Automatic home video abstraction using audio contents

Ming Zhao Caifu Chen Chun Chen Jiajun Bu
School of Computer Science, Zhejiang University, Hangzhou, 310027, P.R.China
Contact: {bjj, chenc}@cs.zju.edu.cn

ABSTRACT

With the increasing number of people who can afford to make videos to record their lives, home videos play more and more important role in people's lives. Video abstraction is an efficient way to help review such a huge amount of home videos. In this paper, an automatic home video abstraction method mainly using audio contents is presented. The audio contents are first segmented and classified as speech, music, silence and special sounds basing on audio short-time features and morphology. Then special sounds are further categorized as songs, laughter, applause, scream and others using Hidden Markov Model (HMM). After that, motion level and blur degree are acquired using the video contents. Finally, video segments containing special effects, such as speech, laughter, song, applause, scream, and specified motion level and blur degree, are extracted as the main parts of the abstract. The remaining parts of the abstract are generated using key frame information. The experimental results show that the proposed algorithm can extract desired parts of home video to generate satisfactory video abstracts.
Keywords: Home video, video summarization, video browsing and indexing, content-based video retrieval

1. INTRODUCTION

The fast evolution of digital video has brought many new applications and consequently, research and development of new technologies, which will lower the costs of video archiving, cataloging and indexing, as well as improve the efficiency, usability and accessibility of stored videos, are greatly needed. Among all possible research areas, one important topic is how to enable a quick browse of a large collection of video data and how to achieve efficient content access and representation. To address these issues, video abstraction techniques have emerged and have been attracting more research interest in recent years. Video abstraction is a short summary of the content of a longer video document. Specifically, a video abstract is a sequence of still or moving images representing the content of a video in such a way that the target party is rapidly provided with concise information about the content while the essential message of the original is well preserved [1]. Theoretically a video abstract can be generated both manually and automatically, but due to the huge volumes of video data and limited manpower, it's getting more and more important to develop fully automated video analysis and processing tools so as to reduce the human involvement in the video abstraction process.

The development of computer multimedia techniques makes it easy for people to capture their lives by way of videos. We call those videos captured by families home videos. Home videos usually add up to many hours of material, which makes it inconvenient for people to review them. The reasons for this are manifold. First, the raw video material is unedited, and therefore long-winded and lacking appealing things. Although video editing would help, it is still too time-consuming. Second, video editing is inflexible and cannot adjust to the viewers' various needs. However, a system capable of abstracting raw videos into shorter ones automatically can not only offer appealing things but also the flexibility for different purpose. For home video, people could not have much time to generate their video abstracts. So, automatic home video abstraction is very useful to the users. They can use their video abstracts for different purpose. They could share their videos with his relatives or friends. And for different people, they want to generate different video abstracts. These video abstracts can differ in many ways including contents, sizes and other things. They could even use video abstracts to save storage. To meet all these needs, an automatic method is very useful. There are two fundamentally different kinds of abstracts: still- and moving-image abstracts. The still-image abstract, also known as a static storyboard, is a small collection of salient images extracted or generated from the underlying video source. The moving-image abstract, also known as moving storyboard, or multimedia summary, consists of a collection of image

sequences, as well as the corresponding audio abstract extracted from the original sequence and is thus itself a video clip but of considerably shorter length. It's usually more natural and more interesting for users to view a moving-image than watching a slide show. So in this paper, a moving-image abstracts are used for home video abstraction.

In the VAbstract system developed by the University of Mannheim, Germany [1][2], the most characteristic movie segments are extracted for the purpose of automatically producing a movie trailer. Specifically, the scenes containing important objects/people are detected and the high-action scenes are extracted; also, the scenes that have a basic color composition similar to the average color composition of the entire movie, are included in the abstract with the hope that they may represent the basic mood of the original movie; moreover, the recognition of dialog scenes is performed. Finally all selected scenes (except the last part of the movie), organized in their original temporal order, forms the movie trailer. There are some interesting ideas in this paper, but some parts of the algorithm are too simple to be effective and will need lots of improvement. However, it was appropriate for making movie trailers, but not very suitable for home videos, because it only consider the conditions in movies. Home videos have their own features. It used some audio features for the recognition of dialog, explosions and gunfire, but very limited and simple. The Informedia Project at Carnegie Mellon University [3] aims to create a very short synopsis of the original video by extracting the significant audio and video information. Particularly, text keywords are first extracted from manual transcript and closed captioning, then the audio skimming is created by extracting the audio segments corresponding to the selected keywords as well as including some of their neighboring segments for better comprehension. Next, the image skimming is created As a result, a set of video frames, which may not align with the audio in time, but may be more appropriate for image skimming in visual aspect are extracted. Finally the video skimming is generated by analyzing the word relevance and the structure of the prioritized audio and image skimming. Experiments of this skimming approach have shown impressive results on limited types of documentary video that have very explicit speech or text contents. However, satisfying results may not be achievable using such a text-driven approach on other videos with a soundtrack containing more complex audio contents. And for home videos, there are nearly no text keywords to be extracted. In the work reported by A. Hanjalic and H.J. Zhang [4], they first cluster all video frames into an optimal number of clusters. One representative frame (key frame) is then chosen from each of these clusters. Lastly, the skimming is generated by concatenating all video shots which contain at least one extracted key frame. But this method neglected the audio contents.

All of the above methods do not mean to abstract home videos, but all videos. However, home videos do have their own characteristics. So R. Lienhart [5][6] proposed an automatic video abstracting method for home video. First, the time and date information of the recordings are obtained. Then, all shots are clustered into 5 different levels based on the date and time, extracted from the video sequence, they are taken. In the next step, a shot shortening process is performed where longer shots are uniformly segmented into 2-minute-long clips. To choose the desired clips, the sound pressure level of the audio signal is calculated and employed in the selection process based on the observation that during important events, the sound is usually more clearly audible over a long period of time than is the case with less important content. Finally, all selected clips are assembled to form the final abstract using pre-designed video transition effects. After extensive investigation, we believe that audio features are more fit and practical for home video abstraction. Usually, people are the focus of home videos and they speak, laugh, applaud or even scream under different circumstances. Different sounds made by people can indicate different events, but important events are often accompanied with special sounds. So, audio contents are very important clues for home video abstraction. Compared with videos' visual features, the audio ones have their own advantages especially for home videos in several ways. First, they are easier to extract. Although we can detect people from the video and even recognize them by visual features, yet it is difficult, time-consuming and not practical for video abstraction. However, as to audio features, less data are processed and it is much easier to detect speech segments and recognize people by their voice. Second, it is easier to detect most important events and appealing things by audio features. For example, if there happen to be any interesting things during traveling, people will talk or laugh or cry out. So these events can be detected by audio contents easily, while it is very difficult or even impossible for visual contents to do so. Although R. Lienhart [5][6] used audio features, the sound pressure level. But it was fairly simple and did not utilize the abundant audio contents effectively.

Based on the above observation, we attempt an abstraction technique mainly using audio contents. The audio special effects such as speech, song, laughter, applause, scream are extracted by audio segmentation and classification. Video special effects (motion level and blur degree) are also acquired. Finally, video segments containing these audio and video special effects are extracted as the main parts of the abstract. The rest of this paper is organized as follows. In section 2, we give an overview of our video abstraction system. Audio segmentation and classification is presented in

section 3 and video content processing is presented in Section 4. Section 5 gives the abstraction algorithm. And experimental results are presented in section 6. We conclude this paper in section 7.

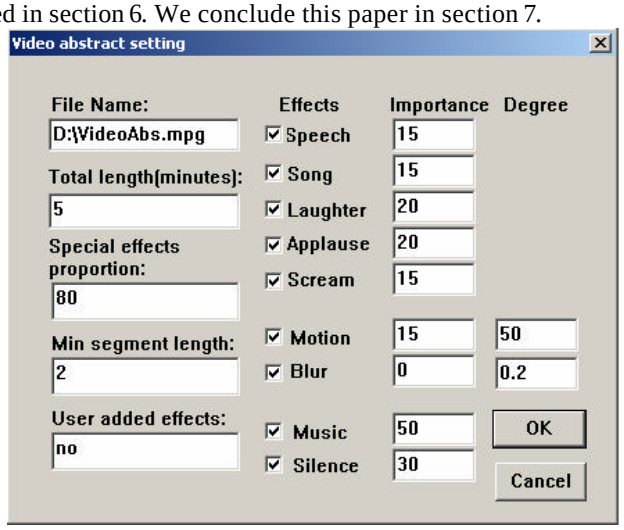

Figure 1 the input interface of the video abstraction system

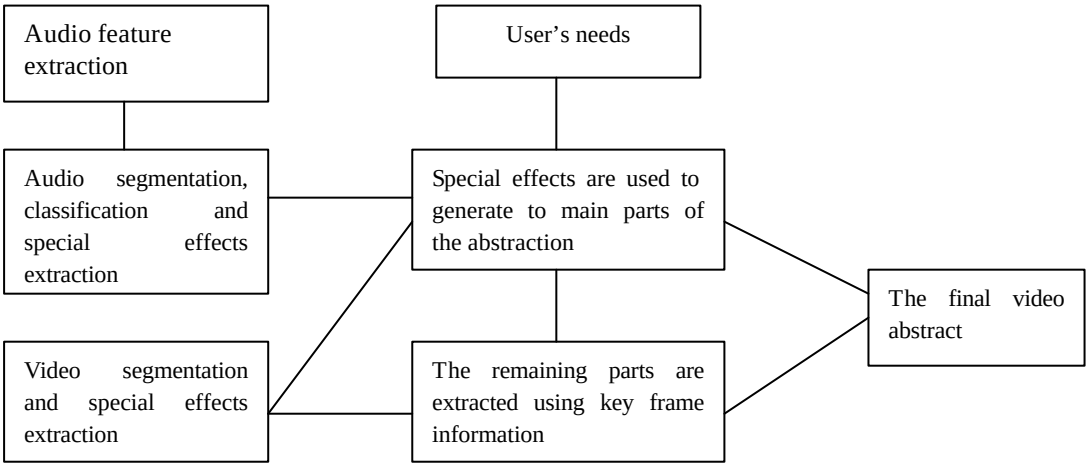

Figure 2 system overview: home video abstraction using audio contents

2. OVERVIEW OF THE VIDEO ABSTRACTION SYSTEM

As discussed in section 1, home video abstracts could be used for various purposes. So home video abstraction must meet these needs. To do this, the video abstraction system should possess flexible properties, with which the users can configure their needs to generate their particular video abstracts. In this paper, flexibility is achieved in the following way. First, a user interface is devised to capture the users' needs. This user interface contains such information as the total length, the desired special effects and their proportion in the final abstract. Figure 1 is the interface. Generally speaking, the desired special effects can be classified as audio special effects and video special events. In this paper, audio special effects are mainly used, because they are more fit and practical for home video abstraction as is discussed in section 1. Then, the special effects are extracted from the videos to generate the main parts of the final video abstract, considering the total length and their proportion. Last, the remaining part of the abstract are extracted from other parts of the video.

Although different audio contents indicate people's different events, important events or appealing scenes are usually implied by some special audio effects, which are very significant for home video abstract. As people generally

play the key role in home video abstract, sounds made by people are relatively more important. In this paper, 5 kinds of audio special effects are selected: speech, song, laughter, scream and applause. The users can choose some of them for their particular aims, or even add other audio effects such as cry, footstep, etc. The selected sounds are used in section 3 as basic types to segment and classify the audio contents. And the abstraction algorithm extracts the video clips containing these audio special effects as the main part of the abstract.

For video special effects, color features and motion features are considered in this paper. Two kinds of video special effects are used: motion level (high-activity or low-activity) and blur degree (clear or blurry). These special effects are effective and computationally efficient. Although face detection and recognition can be used to get the information about people in the videos, they are time-consuming. So this has not been added to our system now. Even without them, we have achieved good results. We will study their use in home video abstraction later.

As regards the remaining part of the video abstract, key frame selection algorithm is used. Then the contents near the key-frames are extracted. In our abstraction system, Video browsing information is simultaneously generated with the video abstracts, for the users are likely to browsing the abstract after the video abstracts are generated. Video browsing aims to provide the user with an efficient tool to grasp the main idea of videos or to find out what are the important contents. Users can use it to understand the videos quickly.

Figure 2 shows the overview of our abstraction system. First, the user input his needs using the system interface. Then audio contents are segmented and classified. Audio features are extracted for each audio segment. A two-level (frame level and model level) classification algorithm is used to classify the audio contents. And video contents are segmented and their features are extracted. After that, the main parts of the video abstract are generated using the special effects selected by the user. Finally, the remaining parts of the abstract are extracted using the key frame information.

3. AUDIO SEGMENTATION AND CLASSIFICATION

Much work had been done on audio segmentation and classification. One basic problem is speech/music discrimination [7][8][9]. But for video abstraction, only speech/music discrimination is not enough. It must distinguish a several types of sounds, i.e. speech, music, song, laughter, applause and scream etc. T. Zhang's work [10][11] is appropriate for this purpose. In order to achieve both the efficiency and accuracy of the segment classification, a hierarchy method is used in this paper. In the frame level, the segments are categorized into 4 basic types: speech, music, silence and special sound. In the model level, the special sound are further classified into 5 types, including songs, laughter, scream, applause and others.

3.1 Audio feature extraction

In this paper, three kinds of audio feature are utilized to segmentation and classification. They are short-time energy, average zero-crossing rate and fundamental frequency, as described in literature [10].

The short-time energy of an audio is defined as $E_n = \frac{1}{N}\sum_m [x(m)w(n-m)]^2$, where $x(m)$ is the discrete time audio signal, n is the time index of the short-time energy, and $w(m)$ is a window function, i.e. $w(n) = \begin{cases} 1 & 0 \le n \le N-1 \\ 0 & otherwise \end{cases}$. The energy function can not only distinguish voiced speech from unvoiced speech, but also detect silence. The short-time zero-crossing rate is defined as $Z_n = \sum_m |\operatorname{sgn}[x(m)] - \operatorname{sgn}[x(m-1)]| * w(n-m)$, where $\operatorname{sgn}[x(n)] = \begin{cases} 1 & x(n) \ge 0 \\ -1 & x(n) < 0 \end{cases}$ and $w(n) = \begin{cases} 1/2 & 0 \le n \le N-1 \\ 0 & otherwise \end{cases}$. The shot-time fundamental frequency is defined as

$F_n = fuf\{\log | FFT(x(m)w(n-m)) |\}$, where $w(n) = \begin{cases} 0.5(1-\cos(2p\frac{n}{N-1})) & 0 \le n \le N-1 \\ 0 & otherwise \end{cases}$. The operator $fuf\{\cdot\}$ estimates the fundamental frequency from the short-time spectrum. It consists of two steps. First, peaks in the spectrum which might represent the harmonics are detected. Second, it is checked whether there are harmonic relations among detected peaks.

3.2 Audio segmentation and classification

The three audio features are extracted sequentially from the audio data. Whenever there is an abrupt change in any of these three features, a segment boundary is declared.

After audio is divided into segments, segment classification is applied to classify them into different types. In order to achieve both the efficiency and accuracy of the segment classification, a hierarchy method is used in this paper.

In the first level (frame level), the segments are categorized into 4 basic types: speech, music, silence and special sounds, using the short-time audio features. The frame level classification include 5 steps: (1) separating silence; (2) separating environmental sounds with special features; (3) distinguishing music; (4) distinguishing speech; and (5) classifying other sounds as special sound. For details, we refer to [10].

In the second level (model level), the segments of special sounds are further classified into 5 types, including song, laughter, scream, applause and others, using HMM [12]. Two types of information are contained in the HMM, i.e. timbre and rhythm. Timbre is generally defined as the quality which allows one to tell the difference between sounds of the same level and loudness when made by different musical instruments or voices. Rhythm is a term originally defined for speech and music. It is the quality of happening at regular periods of time. Here, it is extended to special sound to represent the change pattern of timbres in a sound segment. Each kind of timbre is denoted by one state of HMM, and represented with the Gaussion mixture density. The rhythm information is denoted by transition and duration parameters in HMM. Once HMM parameters are set, sound segments can be classified into available classes by matching the models of these classes. We refer to [11] for details.

4. VIDEO CONTENTS PROCESSING

In order to get the information for video abstraction, the video contents must be first segmented into video shots. Then motion level and blur degree are acquired for each shot.

4.1 Shot detection

A shot designates a video sequence which was recorded by an uninterrupted camera operation. Neighboring shots are concatenated by editing effects such as hard cuts, fades, wipes or dissolves. Most of the editing effects result in characteristic spatio-temporal changes in subsequent frames of the video stream, and can therefore be detected automatically. Various methods have been proposed and implemented successfully. In our abstraction system, we used the twin comparison algorithm that detects not only simple camera breaks but also gradual transitions implemented by special effects such as fades, wipes and dissolves, initially published in [13].

4.2 Motion level

Motion level is measured by color variance. The temporal variance of mean color over all frames in a shot is used as a indicator of the scope of temporal content changes within the shot. Color variance is defined as: $s_T = \left(\frac{1}{T-1} \sum_{t=1}^{T-1} \left(m(t) - m_T \right)^2 \right)^{1/2}$, where $m_T = \frac{1}{T-1} \sum_{t=1}^{T-1} m(t)$, $m(t)$ is the mean color of frame t and T is the total number of frames in a shot. This feature can also be calculated from a sub-sampled number of frames in a shot to reduce computation time. The temporal variance has been successfully used as an activity measure to classify news video clips into anchorperson shots and news shots [14], which indicates that it is an effective feature to distinguish shots of high-activity level from those of low-activity level.

4.3 Blur degree

We used a blur-detection algorithm, proposed by Xavier Marichal et al[15], which can exploit the available DCT information of MPEG compressed videos involving a minimal computational load. It is computational efficient and its result is fairly well. The technique is based on histograms of non-zero DCT occurrences, computed directly from MPEG videos. For MPEG compressed video, the proposed algorithm is suitable for all types of pictures: I-frames, P-frames or B-frames. The objective of blur detection is to provide a percentage indicating the global image quality in terms of blur: 0% would mean that the frame is totally blurred while 100% would mean that no blur at all is present in that particular frame. This blur indicator characterizes the global image blur caused by camera motion or out of focus. Since we focus analyzing MPEG compressed video data, it is desirable that the blur indicator can be directly derived from the DCT layer of an MPEG video bit stream. It is important to select a blur indicator which is as independent as possible from the particular content of an image as well as from the type of MPEG frames (I, P or B). Intuitively, blur is the opposite of edge sharpness. DCT coefficients render this sharpness via the high values of some AC coefficients. This blur measure therefore looks for the absence of such edges into the image, which is considered to prove a blurred image. Three steps lead to the final measure:

1) In order to characterize the global blur, it is proposed to establish a measure that takes into account the DCT information of the entire image as a whole. It is likely that any type of edge will cross some 8x8 blocks at least once in the image. Globalization among all DCT blocks would therefore enable to have an idea about the general edge sharpness, i.e. the global (camera or motion) blur.
2) In order to be as independent as possible of the content of the image, coefficients should not be considered directly since their values are closely related to the type of image they depict. One rather proposes to look at the distribution of null coefficients instead of the values themselves: blur-red images are likely to have all of their high-frequency coefficients set to zero, whatever their content is.
3) In order to remove the dependency to the image size, the number of blocks in the image should divide the number of times a coefficient is not zero. This would limit histogram values to 1. However, coefficients are often zeros in P- and B-frames. In order to homogenize the look of the histogram for all types of pictures, the number of non-zero occurrences of a coefficient is divided by the number of non-zero occurrences of the DC coefficient.

In the implementation, DCT coefficients whose value is inferior to a threshold MinDCTValue are considered null. This thresholding aims at neglecting small values which may result from noise. The threshold is typically set to 8, which is the DC value of a homogeneous luminance block with intensity 1.The idea of the blur estimation algorithm is then to examine the number of coefficients that are (almost) always zero in the image, i.e. to count the number of zeroes (or nearly zero values) in DCT coefficients histograms. In practice, all values inferior to a threshold MaxHistValue are considered as not relevant for the final computation. This threshold is generally set to 0:1, i.e. only coefficients that appear 10% as often as the DC coefficient are taken into account for the blur determination. The final quality measure is obtained via a weighting grid.

5. ABSTRATION USING AUDIO CONTENTS

After the audio segmentation and classification stage, audio contents are divided into segments and each of them belongs to one of the following types: speech, music, silence, laughter, applause, scream, song, and others. As stated in section 2, speech, song, laughter, applause, scream are more important audio contents for home video abstraction. They are called audio special effects. The users can select some of them for their different use. For video contents, the users can specify the motion level and blur degree. Now task of video abstraction is how to use these audio and video contents to generate video abstract according to the user's various requirements.

The abstraction algorithm starts with a user interaction: the users need to provide the target length Len of the abstract, select the audio special effects and video special effects including video motion level and blur degree, specify a proportion a of the target length. Then the following steps are taken to generate the video abstract.

1) The length of abstract containing the audio and video effects is obtained by a proportion a of the target length. In this paper, the default value is 80%. And the user can set this value to his mind for different purpose.
2) If blur degree is specified by the user, each of the other segments containing the special effects are check to see whether their blur degrees meet the user's need. For those which do not meet the user's need, we discard them.

3) The numbers of segments containing each special effect are calculated separately. Let $N_i \quad (i \in [1..K])$ denote the number of them, where K is the number of special effects the user selected. And the total number of all the segments is $N = \sum_{i=1}^{K} N_i$.

4) The length $L_j \quad (j \in [1..N])$ of each extracted special effects segment is calculated using the results of the above steps. In the default manner, all the lengths L_j are the same, i.e. $L_j = a * Len / N \quad (j \in [1..N])$. Due to the fact that the user will generate video abstract for different purpose, special effect contents have different significance in video abstracts. So in this paper, the user can specify the importance of different special effects, which is used as weights $w_i \quad (i \in [1..K])$ for the length of each type of special effect contents. They can be seen in the user interface of Figure 1 as "Importance". Then for every segment $j \in [1..N]$ belonging to special effect type $i \in [1..K]$, its length is $L_j^i = \frac{w_i a * Len}{\sum_{i=1}^{K} w_i * N_i} \quad (j \in [1..N])$.

   Generally speaking, L_j^i should not be shorter than a minimum value, so that every extracted segment can give the user a clear meaning. The user can specify this value through the user interface as "Min Segment Length" in Figure 1. The default value in this paper is 2 minutes. If there exist some segments shorter than this minimum value, we will first delete the segment with shortest length and highest blur degree, and then recalculate the lengths. This procedure will be repeated until all the lengths are longer than the minimum value.

5) Different sounds not only indicate different events, but also imply the different time of the events to take place. As for laughter, applause and scream, the attractive events indicated by them usually take place before them. So the extracted contents should both before and at the beginning of these segments. While for speech and song, things are more complex. If the actual length of a segment is not so long, we can simply extract the contents at the middle of the segment. But if it is long enough, visual features are used to generate the extracted contents, as described at step (6).

6) After important parts are extracted, the remaining contents are extracted from those segments containing audio contents of music, silence and others. Several methods can be used for this purpose. A simple one is the average selection. But visual contents can be very helpful. So, in this paper, the key-frame selection algorithm described in literature [16] is used. Then the contents near the key-frames are extracted. Also, as is shown in Figure 1, the user can specify the importance of music, silence and others, which will be used for the resulting length of them.

7) Generate the video browsing information.

6. EXPERIMENTAL RESULTS

There is no absolute measure of the quality of a home video abstract, because different people have different aims. And even people with the same aim will give different evaluation. So the practical method is asking test persons for their evaluation. We test our algorithm in the following way.

We generated 5 abstracts of 1 to 5 minutes from five home videos ranging from 2 to 3 hours in the default way. First, 10 test persons watched each abstract. Then they browsed the original video. In the end, they were asked to give his evaluation of the abstract on a scale of 0 to 5, corresponding to total disagreement and total agreement respectively. The average evaluation is 4, 4.2, 4.2, 4.5, 4.7.

Then we let the test persons generate the home video at their will and give their evaluation. In this way, the evaluation is 4.3, 4.4, 4.7, 4.7, 4.9.

From the experimental results, we can see that the abstract algorithm works well in home videos. And with the user's interaction, we can achieve even better results. Figure 3 gives an example of video abstract for home video. The raw

video is captured during traveling. The abstract result is showed using the browsing information generated simultaneously with the abstract.

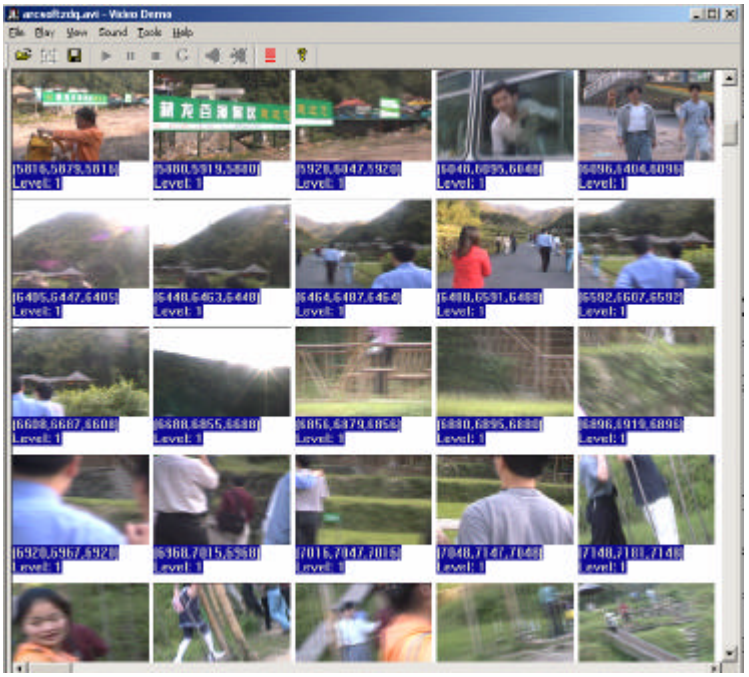

Figure 3 an example of video abstract for home video

7. CONCLUSION

This paper presents an automatic home video abstraction method mainly using audio contents. After extensive investigation, audio contents are believed to be more fit and practical for home video abstraction. So in this paper, audio contents are used as the main clues for home video abstraction. Five types of sounds (laughter, applause, scream, songs and speech) are selected as the most important audio contents. The video clips containing them are extracted as the main part of the abstract. The remaining part are extracted from music, silence and others using visual features. Experimental results show that this method can generate satisfactory home video abstracts. Future work involves improving ways to combine audio and visual contents, studying the use of face detection and recognition in the current system.

11. REFERENCES

1. S. Pfeiffer, R. Lienhart, S. Fischer and W. Effelsberg, “Abstracting Digital Movies Automatically” , Journal of Visual Communication and Image Representation, vol. 7, no. 4, pp.345-353, Dec. 1996.
2. R. Lienhart, S. Pfeiffer and W. Effelsberg, “ Video Abstracting” , Communications of the ACM, vol.40, no.12, pp.55-62, December 1997
3. M. A. Smith and T. Kanade, “ Video Skimming and Characterization through the Combination of Image and Language Understanding Techniques” , CVPR’97, pp. 775-781, 1997.
4. A. Hanjalic and H. J. Zhang, “An Integrated Scheme for Automated Video Abstraction based on Unsupervised Cluster-validity Analysis” , IEEE Transactions on Circuits and Systems for Video Technology, vol. 9, no. 8, pp.1280-1289, Dec. 1999.
5. R. Lienhart, “Abstracting Home Video Automatically” , Proc. ACM Multimedia 99, pp.37-40, Orlando, FL, October 1999
6. R. Lienhart, “ Dynamic Video Summarization of Home Video” , Proc. of IS&T/SPIE, vol.3972, pp. 378-389, Jan. 2000.

7. J. Saunders, “ Real-Time Discrimination of Broadcast Speech/Music” , Proc. ICASSP’96, vol.2, pp.993-996, Atlanta, May, 1996
8. E. Scheirer and M. Slaney, “ Construction and Evaluation of a Robust Multi-feature Speech/Music Discriminator” , Proc. ICASSP, Munich, Germany, pp.1331-1334, April,1997
9. K. El-Maleh, M. Klein, G. Petrucci, and P. Kabal, "Speech/Music Discrimination for Multimedia Applications", Proc. ICASSP, Istanbul, pp. 2445-2448, June 2000.
10. Zhang, Tong and C.-C. Jay Kuo. 1998. "Content-based Classification and Retrieval of Audio" SPIE's 43rd Annual Meeting - Conference on Advanced Signal Processing Algorithms, Architectures, and Implementations VIII, SPIE Vol.3461, p432-443, San Diego, July 1998.
11. T. Zhang and C.-C.J. Kuo, “ Hierarchical Classification for Audio Data for Archiving and Retrieving,” Proc. ICASSP, Phoenix, vol. 6, pp. 3001-3004, Mar, 1999
12. L. Rabinar, B. Juang, Fundamentals of Speech Recognition, Prentice-Hall, Inc., New Jersey, 1993.
13. H.J. Zhang, A. Kankanhalli and S.W. Smoliar. Automatic partitioning of full-motion video. Multimatia Systems, 1(1):10--28, 1993.
14. H.J Zhang, S.Y. Tan, S.W. Smoliar, and G. Yihong. Automatic Parsing and Indexing of News Video. Multimedia Systems, 2(6):256--266, January 1995.
15. Xavier Marichal, Wei Ying Ma and H.J. Zhang. Blur Determination in the Compressed Domain Using DCT Information. ICIP'99, Kobe, September 1999, Proc. Vol. II, pp. 386-389.[14]
16. H.J. Zhang, J.H. Wu, D. Zhong, S.W. Smoliar, ”an Integrated System for Content-based Video Retrieval and Browsing” , Pattern Recognition,Vol.30,No.4,pp.643-658,1997