ONLINE, SIMULTANEOUS SHOT BOUNDARY DETECTION AND KEY FRAME EXTRACTION FOR SPORTS VIDEOS USING RANK TRACING

Wael Abd-Almageed

Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742
wamageed,lsd@umiacs.umd.edu

ABSTRACT

In this paper, we present a novel algorithm for simultaneously detecting shot boundaries and extracting key frames from video sequences or streams in real-time. Multivariate feature vectors are extracted from the video frames and arranged in a feature matrix. Singular value decomposition is then used to factorize the feature matrix and compute the significant singular vectors. The rank of the singular vectors is traced using a sliding window approach. By tracing the computed rank, we are able to determine shot boundaries and extract key frames of the video. Results of experiments conducted on soccer videos show that the algorithm is robust to a wide range of digital effects used during shot transition. Moreover, the algorithm is shown to run in real time making it suitable for online video summarization and multimedia networking applications.

Index Terms— Video summarization, shot detection, Singular Value Decomposition

1. INTRODUCTION

Detecting shot boundaries and extracting key frames from video data are fundamental steps for various video processing applications. In content-based video retrieval and video summarization applications, for example, it is unequivocally important to efficiently perform such tasks.

Many methods exist for shot boundary detection and key frame extraction. However, two limitations are usually noticed, when these methods are used to process sports videos. Firstly, many methods are highly computationally expensive and cannot be used in real-time and/or online video processing application. Secondly, the digital video effects (gradual shot transitions, logo wipes, etc.) used in sports production usually degrade the performance of such methods.

In this paper, we present a novel shot boundary detection and key frame extraction algorithm based on Singular Value Decomposition (SVD). The algorithm extracts low-cost, multivariate color features from the video and constructs a 2D feature matrix. The matrix is then factorized using Singular Value Decomposition. Based on the traced rank, shot boundaries and key frames are extracted. The algorithm has two main advantages. First, it processes the video in an online fashion (i.e. entire video does not need to be available a priori). Second, the algorithm has low computational requirements. Hence, the algorithm is very suitable to video streaming and other multimedia networking applications.

2. RELATED WORK

Even though shot boundary detection and key frame extraction are strongly related, the two problems have usually been treated disjointly. 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 the video data. They jointly perform key frame extraction and object segmentation by maximizing the divergence between objects in the feature space.

In [4], Rong et al. uses a text retrieval method to extract key frames. Each frame is thought of as a word in a document (shot) and the most representative words of a given document are selected as key frames. 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. If the entropy difference is higher than a user defined threshold, then the new frame is assumed to be a key frame.

Gong and Liu in [6] use SVD for video summarization. SVD was applied to a feature matrix constructed from 10% of the total video frames. The rest of the frames were then clusterized based on the output of SVD. Two limitations exist for this algorithm – the need to a priori have the entire video and if the video is too long, computing SVD followed

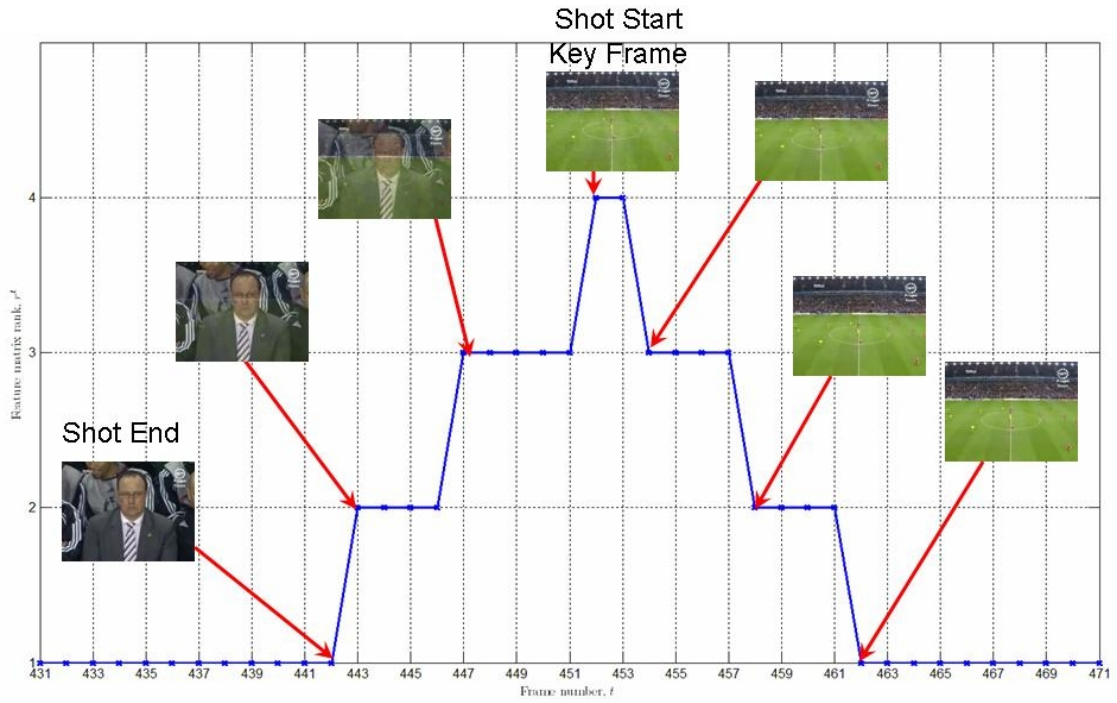

Fig. 1. A soccer video clip illustrating a 10-frame dissolve digital effect. The visual content of the video frames slowly changes during the transition. Consequently, the rank r^t of the feature matrix $\mathbf{X}^t$ increases a few times. Non of the transition frames can be used as a key frame. Algorithms that extract the first frame of every shot as a key frame fail in such situations.

by clustering becomes very computationally expensive.

3. EXTRACTING KEY FRAMES USING RANK TRACING

Singular Value Decomposition is an important matrix factorization technique with numerous applications in multimedia processing. To extract the key frames of a video stream, we use a sliding-window Singular Value Decomposition (sw-SVD) approach. Let F^t be the video frame at time t represented in the Hue-Saturation-Value (HSV) color space. Three histograms, $\mathbf{h}_H$, $\mathbf{h}_S$ and $\mathbf{h}_V$, of lengths l_H, l_S and l_V, respectively, are computed for the three color channels of F^t. We now construct a time-varying feature vector $\mathbf{x}^t$ as shown in Equation (1).

$$\mathbf{x}^t = [\mathbf{h}_H \; \mathbf{h}_S \; \mathbf{h}_V] \tag{1}$$

The vector x^t is of length $L = l_H + l_S + l_V$. For each frame at time $t > N$, we build a $N \times L$ feature matrix $\mathbf{X}^t$ as shown in Equation

$$\mathbf{X}^t = \begin{bmatrix} \mathbf{x}^t \\ \mathbf{x}^{t-1} \\ \vdots \\ \mathbf{x}^{t-N+1} \end{bmatrix}, \tag{2}$$

and $\quad t = N, \ldots T$

where N is a window width and T is the total number of video frames. In other words, $\mathbf{X}^t$ is a time-varying feature matrix representing the features of the current frame and previous $N-1$ frames (hence the name sliding window SVD.) Only color features are used because they are computationally cheap to compute.

The matrix $\mathbf{X}^t$ is factorized using SVD as shown in Equation (3)

$$\mathbf{X}^t = \mathbf{U} \, \Sigma \, \mathbf{V}^T \tag{3}$$

where $\mathbf{U}$ is a matrix of a set of output orthonormal singular vectors of $\mathbf{X}^t$, $\mathbf{V}^T$ is a matrix of a set of input orthonormal singular vectors of $\mathbf{X}^t$ and Σ is a matrix of the singular values of $\mathbf{X}^t$. The diagonal elements of Σ depict the significance of each of the singular vectors of $\mathbf{V}$, and are ordered in a non-increasing order.

Let the singular values be $s_1, \; s_2, \; \ldots, \; , s_N$, with s_1 being the maximum singular value. The rank of $\mathbf{X}^t$ can be estimated as the number of singular values that exceed a user-defined threshold, τ, of the maximum singular value, s_1. In other words, the rank r^t of $\mathbf{X}^t$ is the number of s_i satisfying the condition $\frac{s_i}{s_1} \geq \tau$.

Tracing the computed rank over time reveals two important characteristics about the underlying video stream. First, if the rank of the current feature matrix, $\mathbf{X}^t$, is greater than that of the previous matrix, $\mathbf{X}^{t-1}$, then the visual content of

the current video frame is sufficiently different than the content of the previous frame. Therefore, it cannot be represented using the singular vectors computed at $t-1$. Second, if the rank of the current feature matrix, is smaller than that of the previous matrix, i.e. $r^t < r^{t-1}$, then the visual content of the video stream has been stable long enough to "forget" the previous shot and/or the digital effects during the transition.

Based on these observations, we can draw two conclusions. First, since the rank might increase many times during a a transition, the frame at which the computed rank is maximum is the starting frame of the shot, and can be extracted as the key frame of the shot. Second, the frame at which $r^t > r^{t-1}$and$r^{t-1} = 1$ is the ending frame of the shot.

Finally, in order to initialize the algorithm, the first N frames are used to compute $\mathbf{X}^{t=N}$ and the main algorithm loop starts at $t = N + 1$.

Figure 1 illustrates the idea. At frame 443, a 10-frame dissolve digital effect is used to switch from a close-up "coach" scene to a "wide-field" view. As the dissolve takes place, the rank gradually increases until frame 452. Therefore, frame 442 is marked as the end of the outgoing shot, and frame 452 is considered both the starting frame and key frame of the incoming shot.

4. EXPERIMENTAL RESULTS

The shot boundary detection and key frame extraction algorithm was implemented using Matlab on a 2.16 GHz Pentium D computer, running Microsoft's Windows XP. Processing 320×240 videos, the code runs at 15 frames per second (fps).

Figure 2 shows the result of applying the proposed algorithm on a soccer video from the Europeans Champions League. Despite the heavy use of digital video effects, the algorithm produces correct key frames. For example, the producer used a 10-frame dissolve effect to switch from the coach scene (key frame number 244) and the wide-angle field view (key frame 452), as shown previously in Figure 1. However, the algorithms selects a stable key frame and shows robustness to the digital effects used by the producer.

In Figure 3, we show the results of applying our algorithm on a game taken from the 2006 World Club Championships. Prior to the beginning of the game, the producer artistically switches between scenes of the stadium, the competition's banner, some highlights, the competitions logo, etc. The algorithm accurately extracts the correct frames from the series of the shots.

Due to the lack of ground-truth sports videos, we manually labeled 90 minutes of soccer videos from the UEFA Champions League (30 minutes/game for 3 games.) In order to evaluate the performance of the algorithm we used TRECVID [7] definitions of detection recall and precision as shown in Equation (4). Using a a window of width $N = 12$ and threshold $\tau = 0.25$, we obtained an average recall of 93.7% and average precision of 100%. We are currently building a larger ground truth data set for a more thrurough evaluation of our algorithm.

5. CONCLUSIONS AND FUTURE WORK

We have presented an online algorithm capable of detecting show boundaries and extracting key frames of video data simultaneously. The algorithm is based on a sliding window Singular Value Decomposition approach. Color features are used to construct a feature matrix. Sliding window SVD is then used to compute the rank of the current feature matrix. By analyzing the evolution of the computed rank over time, we are able to detect the boundaries of the video shots and extract representative key frames.

The algorithm can be used for online processing of streamed videos since it does not require the entire video for processing. Moreover, the algorithm runs in real-time on regular computers. Experimental results show that out algorithm is robust to a wide range of digital effects used in shot transitions. Consequently, the algorithm is powerful in selecting key frames of sports videos where digital effects are frequently used.

In order to further extend this work, we are investigating other rank-revealing methods in order to further improve the accuracy and efficiency of our approach.

6. 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," in *IEEE International Conference on Image Processing*, 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] J. Rong, W. Jin, and L. Wu, "Key frame extraction using inter-shot information," in *IEEE International Conference on Multimedia and Expo*, 2004.

[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] Y. Gong and X. Liu, "Video summarization using singular value decomposition," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2000.

[7] Text Retrieval Conference: Video Retrieval, "http://www-nlpir.nist.gov/projects/t2002v/sbmeasures.html," .

$$\text{frame recall} = \frac{\#\text{ frames shared between detected and reference transitions}}{\#\text{ frame of reference transitions}}$$
$$\text{frame precision} = \frac{\#\text{ frames shared between detected and reference transitions}}{\#\text{ frame of detected transitions}} \tag{4}$$

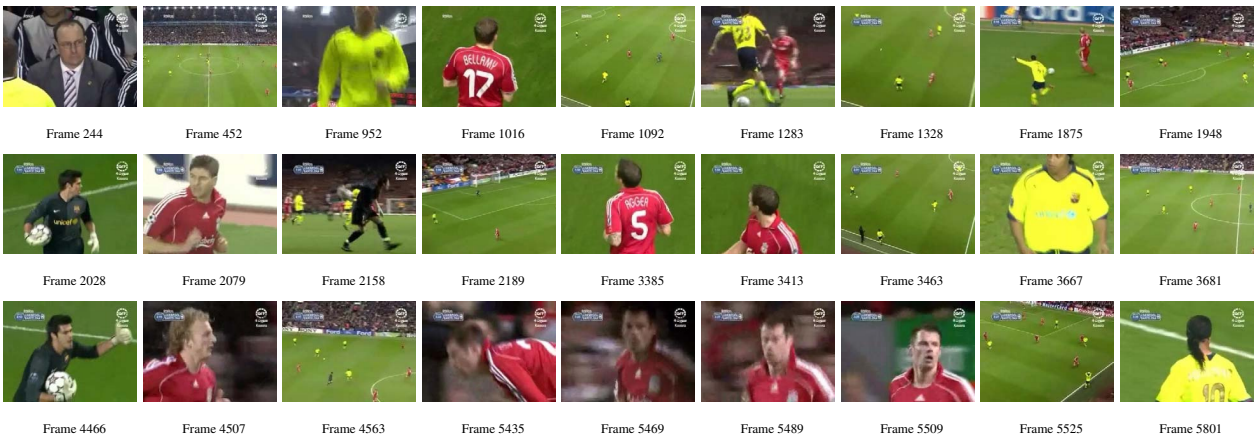

Fig. 2. The 27 key frames of sequence of 6000 frames. The sw-SVD algorithm extracts the correct key frames and shows robustness against digital effects. Figure 1 shows the dissolve digital effect used during transition from the close-up coach scene to the wide-angle field scene. None of the intermediate frames is extracted as a key frame even with significant change of the visual content

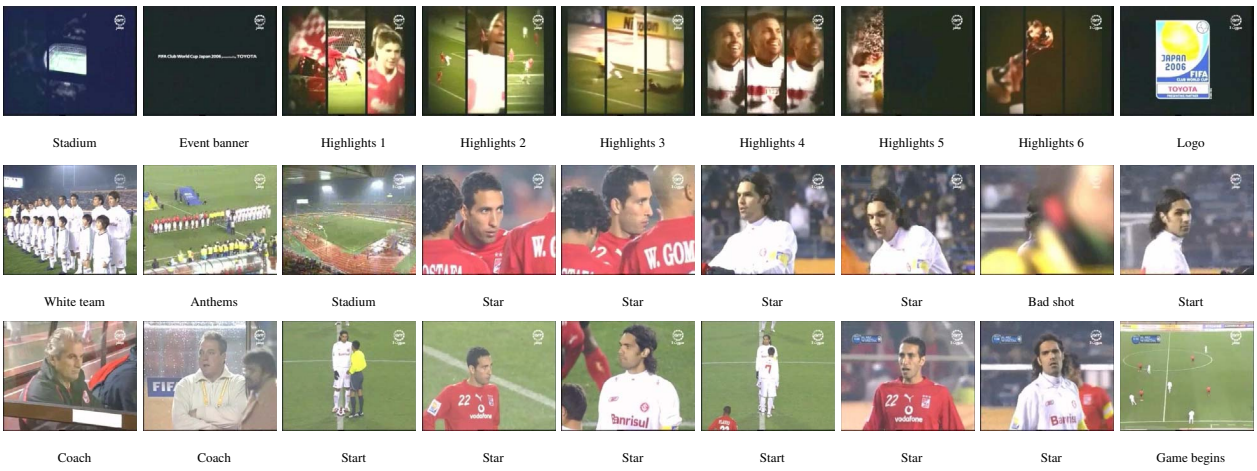

Fig. 3. The 27 key frames of a sequence of 7500 frames taken from pre-game video.