucarella, & Wenxue, 1996; Jiang & Elmagarmid, 1998; Rui, Huang, & Mehratra, 1999; Zhong, Zhang, & Chang, 1997). To address this, we segment a video based on top-down fashion. In our technique, a video is segmented into a number of paragraphs using the time gaps that do not have any audio. We call this segment "paragraph" since it is based on the entire text information in a video. Figure 12 shows a sample of a closed-caption script for a documentary, "The Great War." The time-stamps (that have a time format of hour: minute: second: 1/100 second, and are the relative times from the beginning of video) in the first column indicate the starting times of the audios (i.e., speech, music, etc.) in the second column. However, a blank line is occasionally followed by a time-stamp. For example, the fourth time-stamp (0:1:40:75) is followed by a blank line, and the fifth time-stamp (0:1:42:78) has a sentence. In other words, a no-audio time gap lasts around two seconds (00:1:40:75 ~ 00:1:42:78) between [Dramatic Music] and a sentence "IT COLORED EVERYTHING …".

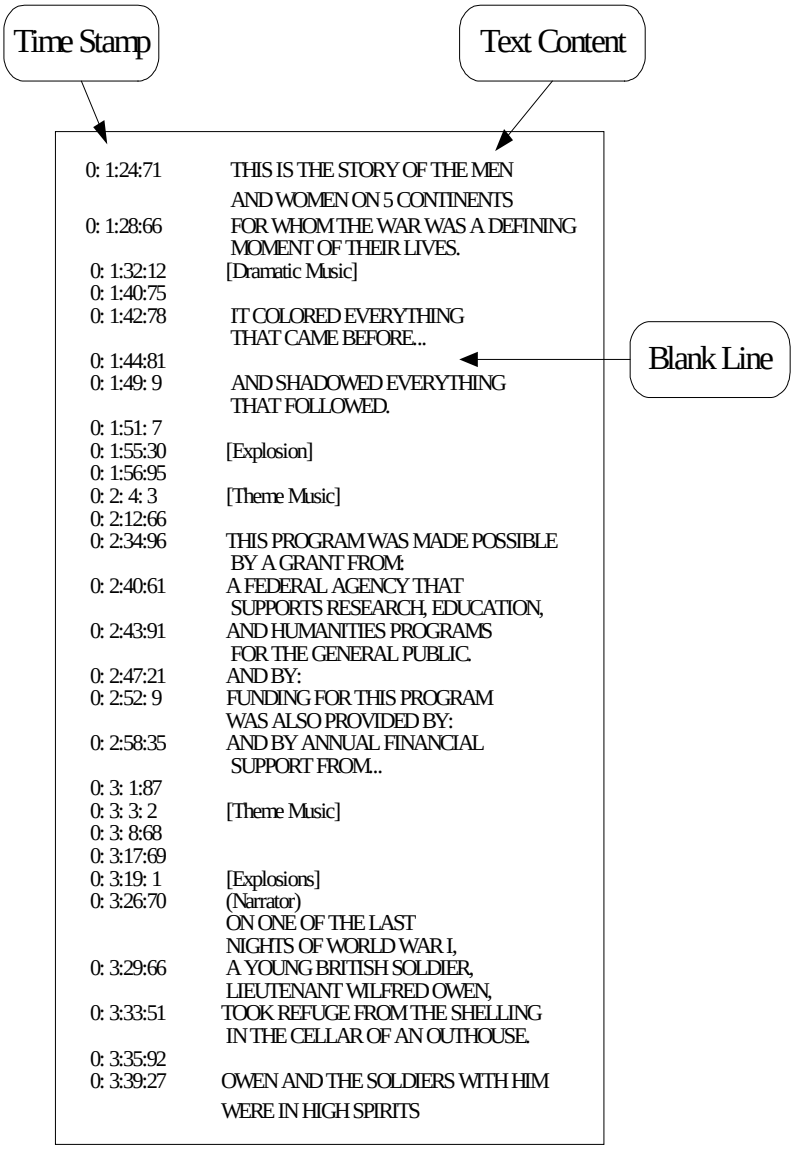
We use these no-audio time gaps to segment a video, but, we only consider the gaps between sentences, music, or sound effects. The gaps in the middle of sentences, music, or sound effects are not used for the segmentation. Figure 13 shows the no-audio time gaps between audios in Figure 12. The long gap between two audios implies semantic segmentation of the original script. In Figure 13, the two largest gaps, #5 and #7, with durations larger than 10 seconds divide the script into three paragraphs. Each paragraph talks about different topics. The first paragraph, which is before Gap #5, tells about the influence of the World War; the second paragraph, which is between Gap #5 and #7, states the contributors of the video; the third paragraph, which is after Gap #7, begins to state a story about a man. In general, an entire video is segmented into a number of paragraphs based on the predefined threshold (e.g., 10 seconds) about the no-audio time gap. If a paragraph is too long, it is re-segmented into subparagraphs.

Summary-Sequence Generation

After a video is segmented into the paragraphs (or subparagraphs), we extract not the words or phrases but the dominant sentences by using one of the Natural Language Processing tools. We can get a number of different versions of summaries that have various lengths by controlling the number of dominant sentences per paragraph (or subparagraph). Since every sentence has its beginning time-stamp and ending time-stamp (which is the beginning of the next one), it is convenient to extract audio and video corresponding to a target sentence. Suppose the sentence, "THIS IS THE STORY OF THE MEN AND WOMEN ON FIVE CONTINENTS FOR WHOM THE WAR WAS A DEFINING MOMENT OF THEIR LIVES" in Figure 12, appears in the summarized text. Then, we use its beginning time-stamp "0:1:24:71" and ending time-stamp "0:1:32:12" to allocate the corresponding audio and video with a length of 7.41 seconds.

The result of our video-skimming approach is called *video digest* to distinguish it from the others. Here are the steps of our approach to get video digest:

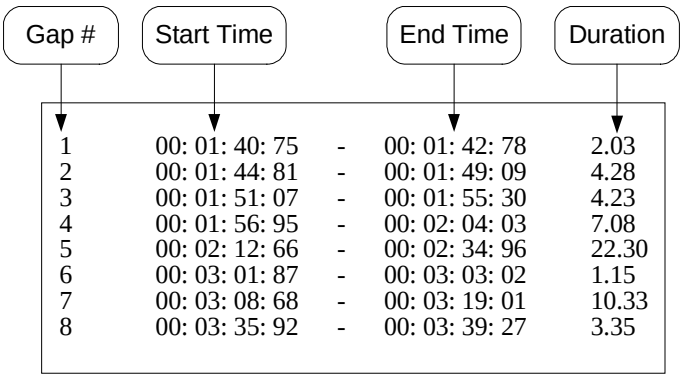
Figure 12. Sample of closed-caption script for "The Great War"



Step 1. Extract the closed-caption script (including time-stamp) as seen in Figure 12 from a video using a caption-decoder device. If caption data is not available, speech recognition can be used to get the same text information.

Step 2. Compute the no-audio time gaps followed by the blank lines as shown in Figure 13, then segment the entire text into paragraphs (or subparagraphs) based on these gaps, using a certain threshold.

Figure 13. No-audio time gaps in Figure 12



Step 3. Extract a number of dominant sentences from each paragraph (or subparagraph). We can control the length of the summary by controlling the number of dominant sentences.

Step 4. Extract videos and audios corresponding to the dominant sentences and concatenate them together in the temporal order.

The advantages of our approach are:

- Our segmentation results reflect the semantic decomposition of a video so that the generated summary is optimal.
- Instead of words or phrases, we introduce a more effective and efficient unit — a sentence — for video segmentation and summary generation.
- Our approach can build several versions (with various lengths) of summaries to satisfy numerous applications with diverse requirements.
- The proposed approach is independent of any specific symbol or domain knowledge.
- Since our approach is based on the spoken sentences, seamless concatenation is easy to achieve automatically.

Experiment Results of Segmentation

Six documentary videos, "The Great War," "Solar Blast," "Brooklyn Bridge," "Nature's Cheats," "Poison Dart Frogs," and "Red Monkey of Zanzibar," are used as the test materials. A number of segments separated by the no-audio time gaps whose duration is larger than 10 seconds are shown in Table 1.

For example, Figure 14 shows the entire no-audio time gaps for a video, "The Great War." There are 11 gaps larger than 10 seconds that separate the entire video into 12 segments (paragraphs).

To measure the effectiveness of our text segmentation approach, we use the *recall* and *precision* metrics. *Recall* (*C/T*) is the ratio of the number (*C*) of paragraph boundaries

Table 1. Segmentation results for test videos (video length format is hh:mm:ss:1/100second)

Video Name	Video Length	Paragraphs
The Great War	00:58:38:20	12
Red Monkey of Zanzibar	00:26:52:20	19
Solar Blast	00:57:07:15	16
Nature's Cheats	00:26:40:09	7
Poison Dart Frogs	00:27:12:18	19
Brooklyn Bridge	01:01:16:10	15

detected correctly over the actual number (*T*) of paragraph boundaries. *Precision* (*C/D*) is the ratio of the number (*C*) of paragraph boundaries detected correctly over the total number (*D*) of paragraph boundaries detected correctly or incorrectly. The performance of our method is illustrated in Table 2. The actual number (*T*) of paragraph boundaries of each video is subjectively defined based on our understanding of the content because the original scripts of all the videos are not in paragraph format. There may be a variant value of *T*, depending on different segmentation granularities used.

As seen in Table 2, the overall results are very good. However, the performance for the video, "Nature's Cheats" is not as good as the others because it has a collection of different natural phenomena, and there are some long, speechless portions in the original video that use just pictures to depict the phenomena.

Figure 14. Pauses of video in "The Great War"

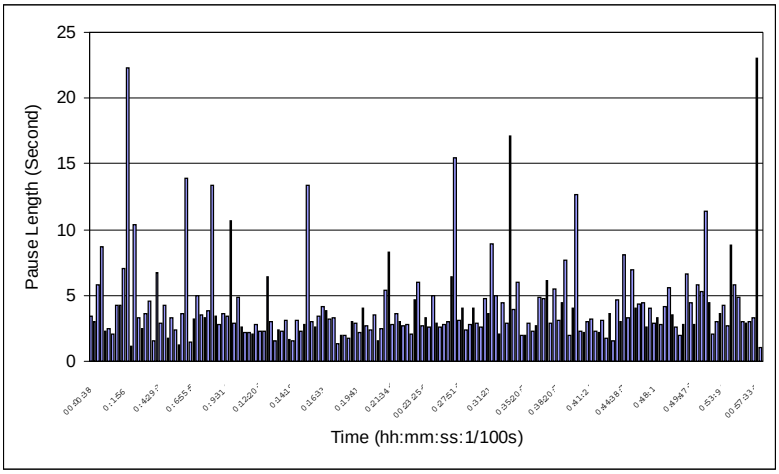


Table 2. Recall and precision of video digest

Video Name	*C*	*D*	*T*	*C/D*	*C/T*
The Great War	11	11	11	1	1
Red Monkey of Zanzibar	18	18	18	1	1
Solar Blast	15	15	15	1	1
Nature's Cheats	5	6	5	0.83	1
Poison Dart Frogs	18	18	18	1	1
Brooklyn Bridge	14	14	14	1	1

Experiment Results of Text Summarization

In our experiments, we use the AutoSummarize tool of Microsoft Word to extract the dominant sentences from the original text script because of its convenience in changing the summary length. Figure 15 shows the AutoSummarize dialog window requiring the user to choose "Type of summary" and "Length of summary."

Here, we give an example of applying different summarization ratios to the third paragraph that is segmented from the video, "The Great War," using the proposed technique. The actual content of the paragraph is shown in Figure 16.

The results of applying different summarization ratios by changing "Percent of Original" in Figure 15 are shown in Figure 17 (a), (b), and (c). If we put 5% as "Percent of Original," we get the result in Figure 17(a) (left). In other words, the most important

Figure 15. Dialog window of AutoSummarize

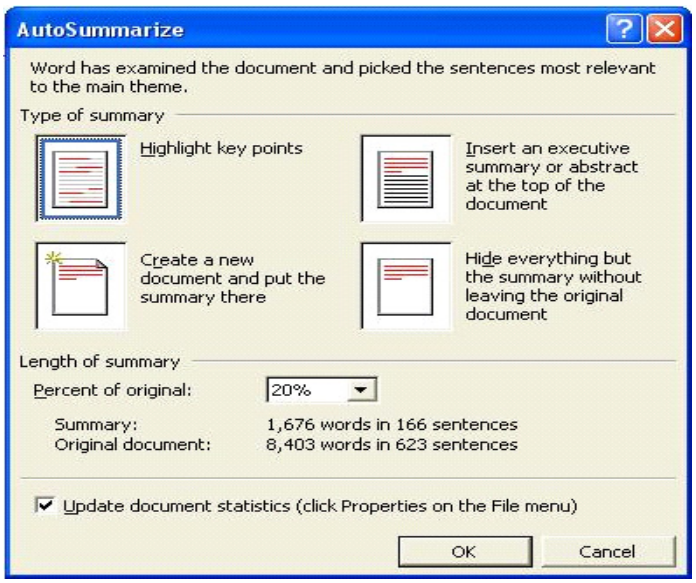


Figure 16. The third paragraph of "The Great War"

> On one of the last nights of World War I, a young British soldier, lieutenant Wilfred Owen, took refuge from the shelling in the cellar of an outhouse. Owen and the soldiers with him were in high spirits. There was finally hope they'd live to see the end of the war. My dearest mother, so thick is the smoke in this cellar that I can hardly see by a candle 12 inches away. So thick are the inmates that I can hardly write for pokes, nudges and jolts. On my left, the company commander snores on a bench. It is a great life. I am more oblivious than alas, yourself, dear mother, of the ghastly glimmering of the guns outside and the hollow crashing of the shells. I hope you are as warm as I am, as serene as I am in here. I am certain you could not be visited by a band of friends half so fine as surround me here. There's no danger down here, or if any, it will be well over before you read these lines. At 11:00 on November 11, 1918, the war ended. One hour later, in the English town of Shrewsbury, there was a knock on the door of this house, the home of Tom and Susan Owen. As their neighbors celebrated the end of the war, the Owens were handed a telegram. In the war's final week, their son Wilfred had been killed, shot in one of the last assaults on the German lines. Wilfred Owen is known as one of his nation's greatest poets. The loss of such a promising life was a tragedy. And yet, he was just one of 9 million people killed in WWI. Of all the questions, these come first: how did it happen? And why?

5% of this paragraph is a sentence, "At 11:00 on November 11, 1918, the war ended." If we put 10% as "Percent of Original," we get the result in Figure 17(a) (right). If we put 20% as "Percent of Original," we get the result in Figure 17(b), and so on.

The relationship among these different lengths of summaries is that the summary result of the larger summary ratio includes that of the smaller summary ratio as seen in Figure 18. For instance, the summary result of the 20% ratio includes all that of the 10%. The summary contents of the 20~40% ratios are found to express the idea of the paragraph reasonably.

As the examples of using different summary ratios demonstrate, all six videos in our test set are segmented into the corresponding number of paragraphs as seen in Table 1 by using a 10- second threshold for the no-audio time gaps. Then, by applying different summarization ratios, we get the results in Table 3 (Time format is hh:mm:ss:1/100s). It is very useful for a user since she/he can choose any ratio at runtime. If a user just wants to see a quick view for the semantic meaning of the video content, the user can select a brief video digest of about 5~10% of the original video length. Larger ratio values (50~80%) can be used for various purposes, in case of time shortage. Figure 19, 20, and 21 show the actual contents of summaries using the 5% ratio from three test videos, "Red Monkey of Zanzibar," "Nature's Cheats," and "Poison Dart Frogs." As seen in these figures, they have the most important points in the videos.

FUTURE WORK

The preliminary result of our video digest method is promising. It is efficient in keeping the semantic meaning of the original video content, and provides various

Figure 17(a). Summary results of ratio 5% (left) and 10% (right)

At 11:00 on November 11, 1918, the war ended.	On one of the last nights of World War I, a young British soldier, lieutenant Wilfred Owen, took refuge from the shelling in the cellar of an outhouse. At 11:00 on November 11, 1918, the war ended.

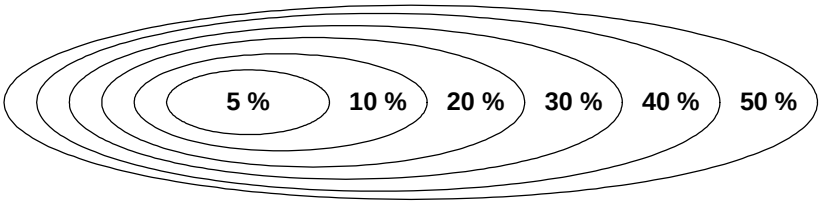
Figure 17(b). Summary result of ratio 20%

On one of the last nights of World War I, a young British soldier, lieutenant Wilfred Owen, took refuge from the shelling in the cellar of an outhouse. It is a great life. At 11:00 on November 11, 1918, the war ended. Wilfred Owen is known as one of his nation's greatest poets.

Figure 17(c). Summary result of ratio 50%

On one of the last nights of World War I, a young British soldier, lieutenant Wilfred Owen, took refuge from the shelling in the cellar of an outhouse. Owen and the soldiers with him were in high spirits. There was finally hope they'd live to see the end of the war. So thick are the inmates that I can hardly write for pokes, nudges and jolts. On my left, the company commander snores on a bench. It is a great life. I am more oblivious than alas, yourself, dear mother, of the ghastly glimmering of the guns outside and the hollow crashing of the shells. At 11:00 on November 11, 1918, the war ended. In the war's final week, their son Wilfred had been killed, shot in one of the last assaults on the German lines. Wilfred Owen is known as one of his nation's greatest poets. The loss of such a promising life was a tragedy.

Figure 18. Relationship among the different lengths of summaries



versions with various lengths of video summaries automatically. To improve the proposed scheme more, we will address the following issues:

- Currently, we are using 10 seconds as a threshold value for paragraph segmentation. This value cannot be universal for all different types of videos. We will study an algorithm to find an optimal value for each video since one value for all types is not practical.

Table 3. Different summarization ratios of six test videos

Text Percentage	5%	10%	20%	50%	100%
The Great War	00:02:47	00:05:20	00:10:05	00:24:24	00:58:34
Red Monkey of Zanzibar	00:00:42	00:02:01	00:03:22	00:09:00	00:26:52
Solar Blast	00:02:31	00:04:29	00:09:13	00:15:25	00:57:07
Nature's Cheats	00:00:48	00:02:05	00:03:58	00:12:31	00:26:40
Poison Dart Frogs	00:00:39	00:02:08	00:03:28	00:09:35	00:27:12
Brooklyn Bridge	00:03:06	00:05:28	00:09:45	00:27:50	01:01:16
Total	00:10:33	00:21:31	00:39:41	01:38:45	04:17:41

Figure 19. Summary result of "Red Monkey of Zanzibar" using ratio 5%

Shy and reclusive, these monkeys are lazing high in the trees the way forest monkeys have always lived. Zanzibar Island, 25 miles from the Tanzanian coast. Remarkably, every Shamba monkey seems to know that there's nothing like a piece of charcoal to ease indigestion. As more and more bush is destroyed, the duikers may have no place left to hide. A greater bush baby. In the forest, the smell of smoke is a smell of danger. Soon the whole troop finds the courage to taste the charred sticks.

Figure 20. Summary result of "Nature's Cheats" using ratio 5%

Strong males fight for the right to win females. The plumage of the males varies in color. Not all females are won by fighting. Male Natterjack toads court their females by serenading them at dusk. Ant larvae and eggs are kept in a special chamber. Deep inside the reed bed, well hidden from predators, a reed warbler is brooding her eggs. The Nephila spider, goliath, has just caught a butterfly for lunch. As it moves from plant to plant, it unwittingly transfers the pollen, so fertilizing the lilies. The most theatrical con artist is a hog nosed snake. The indigo snake retreats in disgust. The lily trotter stays put. Hardly a cheat? The male bees home in and make frenzied attempts to mate with the flowers.

Figure 21. Summary result of "Poison Dart Frogs" using ratio 5%

The islands are thick with tropical vegetation. Most frogs stay hidden. Scientist Kyle Summers is not intimidated. This frog can be handled without any risk. It's called a strawberry poison dart frog. These frogs eat a lot of ants, and that's unusual. Brown tree frog males over-inflate their throats to amplify their calls. Poison frogs do things differently of course. So females on one island might prefer red males, whereas on another island they might prefer a green male. Banana plantations and coconut groves have replaced natural rainforest.

- The current data set of videos has around four hours of six documentaries. We will include other types of videos such as movies and TV dramas to which we will apply our scheme to generate various summaries.
- We will implement a prototype that processes the first step through the last step automatically.
- We will organize and represent the summary results using MPEG-7 standard and XML.

CONCLUDING REMARKS

We presented two types of video abstractions, video summary and video skimming, in this chapter. As we mentioned, video summary is a set of salient images (key frames) selected from an original video sequence. Video skimming, which is called a preview, consists of a collection of image sequences along with the corresponding audios from an original video sequence. It can be classified into two sub-types: highlight and summary sequence. The highlight has the most interesting and attractive parts of a video, while the summary sequence renders the impression of the content of the entire video. Among all types of video abstractions, summary sequence conveys the highest semantic meaning of the content of an original video.

We discussed a number of methods for video summary and video skimming, and introduced a new technique to generate video- summary sequences. In this new approach, the video segmentation is performed by a top-down fashion to reflect the content of the video. One of the natural language processing tools is used effectively to produce various lengths of different summaries. We tested the proposed approach based on four hours of documentary videos, and the test results provided the promising results. We will further investigate the issues mentioned in Future Work.

REFERENCES

Abdel-Mottaleb, M., & Dimitrova, N. (1996). CONIVAS: CONtent-based image and video access system. *Proceedings of ACM International Conference on Multimedia*, Boston, MA, 427-428.

Adjeroh, D.A., & Lee, M C. (1997). Adaptive transform domain video scene analysis. *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, Ottawa, Canada, 203-210.

Agnihotri, L. (2001). Summarization of video programs based on closed captions. *Proceedings of SPIE*, Vol.4315, San Jose, CA, 599-607.

Alexander, G. (1997). Informedia: News-on-demand multimedia information acquisition and retrieval. In M. Maybury (Ed.), *Intelligent Multimedia Information Retrieval*, pp. 213-239. Menlo Park, CA: AAAI Press.

Ardizzone, E., & Cascia, M. (1997). Automatic video database indexing and retrieval. *Multimedia Tools and Applications*, 4, 29-56.

Babaguchi, N. (2000). Towards abstracting sports video by highlights. *Proceedings of IEEE International Conference on Multimedia and Expo*, 2000 (ICME 2000), New York, 1519-1522.

Chang, S., Chen, W., Meng, H.J., Sundaram, H., & Zhong, D. (1997). VideoQ: An automated content-based video search system using visual cues. *ACM Proceedings of the Conference on Multimedia* '97, Seattle, Washington, 313-324.

Christel, M., Smith, M., Taylor, C., & Winkler, D. (1998). Evolving video skims into useful multimedia abstractions. *Proceedings of CHI 1998*, Los Angeles, CA, 171-178.

Corridoni, J.M., Bimbo, A.D., Lucarella, D., & Wenxue, H. (1996). Multi-perspective navigation of movies. *Journal of Visual Languages and Computing*, 7, 445-466.

DeMenthon, D., Kobla, V., & Doermann, D. (1998). Video summarization by curve simplification. *Proceedings of ACM Multimedia 1998*, 211-218.

Dufaux, F. (2000). Key frame selection to represent a video. *Proceedings of IEEE 2000 International Conference on Image Processing*, Vancouver, BC, Canada, 275-278.

Ferman, A., Gunsel, B., & Tekalp, A. (1997). Object-based indexing of MPEG-4 compressed video. *Proceedings of SPIE*-3024, San Jose, CA, 953-963.

Fujimura, K., Honda, K., & Uehara, K. (2002). Automatic video summarization by using color and utterance information. *Proceedings of IEEE International Conference on Multimedia and Expo*, 49-52.

Gerum, C. (1996). *Automatic recognition of audio-cuts* (Automatische Erkennung von Audio-Cuts). Unpublished Master's thesis, University of Mannheim, Germany.

Girgensohn, A., & Boreczky, J. (2000). Time-constrained key frame selection technique. *Multimedia Tools and Applications*, 11(3), 347-358.

Griffin, D.W., & Lim, J.S. (1984). Signal estimation from modified shot-time Fourier transform. *IEEE Transaction on Acoustics, Speech, and Signal Processing*, ASSP-32(2), 236-243.

Gunsel, B., Ferman, A., & Tekalp, A. (1996). Video indexing through integration of syntactic and semantic features. *Proceedings of 3rd IEEE Workshop on Applications of Computer Vision(WACV'96)*, Sarasota, FL, 90-95.

Hammoud, R., & Mohr, R. (2000, Aug.). A probabilistic framework of selecting effective key frames from video browsing and indexing. *Proceedings of International Workshop on Real-Time Image Sequence Analysis*, Oulu, Finland, 79-88.

Hanjalic, A., & Zhang, H. (1999). An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis. *IEEE Transaction on Circuit and Systems for Video Technology*, 9(8), 1280-1289.

He, L., Sanocki, E., Gupta, A., & Grudin, J. (1999). Auto-summarization of audio-video presentations. *Proceedings of ACM Multimedia'99*, Orlando, FL, 489-493.

Jiang, H., & Elmagarmid, A. (1998). WVTDB - A semantic content-based video database system on the World Wide Web. *IEEE Transactions on Knowledge and Data Engineering*, 10(6), 947-966.

Jiang, H., Helal, A., Elmagarmid, A.K., & Joshi, A. (1998). Scene change detection techniques for video database system. *Multimedia Systems*, 186-195.

Kang, H. (2001a). Generation of video highlights using video context and perception. *Proceedings of Storage and Retrieval for Media Databases*, SPIE, Vol. 4315, 320-329.

Kang, H. (2001b). A hierarchical approach to scene segmentation. *IEEE Workshop on Content-Based Access of Image and Video Libraries* (CBAIVL 2001), 65-71.

Kim, C., & Hwang, J. (2001). An integrated scheme for object-based video abstraction. *Proceedings of ACM Multimedia 2001*, Los Angeles, CA, 303-309.

Li, B., & Sezan, I. (2002). Event detection and summarization in American football broadcast video. *Proceedings of SPIE, Storage ad Retrieval for Media Databases*, 202-213.

Li, Y. (2001). Semantic video content abstraction based on multiple cues. *Proceedings of IEEE ICME 2001*, Japan.

Li, Y., Zhang, T., & Tretter, D. (2001). An overview of video abstraction techniques. Retrieved from the World Wide Web: *http://www.hpl.hp.com/techreports/2001/HPL-2001-191.html*

Lienhart, R., & Pfeiffer, S. (1997). Video abstracting. *Communications of the ACM*, 4(12), 55-62.

Lienhart, R., Pfeiffer, S., & Effelsberg, W. (1996). The MoCA workbench: Support for creativity in movie content analysis. *Proceedings of the IEEE Int. Conference on Multimedia Systems '96*, Hiroshima, Japan.

Low, C.Y., Tian, Q., & Zhang, H. (1996). An automatic news video parsing, indexing and browsing system. *Proceedings of ACM International Conference on Multimedia*, Boston, MA, 425-426.

Nagasaka, A., & Tanaka, Y. (1991). Automatic video indexing and full-video search for object appearance. *Proceedings of the IFIP TC2/WG2.6, Second Working Conference on Visual Database Systems*, North-Holland, 113-127.

Neuburg, E.P. (1978). Simple pitch-dependent algorithm for high quality speech rate changing. *Journal of the Acoustic Society of America*, 63, 2, 624-625.

Ngo, C.W., Pong, T.C., & Zhang, H.J. (2001, Oct.). On clustering and retrieval of video shots. *Proceedings of ACM Multimedia 2001*, Ottawa, Canada, 51-60.

Oh, J., & Hua, K.A. (2000). Efficient and cost-effective techniques for browsing and indexing large video databases. *Proceedings of ACM SIGMOD*, Dallas, TX, 415-426.

Oh, J., Hua, K. A., & Liang, N. (2000). A content-based scene change detection and classification technique using background tracking Sept 30 - Oct 3, San Jose, CA, 254-265.

Oh, J., & Sankuratri, P. (2002). Computation of motion activity descriptors in video sequences. In N. Mastorakis & V. Kluev (Eds.), *Advances in Multimedia, Video and Signal Processing Systems*, pp. 139-144. New York: WSEAS Press.

Omoigui, N., He, L., Gupta, A., Grudin, J., & Sanocki, E. (1999). Time-compression: System concerns, usage, and benefits. *Proceedings of ACM Conference on Computer-Human Interaction*, 136-143.

Orr, D. B. (1971). A perspective on the perception of time-compressed speech. In P. M. Kjldergaard, D. L. Horton, & J. J. Jenkins (Eds.), *Perception of Language*, pp. 108-119. Englewood Cliffs, NJ: Merrill.

Pfeiffer, S., Lienhart, R., Fischer, S., & Effelsberg, W. (1996). Abstracting digital movies automatically. *Journal of Visual Communication and Image Representation*, 7(4), 345-353.

Ramer, U. (1972). An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1, 244-256.

Rui, Y., Huang, T., & Mehratra, S. (1999). Constructing table-of-content for videos. *ACM Multimedia Systems*, 7(5), 359-368.

Smith, M., & Kanade, T. (1997). Video skimming and characterization through the combination of image and language understanding. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, 775-781.

Sun, X., Kankanhalli, M., Zhu, Y., & Wu, J. (1998). Content-based representative frame extraction for digital video. *Proceedings of IEEE Multimedia Computing and Systems'98*, Austin, TX, 190-193.

Sundaram, H., & Chang, S. (2000). Video scene segmentation using video and audio Features. *ICME2000*, 1145-1148.

Swanberg, D., Shu, C., & Jain, R. (1993). Knowledge-guided parsing in video databases. *Proceedings. of SPIE Symposium on Electronic Imaging: Science and Technology*, San Jose, CA, 13-24.

Taskiran, C., Amir, A., Ponceleon, D., & Delp, E. (2002). Automated video summarization using speech transcripts. *Proceedings of SPIE*, Vol. 4676, 371-382.

Taskiran, C., & Delp, E. J. (1998). Video scene change detection using the generalized trace. *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP*), Seattle, Washington, 2961-2964.

Truong, B., Dorai, C., & Venkatesh, S. (2000). New enhancements to cut, fade and dissolve detection processes in video segmentation. *Proceedings of ACM Multimedia 2000*, Los Angeles, CA, 219-227.

Uchihashi, S. (1999). Video Manga: Generating semantically meaningful video summaries. *Proceedings of ACM Multimedia'99*, Orlando, FL, 383-392.

Wactlar, H., Christel, M., Gong, Y., & Hauptmann, A. (1999). Lessons learned from building terabyte digital video library. *Computer*, 66-73.

Wang, J., & Chua, T. (2002). A framework for video scene boundary detection. *Proceedings of the 10th ACM international conference on Multimedia*, Juan-les-Pins, France, 243-246.

Wolf, W. (1996). Key frame selection by motion analysis. *Proceedings of IEEE International. Conference on Acoustics, Speech, and Signal Processing*, Atlanta, GA, 1228-1231.

Yfantis, E.A. (2001). An algorithm for key-frame determination in digital video. *Proceedings of 16th ACM Symposium on Applied Computing (SAC 2001)*, Las Vegas, 312-314.

Yoshitaka, A., Hosoda, Y., Hirakawa, M., & Ichikawa, T. (1998). Content-based retrieval of video data based on spatiotemporal correlation of objects. *Proceedings of 1998 IEEE Conference on Multimedia Computing and Systems*, Austin, TX, 208-213.

Yu, H., & Wolf, W. (1997). A visual search system for video and image databases. *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, Ottawa, Canada, 517-524.

Zabih, R., Miller, J., & Mai, K. (1995). A feature-based algorithm for detecting and classifying scene breaks. *Proceedings of the Third ACM International Conference on Multimedia*, San Francisco, CA, 189-200.

Zhang, H.J. (1997). An integrated system for content-based video retrieval and browsing. *Pattern Recognition*, 30(4), 643-658.

Zhao, L., Qi, W., Li, S., Yang, S., & Zhang, H. (2000). Key-frame extraction and shot retrieval using nearest feature line (NFL). *Proceedings of ACM Multimedia Workshop* 2000, Los Angeles, CA, 217-220.

Zhong, D., Zhang, H., & Chang, S. (1997). *Clustering methods for video browsing and annotation.* Columbia University.

Zhuang, Y., Rui, Y., Huang, T., & Mehrotra, S. (1998). Adaptive key frame extraction using unsupervised clustering. *Proceedings of International Conference on Image Processing*, Chicago, IL, 870-886.