A New Technique for Shot Detection and Key Frames Selection in Histogram Space

Seung Hoon Han, Kuk Jin Yoon, In So Kweon[o]
Robotics & Computer Vision Lab. , EE. Dept., KAIST 373-1 Kusong-dong, Yusong-ku, Taejon, 305-701, Republic of Korea
shhan@covra1.kaist.ac.kr, kjyoon@covra1.kaist.ac.kr, iskweon@cais.kaist.ac.kr

Abstract

A video stream consists of a number of shots each of which has the boundary property, such as cut, fade, dissolve, wipe, etc. The shot boundary such like cut is detected easily through any previous works. However, the videos with more than two types of shot and the large motion of camera or objects have difficulties in extracting the boundary between the adjacent shots. False alarms are increased in such a shot with camera and object movements. This paper proposes the shot detection algorithm which accentuates edit constancy effects while suppressing motion effects using low pass filtering in histogram space. Edit constancy effects are defined as shapes of cut or fade/dissolve in low pass filtered frame differences signal. And this paper also presents the shot representation method selecting key frames effectively based on contents within a shot.

1. Introduction

A shot is defined as the consecutive frames from the start to the end of recording in a camera. It shows a continuous action in an image sequence. The cut boundaries show an abrupt change in image intensity or color, while those of fades or dissolves show gradual changes between frames. The detection of the latter is more difficult than the detection of the former. The image motion due to camera or object movements makes the shot boundary detection problem more complex. There is also a slow change in intensity for the frames with image motion. These may take their changes as shot boundaries resulting in false alarms which cause degrade the precision.

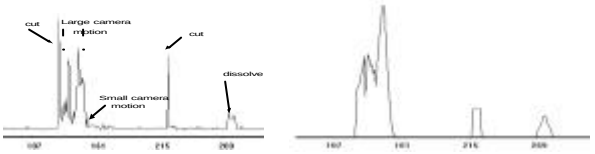
Twin-comparison [1] was developed to find shot boundaries among cuts and fades/dissolves using two thresholds. Gunsel and Tekalp [4] proposed one threshold method using Otsu method to find the threshold automatically. However, this system was presented for detection of cut-type shot boundaries. In model-based method [3,5], the edit effect showing gradual changes(fades, dissolves, etc) presents edit invariant property that is used in classifying shot boundaries.

This paper proposes a framework in which accentuates edit constancy effects by applying low pass filtering to histogram differences between frames, while suppressing motion effects causing false alarms. Edit constancy effects are rectangular shapes of cut and triangular shapes of fades/dissolves in filtered histogram differences after applying window convolution to original histogram differences. And this paper also presents the shot representation method showing the key frames effectively based on contents in each shot. The most common method to select key frames is the temporal sampling method. But this method does not provide the successful representation in general. This paper presents in details the shot detection method in section 2 and the key-frame selection method in section 3.

2. Shot boundary detection in low-pass filtered histogram space

First, using equation (1), color histogram differences between adjacent frames are calculated and showed in Fig. 1(A). G is the number of histogram bins. Fig. 1(A) shows that the sharp peaks are presented in cut boundaries and their values are large relative to other boundary values. In dissolves or fades, the differences and the rate of change are small. However, the differences values don't make constant plateau. Fig. 1(A) also shows the differences between frames with large camera motion are changed arbitrarily in magnitude, and are falsely detected as cut-type boundaries. These differences due to the large camera motion are detected as false alarms in most shot detection systems without motion analysis. Now, the

differences in frames with small camera motion are smaller than those of frames with large camera motion, but are comparable to those of gradually changed frames, such as dissolve frames. These also result in false alarms. Fig. 1(B) shows the filtered frame differences which are generated by average-clipping and then applying a local window convolution to Fig. 1(A). This figure represents clear shapes of each shot boundary, e.g., the rectangular shape for cut-type boundaries and the triangular shapes for dissolve-type boundaries. This paper defines edit constancy effects as the rectangular shape for the cut and the triangular shape for gradually scene change, such as fade or dissolve.



(A) (B)

Fig. 1 (A) Histogram differences between frames (B) Low-pass filtered histogram differences between frames

The color histogram differences between adjacent frames are defined as the following.

$$fd[i] = \sum_{k\in(R,G,B)} \sum_{j=1}^{G} \left| H^k{}_i(j) - H^k{}_{i-1}(j) \right| \tag{2}$$

In an ideal case, the cut boundary showing abrupt frame change is represented by delta function like equation (3). The gradual scene change such as fade or dissolve shows the form of equation (4) if the differences between frames are constant during the transition between two shots.

$$fd_c[i] = a_i d(i - i_c) \quad \text{for cut boundary} \tag{3}$$

$$fd_e[i] = b_i rect\left(\frac{i - i_e}{T_e}\right) \quad \text{for gradual scene changes}$$
$$rect(x) = \begin{cases} 1, & |x| \le 1/2 \\ 0, & |x| > 1/2 \end{cases} \tag{4}$$

Noises and motion in an image can be reduced by clipping fd[i] by average of frame differences. This clipped signal $fd_{clip}[i]$ is represented in equation (5).

$$fd_{clip}[i] = \begin{cases} fd[i] - avg\{fd[i]\} & \text{if } fd[i] > avg\{fd[i]\} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Convolving the $fd_{clip}[i]$ by the window whose width is M and magnitude 1/M, we get equation (6) generating the Fig. 2(B). After this convolution process, the cut boundary has the form of equation (7) and the gradual scene change has the form of equation (8).

$$fd_{conv}[i] = fd_{clip}[i] * \frac{rect\left(\frac{i}{M}\right)}{M} \tag{6}$$

$$fd^{cut}_{conv}[i] = \frac{a_i}{M} rect\left(\frac{i - i_c}{M/2}\right) \tag{7}$$

$$fd^{edit}_{conv}[i] = \frac{b_i}{M} tri\left(\frac{i - i_c}{M/2}\right)$$
$$tri(x) = \begin{cases} 1 - |x|, & |x| \le 1 \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

Note that the rectangular shape is generated for the cut boundary from equation (7) and the triangular shape is generated for the boundary with gradual scene changes, fade or dissolve from equation (8). We define these shapes as edit constancy effects. After convolution, Otsu method is applied to find one threshold that is used to determine shot boundaries above threshold. For the cut, the center of a rectangle is declared a boundary frame, while the two ends of tail in a triangle are declared two boundaries for gradual scene changes, fade/dissolve. The optimal threshold is found by minimizing equation of Otsu method (9).

$$s_W^2(t) = q_1(t)s_1^2(t) + q_2(t)s_2^2(t)$$
$$q_1(t) = \sum_{i=0}^{t-1} p(i), \; q_2(t) = \sum_{i=t}^{N} q(i)$$
$$s_1^2(t) = \text{the variance of the fd[k] in the first cluster}$$
$$s_2^2(t) = \text{the variance of the fd[k] in the second cluster} \tag{9}$$

3. Key frames selection using adaptive temporal sampling

There are two commonly employed representations of video contents: shot-based and object-based. The efficient shot representation and visualization is important for indexing performance and user interface. In the shot-based video indexing system, the representation of the visual contents in a shot is generally achieved by using key frames. For some applications, such as news video indexing, a single

frame may be sufficient to represent the contents of the entire frames in a shot. However, if the shot has more complex contents, more key frames are needed. The most common method to select key frames is the temporal sampling method. This method is easy to use and fast. But this method does not provide the successful representation in general because it does not consider the variation of the contents within a shot. The other approaches, such as clustering [2] and automatic threshold method [4], were proposed. These methods select the frames that have the large difference to the previous frame in color as the key frames. But, to represent the shot contents as a whole, key frames must contain the event flow within the shot, although there are not large differences between successive frames. In the Fig. 1, region 1 and region 3 have the larger color change between successive frames than region 2. This implies that there may be some meaningful events in the region 1 and region 3. If we select key frames by temporal sampling(Fig. 2(A)), we have small number of frames in the region 1 and region 3 but large number of frames in the region 2. This sampling method is not efficient, and the selected key frames do not provide successful representation of the shot. Also, if we use the automatic threshold method, we couldn't obtain any information about region 2.

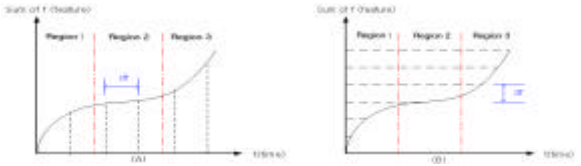


Fig. 2: Temporal sampling (A) and Adaptive temporal sampling(B)

To overcome these limitations, we propose an algorithm, called adaptive temporal sampling. We use color histogram differences which are obtained in a shot change detection process as feature data, given by equation (2). The sum of feature is defined the accumulated value of the color histogram differences in a shot. In the Fig. 2, y-axis denotes the sum of feature value. The adaptive temporal sampling method results in the key frames that are sampled at the constant interval in y-axis. So automatically we can get a larger number of frames in the region 1 and region 3 and small number of frames in the region 2. In other words, we select more frames in a rapidly changing shot region using this method. Also, the event flow of the shot content is acquired. Fig. 2 shows the two different methods of the key frame selection.

4. Experiments & Results

A golf video in this figure is a competition game of Tiger Woods & David Duval. All experiments are done in golf videos. The performances are evaluated based on equation (10). A Golf video has various types of shot boundary, the large camera movements and large object motion. The scenes with camera tracking a golf ball which is hit by a golf club and then flying onto the sky and falling onto the green are usually mistaken as consisting of a few shots, which cannot be avoided in any shot detection systems without exact motion analysis and object tracking. After applying the proposed shot detection algorithm to three golf videos, we can get the results presented in Table 1. Two videos except "Woods & Duval II" have scenes with camera tracking a ball flying into the sky and falling onto the green. These scenes result in a few false alarms which decrease precision. Table 1 shows a high precision result on "Woods & Duval II" and low precision results on others.

$$\mathrm{Recall} = \frac{Correct}{Correct + Missed}$$
$$\mathrm{Precision} = \frac{Correct}{Correct + FalseAlarms} \qquad (10)$$

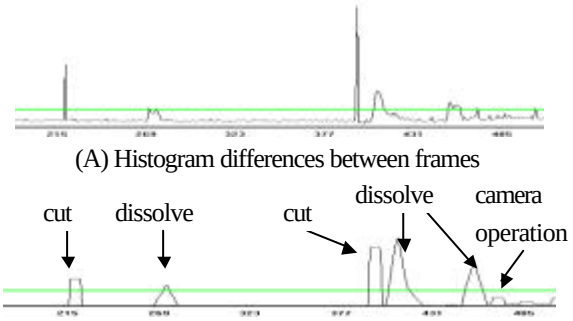
Fig. 3 represents the change of color histogram differences before and after the low pass filtering is applied. In Fig. 3(A) showing the histogram differences between frames according to equation (2), there are difficulties in discriminating between camera movement and dissolve frames. In the Fig. 3(B), on the contrary, the clear detection between shot boundaries and camera movements is possible after the low-pass filtering is applied to an original sequence of differences. The threshold found by Otsu method is presented in Fig. 3(B)

Fig. 4 shows the key frame selection results. The shot contains the 136 frames and the frame number in the shot is from 341 to 476. Fig. 4(A) shows the key frames selected by thresholding the color histogram differences. The key frames are selected within 14 frames. This method does not represent the whole shot contents. Fig. 4(B) shows the results of proposed

algorithm. The result shows that the selected the key frames represent the whole shot contents.

Table 1 Precision and Recall results on golf videos

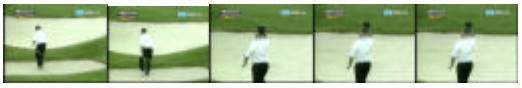
	PGA (4724 frames)		Woods & Duval I (1609 frames)		Woods & Duval II (no dissolve)	
	cut	dissolve	cut	dissolve	cut	dissolve
Correct	16	21	9	4	23	
Missed	1	1	1	1	1	
False Alarms	2	4	2	3	1	4
Recall	0.94	0.95	0.90	0.80	0.96	
Precision	0.89	0.84	0.82	0.57	0.96	



(A) Histogram differences between frames

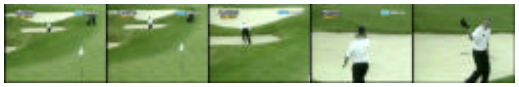
(B) Low-pass filtered histogram differences between frames

Fig. 3 Histogram differences between frames



427 428 436 437 440

(A) Using frame difference threshold method



358 394 420 437 460

(B) Using proposed algorithm

Fig. 4 Key frames and the frame number

The key frames selected by the proposed algorithm show the shot content more effectively than other conventional algorithms. Fig. 5(A) shows the color histogram differences and threshold value, and Fig. 5(B) shows the accumulated histogram differences and sampling ratio. In Fig. 5(A) the key frames are selected only in a part of the shot.

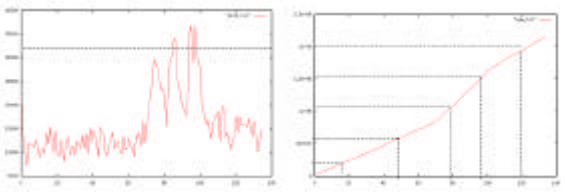


Fig. 5 Frame feature and the sum of frame feature.

5. Conclusions & Further Works

We have presented in this paper the shot detection algorithm and the key frames selection method. The proposed shot detection method utilizes low-pass filter to reduce false alarms caused by image motion such as camera and objects movements. Because this method uses only color histograms as feature data, the edit constancy effects are usually distorted in real images. New features resulting in edit constancy effects similar to ideal ones will be developed in the future. For the key frames selection method, we also proposed the adaptive temporal sampling algorithm. The experimental results show that this algorithm select key frames considering the temporal variation of the histogram differences automatically. For more meaningful key frame selection, more information of the frame is needed, such as motion, texture.

References

[1] H.J.Zhang, A.Kankanhalli, S. W. Smoliar, and S.Y. Tan, “Automatic partitioning of full motion video”, ACM Multimedia systems 1(1), 1993,10-28

[2] A. Mufit Fermann, A. Murat Tekalp, "Efficient filtering and clustering methods for temporal video segmentation and visual summarization," J. Vis. Comm. and Image Rep., vol. 9, no. 4 (special issue), pp. 336-351, Dec. 1998.

[3] A. Hampapur, R. Jain, T.E. Weymouth,

"Production model based digital video segmentation", Multimedia Tools and Applications, vol.1, pp9-46,1995

[4] Bilge Gunsel and A. Murat Tekalp, "Content-based video abstraction", Proc. IEEE Int. Conf. Image Proc., Chicago, IL, Oct. 1998.

[5] S. Moon-Ho Song, Tae-Hoon, Kwon, Woonkyung M.Kim, "On Detection of Gradual Scene changes for parsing of Video data", SPIE: Storage and Retrieval for Image and Video Databases 1998, pp404-423