Key Frame Extraction Towards Kernel-SIFT Identification

Zhenyu Wu[1], Ruiqing Wu[1], Hongyang Yu[2], Bin Tang[1]

[1]Department of Electronic and Engineering, University of Electronic Science and Technology of China 611731, China
[2]Research Institute Electronic Science and Technology, University of Electronic Science and Technology of China 611731, China
zywu@uestc.edu.cn, rqwu@uestc.edu.cn, hyyu@uestc.edu.cn, bint@uestc.edu.cn

Abstract - Key frame detection is a fundamental technique in video retrieving and video summarizing. In this paper, an effective and efficient content-based key-frame detection method is proposed. This method is based on content analysis and features detection from key frames. There are two main contributions in the proposed method. Firstly, shot cuts dividing scheme is studied here to exclude transition frames (such as fade in and fade out frames), which may disturb key-frame detection. On the other hand, key frames are selected according to each frame's status in the whole shots, which is determined by kernel-SIFT matching rate. Experimental results show that the proposed method can extract the most effective key frames, which have represented the whole shots very well and contained significant features than other histogram-based methods.

Index Terms - key frame, SIFT, shot cut detection

I . Introduction

In digital multimedia flooding age, millions of cameras over the world capture a gigantic amount of video data every day and raises new challenges: How to solve the mass video data's management and storage problems? How to edit such immense video data to produce programs? How to retrieve video streams efficient in the huge video stream pool? Hence it is valuable to allow people and systems to retrieve or gain certain perspectives of a video without traversing all the video data. In content-based video retrieval and video summarizing area, key frame is used to describe the key image of a shot, which usually reflects the main elements of a shot. Also, a user can complete quick-view for the entire video content through key frame set. So the result of key frame extraction plays an important role in content-based video retrieval and video summarization.

Generally, there are two kinds of content-based key-frame detection algorithms, besides manual annotation. The one is shot boundary detection based static approach, which will take fixed positions in shots as the key frames (such as the first, middle or the last frame of one shot) [1]. This kind of approach is simple, but too depended on shot boundary detection. Moreover, gradual shot transitions will degrade this kind of key frame detection approach. The other one is dynamic algorithm, which extracts key frames by calculating inter-frame differences according to various kinds of histograms (such as global, local and edge histogram) or taking visual attention features [6] [7]. However the above general kinds of key frame detection approaches are not efficient enough, because boundary detection is a hard open issue by itself and determining key-frames by inter frame's differences depend heavily on thresholds' choosing and those methods will be affected heavily by illumination change and scale. So these methods can hardly guarantee the extracted key frames' number (either too much or too little) and their qualities (such as gradual shot transitions or unclear frames may disturb video summarize and video retrieval). Since the number of key frames and their qualities directly determined the performance (speed and accuracy) of video retrieval and video summarization, it is urgent to produce new effective key-frame detection algorithm.

This paper proposes a new key-frame extraction algorithm, which is based on content analysing and frame status determining. This algorithm aims to pick out key frames which can best represent the whole shots and possess the most significant features in the whole shots. Firstly, we take the second derivative of variance in the current frame to segment video streams into shot cuts coarsely and exclude the gradual shot transition frames. Secondly, we extract significant features of every candidate frame within one shot by proposed kernel scale invariant feature transform and select the frame with the most common features in the same shot or sub-shot as the key frame. The algorithm has two main advantages. Firstly, it processes the video in an online fashion (i.e. entire video does not need to be available a priori). Secondly, the algorithm can extract more appropriate key frames for multimedia applications especially for video search and summarization.

The rest of this paper is organized as follows. Section II provides a brief review of general existing approaches to key frame detection. Section III describes the details of the proposed key frame extract algorithm by features detection and frame's status consideration. Section IV presents the experimental results to demonstrate the performance gain of the proposed algorithm. Finally, Section V concludes this paper with a summary.

II . Existing approaches in key frame extraction

In the previous work, many researchers have deeply studied shot boundary detection based key frame extraction algorithms. Regularly, a boundary detection algorithm is first used to detect the shots followed by a key frame extraction. In [1], the first frame of the shot is selected as the key frame. Zhuang et al. in [2] assume that the shot boundaries have been detected and use an unsupervised clustering algorithm to find the key frame of the given shot. This approach has two limitations: the dependency on the boundary finding algorithm and the high computational complexity. Song and Fan in [3] use a unified spatio-temporal feature space to characterize

video data and extract key frames by calculating FF-ISF. Maria [4] proposed a real-time key frame extraction method by DCT coefficients, which can reduce computational cost with the expense of accurate. An entropy-based method was introduced in [5] where the entropy of a grayscale frame is computed and compared with that of the previous frame.

On the other hand, some research focused on combining shot detection and key-frame extraction. Such as Liu et al.[7] proposed a joint frame-work to integrate both shot boundary detection and key frame extraction by a probabilistic model, and treated the key frame extraction as a maximum posterior problem which can be solved by adopting alternate strategy. A.Wael [6] proposed a novel method to detect show boundaries and extract key frames of video data simultaneously based on a sliding window Singular Value Decomposition approach.

III. Proposed Key frame Extraction Algorithm

In the proposed algorithm, shot cut detection is taken as the first step before key frame extraction. Different from conventional approaches, during shot cut detecting we will exclude those frames which are meaningless (black screen) and the gradual shot transitions or unclear frames, since they cannot represent shots.

The kinds of shot changing are sudden changing, dissolving and fading, which can be modelled as following:

$$H_{sudden}(n)=\begin{cases} f(n) & n<K \\ g(n) & n\geq K \end{cases} \tag{1}$$

$$H_{dissolve}(n)=\begin{cases} f(n) & n<K \\ (1-\alpha(n))f(n)+\alpha(n)g(n) & K\leq n<L \\ g(n) & n\geq L \end{cases} \tag{2}$$

$$H_{fadein}(n)=\begin{cases} f(n) & n<K \\ (1-\alpha(n))C+\alpha(n)f(n) & K\leq n<L \\ g(n) & n\geq L \end{cases} \tag{3}$$

$$H_{fadeout}(n)=\begin{cases} f(n) & n<K \\ (1-\alpha(n))f(n)+\alpha(n)C & K\leq n<L \\ g(n) & n\geq L \end{cases} \tag{4}$$

where $H(n)$ is the n^{th} frame in video sequences, $f(n)$ and $g(n)$ denote the continuous two shots with variance ∂_f^2 and ∂_g^2. K to L are shot changing frames, $\alpha(n)=\frac{n-K}{L-K}$, C is monochrome frame. Studying the first and second derivative of frame's variance (shown in (5) - (8)), we can detect shot cuts and exclude those meaningless (black screen) and the gradual shot transition frames easily.

$$\Delta^2\partial^2_{Sudden}(n)=\begin{cases} 0 & n<K-1 \\ \partial_g^2-\partial_f^2 & n=K-1 \\ \partial_f^2-\partial_g^2 & n=K \\ 0 & n>K \end{cases} \tag{5}$$

$$\Delta^2\partial^2_{dissolve}(n)=\begin{cases} 0 & n<K \\ \frac{\partial_f^2+\partial_g^2}{(L-K)^2}-\frac{2\partial_f^2}{L-K} & n=K \\ \frac{2(\partial_f^2+\partial_g^2)}{(L-K)^2} & K+1\leq n<L \\ \frac{\partial_f^2+\partial_g^2}{(L-K)^2}-\frac{2\partial_g^2}{L-K} & n=L \\ 0 & n>L \end{cases} \tag{6}$$

$$\Delta^2\partial^2_{fadein}(n)=\begin{cases} 0 & n<K-1 \\ -\partial_f^2 & n=K-1 \\ \frac{\partial_g^2}{(L-K)^2}+\partial_f^2 & n=K \\ \frac{2\partial_g^2}{(L-K)^2} & K+1\leq n<L \\ \frac{\partial_f^2}{(L-K)^2}-\partial_g^2 & n=L \\ 0 & n>L \end{cases} \tag{7}$$

$$\Delta^2\partial^2_{fadeout}(n)=\begin{cases} 0 & n<K \\ \frac{\partial_g^2}{(L-K)^2}+\partial_f^2 & n=K \\ \frac{2\partial_f^2}{(L-K)^2} & K+1\leq n<L \\ \partial_g{}^2 & n=L-1 \\ \frac{\partial_f^2}{(L-K)^2}-\partial_g{}^2 & n=L \\ 0 & n>L \end{cases} \tag{8}$$

In the key-frame extract step, different from those histogram-based conventional approaches, we design a kernel SIFT feature extraction algorithm here. The proposed algorithm shall pick out key frames and generate key features for video retrieval at the same time.

Our algorithm for local descriptors accepts the input as standard SIFT descriptor: the sub-pixel location, scale, and dominant orientations of the key point. And then project the gradient image descriptor into a kernel-space to derive a compact feature vector. This feature vector is significant smaller than the standard SIFT feature vector, and can be used with the same matching algorithms. The Euclidean distance between two feature vectors is used to determine whether the two vectors correspond to the same key point in different images.

Kernel-space we defined here is spanned by feature vectors which are selected according to the matching rate. It is gotten as follows: (1) calculate Euclidean distance between key point descriptors among candidate frames. (2) rank them by

matching rate, and select the top *n* (most of the results described in this paper use *n*=50) key point descriptors to span the kernel-space. (3) store this projection matrix into memory.

N-element kernel-SIFT is gotten by projecting the 128-element SIFT vectors into kernel-space. Finally, frames with most matches will be chosen as key frame in the current video shot.

To apply the proposed key frame extraction algorithm into video retrieval application, we shall do following steps:

(1) Generate the kernel-spaces and store the corresponding kernel-matrixes for each seed video steam's shot cuts.

(2) Project 128-element SIFT descriptors by kernel-space to get N-element Kernel-SIFT descriptors for each candidate frames.

(3) Do match processing, and then choose the frame with most matches as key frame in the current video shot.

(4) Calculate the largest Kernel-SIFT distance between key frame and other frames within one video shot, and then store it for threshold generating.

(5) Do shot cut determining for video streams to be searched, and exclude those frames not suitable to be key frames.

(6) Arbitrarily pick up one frame in each shot cut, and generate their Kernel-SIFT descriptors by stored Kernel matrixes of seed video streams.

(7) Calculate the Kernel-SIFT distances between key frames and frames from streams to be searched.

(8) Sequence seed streams and the streams to be searched according to the Kernel-SIFT distances.

IV. Experimental Results

To evaluate the performance of the proposed key frame extraction algorithm, five hundreds video streams are selected, including various contents such as news, sports, movie, cartoon, natural scene, face etc. The experiments were performed on i5 computer, running Windows 7. In order to evaluate the performance of the algorithm we used definitions of detection recall and precision as shown in Equation (9). In our simulation we obtained an average recall of 97% and average precision of 100%.

$$recall = \frac{\#\ frames\ shared\ by\ detected\ submitted\ transition\ \&\ matching\ ref.\ transition}{\#\ frames\ of\ detected\ reference\ transition}$$

$$precision = \frac{\#\ frames\ shared\ by\ detected\ submitted\ transition\ \&\ matching\ ref.\ transition}{\#\ frames\ of\ detected\ submitted\ transition} \quad (9)$$

Figure 1 shows the result of applying the shot cut detection algorithm mentioned in section III on a typical stream. Studying it, we can segment video stream into shot cuts accurately and ignore those unsuitable frames easily. For space limited, Figure 2 demonstrates the key frames extracted by the proposed algorithm, method in [1] which takes fixed frame position during shots and method described in [6] which extracts key frames of video data simultaneously based on a sliding window SVD for four video segment (Obama's inaugural address, soccer game, car racing and glasgow). Since method in [1] took the first frame each shot cut as the key frame, it cannot avoid taking transition frames as key frames. As displayed in Figure 2, there are several dissolving frames in video segments are chosen as key frames. SVD method in [6] is affected heavily on the search window length and shot cut changing frequency. There will be key frames missing in the case that shot cut changing frequency smaller than search window length. Moreover, both methods do not consider the status of frames, so they can hardly extract more representtable frames as key frames. The proposed key frame detection project those vectors to get Kernel-SIFT vectors, and determine the key frames by matching rate.

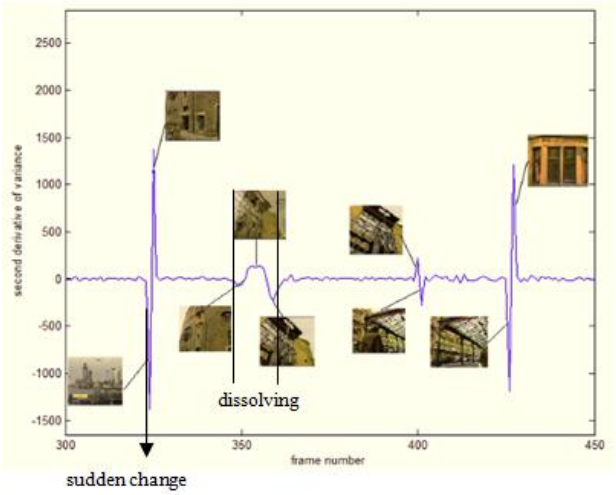

Fig. 1 shot cuts detection result

(a) Method in [1]

(b) SVD (N=15,τ=0.25) method in [6]

(c) Proposed algorithm

(d) Manual annotation

Fig. 2 Extracted key frames of four different kinds of video segments

The proposed algorithm can be used for online processing of streamed videos, since it does not require the entire video for processing. Moreover, this algorithm can extract the least number of key frames which have the most common features and not sensitive to scale, illumination change. Experimental results show that the proposed algorithm is robust to a wide range of digital effects of shot transitions, and can extract the most meaningful key frames similar with manual annotation compared with those histogram-based or SVD-based methods.

V. Acknowledgment

The work is supported by Basic Operating Expenses of the Chinese Universities at ZYGX2012J024

References

[1] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," in Second working Conference on Visual Database Systems, 1992.

[2] Z. Yueting, R. Yong, T. S. Huang, and S. Mehrotra, "Adaptive key frame extraction using supervised clustering," Proceeding in IEEE ICIP, 1998.

[3] X. Song and G. Fan, "Key-frame extraction for object-based video segmentation," in IEEE Proc. Int. conference on Acoustics, Speech and Signal Processing, 2005.

[4] M. chatzigiorgaki and A. N. Skodras, "real-time keyframe extraction towards video content identification," Proceeding of 16th International conference on Digital Signal Processing, 2009.

[5] M. Mentzelopoulos and A. Psarrou, "Key-frame extraction algorithm using entropy difference," in Proceedings of the ACM SIGMM International workshop on Multimedia Information Retrieval, 2004.

[6] A. Wael, "online, simultaneous shot boundary detection and key frame extraction for sports videos using rank tracing," in Processing of IEEE ICIP, 2008.

[7] H. Liu, W. Meng, Z. Liu, "Key frame extraction of online video based on optimized frame difference," Proceeding of 19th IEEE ICFSKD, 2012.