Fast Scene Change Detection using Visual Spatiotemporal Pattern

Yeon-Seok Seong, Jung-Rim Kim, Myonghoon Kim,
Ja-Cheon Yoon, and Sanghoon Sull
Dept. of Electronics and Computer Engineering, Korea University
{ysseong, jrkim, mhkim, jcyoon, sull}@mpeg.korea.ac.kr

Abstract: In this paper, we propose an algorithm of fast scene change detection based on a visual spatiotemporal pattern, called visual rhythm, corresponding to sampled subset of the reduced image sequence obtained by partially decoding an MPEG-2 compressed video. The proposed method partially-decodes a subset of macroblocks, in each frame, necessary to generate the visual rhythm of a compressed video for fast scene change detection. The experimental results with real broadcast HDTV streams demonstrate the feasibility of our approach.

Keywords: scene change, visual rhythm

Ⅰ. Introduction

As the adoption of digital high-definition television (HDTV) broadcasting continues to gain popularity due to its higher quality video and audio, more HDTV streams becomes available to consumers HDTV. The formats of digital broadcast streams are 480p for standard definition television (SDTV), and 720p and 1080i for high definition television (HDTV). The latest digital TV set-top box (STB) equipped with hard disk, called as a personal video recorder (PVR), allows digital recording of broadcast TV streams. Thus it will be convenient if PVR users are provided with a variety of advanced features such as video browsing, video editing, content-based video retrieval, and skip-play, all of which can be efficiently implemented by first detecting scene changes.

A number of researchers have proposed scene change detection algorithms. Nagasaka et al. [1] uses various difference metrics such as difference of gray-level sum, sum of gray-level difference, and difference of gray-level histogram. This method is based on the processing of pixels in spatial domain, and thus computational complexity is very high for HD MPEG-2 compressed video.

In order to reduce the high complexity caused by IDCT, the approaches for scene change detection in MPEG compressed domain have been developed [2]. Yeo et al. [2] proposed a scene change detection method using reduced image sequence from MPEG-2 compressed video. They use pixel difference and histogram difference of reduced images of successive frames as a metric. However, this method is still based on the use of the whole pixels of each reduced frame and thus the speed of scene change detection could be improved if only part of each reduced frame is utilized without generating the whole pixels for each full reduced frame.

Another approach for fast scene change detection algorithms is to utilize a visual rhythm (VR) [3], also called temporal slice [4]. A visual rhythm image is typically obtained by sampling pixels lying along a sampling path, such as a diagonal line traversing each frame as shown in Fig. 1. A line image is produced for the frame, and the resulting line images are stacked, one next to the other, typically from left-to-right. In this manner, the visual rhythm image contains patterns or visual features that allow the viewer/operator to distinguish and classify many different types of video effects, (edits and otherwise), including: cuts, wipes, dissolves, fades, camera motions, object motions, flashlights, zooms, etc. The different video effects manifest themselves as different patterns on the visual rhythm image. The scene change detection based on a VR has low computational complexity due its use of a set of sampled pixels per frame and can also be used for visual shot verification [5]. These methods are based on the use of part of pixels in each frame, thus each full frame needs to be decoded which is computationally expensive.

In this paper, we introduce an efficient scene change detection that is suitable for HD streams by fast generating a VR from a set of sampled pixels for each reduced size frame obtained in MPEG-2 compressed domain. The computational savings for our proposed method is due to the use of part of each reduced frame without generating each full reduced frame.

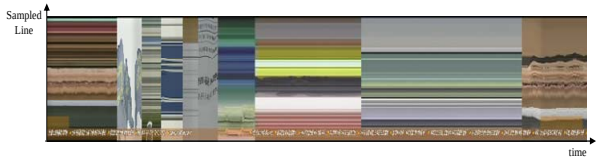


Fig. 1. An example of visual rhythm

II. Proposed Algorithm

1. Visual Rhythm of DC Images

For the fast scene change detection, we utilized VR generated from sampled subset of DC images directly obtained in compressed domain from a compressed video encoded such as in MPEG-2. The DC images of a video correspond to block-wise averages of 8 x 8 spatial blocks in the corresponding frame. Let $f_{DCImage}(x, y, t)$ be the representation of a DC image at time t, then the spatially reduced video or sequence of DC images may be expressed as:

$$f_{DCImage}(x,y,t) = \frac{1}{64}\sum_{k_x=0}^{7}\sum_{k_y=0}^{7} f_V(8x+k_x, 8y+k_y, t), \quad (1)$$
$$\text{for } x,y,t \in \{0,1,2,\ldots\}.$$

Using the DC images of a video, we define the visual rhythm VR of the video as follows:

$$VR = \{f_{VR}(z,t)\} = \{f_{DCImage}(x(z), y(z), t)\}, \quad (2)$$

where x(z) and y(z) are one-dimensional functions of the independent variable z.

Fig. 2 shows an example of the sampling strategies (horizontal, vertical, diagonal) for the construction of VR. The characteristic of a VR is sensitive to a direction of sampling. A horizontal VR discloses principally a horizontal motion of an object in the video, and oppositely a vertical VR discloses a vertical motion. A diagonal VR discloses both motions in horizontal and vertical directions and is thus effective to represent the characteristics of a video. However, a diagonal VR needs more sampled pixel values than a horizontal or vertical VR.

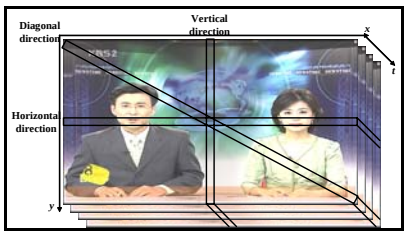


Fig. 2. Three sampling strategies of visual rhythm of DC images

2. DC Image Reconstruction in Compressed Domain

We use DC images to fast generate the VR for scene change detection. A DC image is obtained by reducing the height and width of a compressed video by a factor of eight as described in [6]. Since the height and width of HDTV streams are usually large, the DC images are usually sufficient to represent visual features of an original video.

The method proposed by Song *et al.* [6] is used to generate DC images and it is summarized as follows. In Fig. 3, P_{ref} is the block to be referenced by the current block, P_0, P_1 ,P_2 , and P_3 are the neighboring blocks of P_{ref} , and $(\Delta x, \Delta y)$ is the motion vector. To construct an accurate DC image of the noninterlaced and frame based motion compensated block, the DC coefficients of P_{ref} is expressed as follows:

$$(DCT(P_{\text{ref}}))_{00} = \frac{1}{64}\sum_{i=0}^{3}\left\{\sum_{m=0}^{7}\sum_{l=0}^{7} w_i h_i (DCT(P_i))_{ml}\right\}, \quad (3)$$

where DCT is the discrete cosine transform (DCT) of block Pi.

The MPEG-2 video standard also supports the mixture of frame and field-based motion compensation. Thus, the DCT domain deinterlacing and inverse motion compensation are needed to handle the mixture of different types of motion compensation, such as field based motion compensation and frame based motion compensation for an accurate DC image. Song et al. [6] also proposed approximate reconstruction of DC image by fast DCT domain deinterlacing and inverse motion compensation operation using one DC and two AC coefficients.

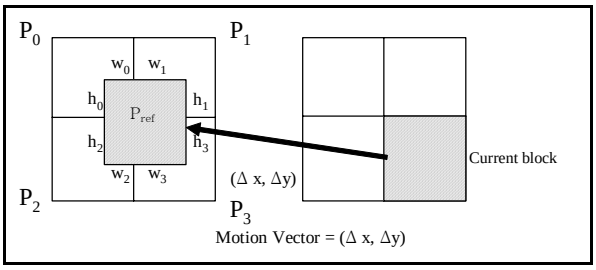


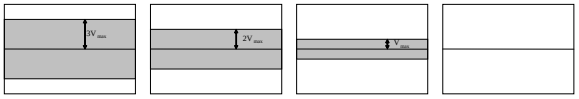
Fig. 3. Prediction of an 8 x 8 block

3. Scene change Detection Using Visual Rhythm

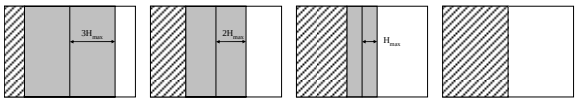
HDTV broadcasting stream is transmitted in MPEG-2 transport stream (TS). A TS is formed by multiplexing video, audio and auxiliary data streams where a video stream consists of GOP (Group of Pictures) containing I-, P-, and B-frames. A P-frame is encoded relative to the past reference frame and a B-frame is encoded relative to the past reference frame, the future reference frame, or both frames. The upper limit of motion vectors, also called the maximum motion

vector in this paper, is determined during encoding process.

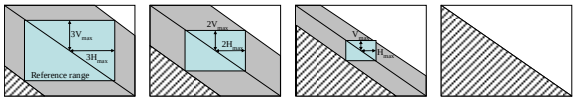
Fig. 4 illustrates the macroblocks of the pixels to be partially decoded for three sampling strategies for generating VRs from a MPEG-2 compressed video consisting of a GOP with one I-frame and three P-frames. Note that the partial decoding mentioned above involves variable-length decoding, extraction of one DC and two AC coefficients for each block in a macroblock to generate a pixel of a DC image. The I0-frame is referenced by P_1-frame, P_1-frame is referenced by P_2-frame, and P_2-frame is referenced by P_3-frame. P_3-frame is assumed not to be used for reference and thus just a set of macroblocks along the VR sampling line needs to be partially-decoded. The shaded areas in P_2-, P_1-, and I_0-frames describe the macroblocks, extended by as much as the appropriate sizes of H_{max} and V_{max}, which can be used for motion compensation. Thus, if the length of a GOP is shorter, the number of marcoblocks to be partially decoded becomes smaller. In this way, a VR can be efficiently constructed.



(a) Horizontal direction



(b) Vertical direction



(c) Diagonal direction

Fig. 4. Examples of partially decoded macroblocks: The H_{max} and V_{max} represent the maximum horizontal and vertical motion vectors, respectively. The left and left-lower parts, shown in slanted lines in (b) and (c), respectively, are not partially-decoded, but just variable-length decoded.

Scene change can be detected based on the discontinuity in VR. The discontinuity of VR in between time t-1 and t can be expressed as follows:

$$d_s(t) = \sum \left| f_{VR}(z,t) - f_{VR}(z,t-1) \right| . \qquad (4)$$

If $d_S(T)$ is larger than a threshold value, then a scene change is detected.

III. Experimental Results

We implemented our method for the PC system with 2.6GHz CPU, 1GB memory and MS Windows 2000, and tested it for three ATSC compliant HDTV streams with the resolution of 1920 and 1080 pixels with interlaced 15-frame GOP containing one I-frame and four P-frames. The maximum horizontal and vertical motion vectors turned out to be 201.5 and 123, respectively. It is noted that for simplicity 10 B-frames for a GOP are not considered for scene change detection in our current implementation. A scene change method based on pixel difference of full DC images generating by using Song's method [10] was implemented for the comparison with our proposed method. Table 1 shows the resulting scene change detection times for three HD streams by using Song's method that processes all macroblocks in I- and P-Frames, and by using horizontal, vertical, and diagonal VRs. Our proposed method reduces the scene change detection times by 50% (horizontal VR), 35% (vertical VR), and 25% (diagonal VR) than that of using full DC images. Table 2 shows the numbers of reference macroblocks to be partially decoded for the purpose of the generation of the horizontal, vertical, and diagonal VRs, respectively.

Table 1 Comparison of Scene Change Detection Time.

	HD stream #1	HD stream #2	HD stream #3
Total duration	74 sec	51 sec	86 sec
Use of full DC images	36.4 sec	24.5 sec	44.1 sec
Use of Horizontal VR	18.2 sec	12.0 sec	22.3 sec
Use of Vertical VR	23.9 sec	16.2 sec	30.3 sec
Use of Diagonal VR	27.6 sec	18.4 sec	34.2 sec

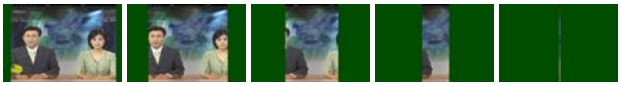
Table 2 Ratio of the number of skipped macroblocks for each VR sampling strategy to the number of whole macroblocks for full DC image.

	I_0	P_1	P_2	P_3	P_4
Horizontal VR	4%	28%	51%	75%	99%
Vertical VR	6%	17%	28%	39%	50%
Diagonal VR	1%	6%	16%	33%	50%

Table 2 shows the ratio of the number of skipped macroblocks for horizontal VR, (vertical VR, and diagonal VR) to the number of whole macroblocks for full DC image. It is observed that the number of skipped macroblocks for the horizontal VR is larger than that for the vertical VR, which can be also expected from Table 1 showing that the scene detection time of the horizontal VR is less than that of the vertical VR. The generation of a horizontal VR is faster than that of vertical and

diagonal VR since the unnecessary macroblocks of the horizontal VR can be skipped by using MPEG-2 slice start code while the unnecessary macroblocks for the vertical and diagonal VRs cannot be skipped (i.e. variable length decoded). Note that in case of frame I0, most of macroblocks need to be partially decoded.

(a) Horizontal



(b)Vertical

(c) Diagonal

Fig. 5. The resulting frames showing the partially decoded macroblocks for VR. The green area represents the macroblocks not partially decoded. Note that the left and left-lower parts in (b) and (c) are not reconstructed but variable length decoded.

Fig. 5 illustrates the partially decoded macroblocks to generate the horizontal, vertical, and diagonal VRs, respectively. The first images of each row are nearly fully reconstructed and the second, third, fourth images of each row are partially reconstructed within the range of the maximum motion vector. In case of the last images of each row, only the macroblocks belonging to a sampling line of VR are reconstructed since last P-frame of a GOP are not used for reference frames.

Precision and recall are also used in this paper for performance evaluation as follow:

$$precision = \frac{n_c}{n_c + n_f},\ recall = \frac{n_c}{n_c + n_m}, \qquad (5)$$

where n_c is the number of correct scene change detection, n_f is the number of false detection, and nm is the number of missed.

The average precision and recall rate of our proposed method are about 81% and 93%, respectively. This demonstrates that our proposed method is faster than existing methods at the small expense of accuracy.

Ⅳ. Conclusion

In this paper, we have proposed an efficient scene change detection algorithm using visual rhythm corresponding to sampled subset of the reduced image sequence obtained by partially decoding an MPEG-2 compressed video. Our method reconstructs macroblocks of reference frame within the upper limit of motion vectors for the fast generation of visual rhythm

The experiments show that our algorithm is 1.3 to 2 times faster than traditional methods reconstructing all macroblocks.

References

[1] A. Nakasaka, Y. Tanka, "Automatic video indexing and full motion search for object appearance," Proc. *IFIP on Visual Database System* Vol. 2 pp. 113-127, 1992.

[2] B. Yeo, B. Liu, "Rapid scene analysis on compressed video," *IEEE Transaction on Circuit and System for Video Technology*. Vol. 5 pp. 533-544, 1995.

[3] M. G. Chung, H. Kim, S. M. Song, "Scene boundary detection method," Proc. *International Conference on Image Processing*, Vol. 3 pp. 993-936, 2000.

[4] C. W. Ngo, T.C. Pong, R. T. Chin, "Detection of gradual transitions through temporal slice analysis," Proc. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1 pp. 36-41, 1999.

[5] H. Kim, J. Lee, S. M. Song, "An efficient graphical shot verifier incorporating visual rhythm," Proc. *IEEE International Conference on Multimedia Computing and Systems*, Vol. 1 pp. 827-834, 1999. uly, 1990.

[6] J. Song, B. Yeo, "Fast extraction of spatially reduced image sequences from MPEG-2 compressed video," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 9 pp. 1100-1114, 1999.