An Evaluation of Video Cut Detection Techniques

Sandberg Marcel Santos[1], Díbio Leandro Borges[2], and Herman Martins Gomes[1]

[1] Departamento de Sistemas e Computação, Universidade Federal de Campina Grande, Av. Aprígio Veloso s/n, 58109-970 Campina Grande PB, Brazil
{sandberg,hmg}@dsc.ufcg.edu.br
[2] Departamento de Ciência da Computação, Fundação Universidade de Brasília, Campus Universitário Darcy Ribeiro, 70910-900 Brasília DF, Brazil
dibio@unb.br

Abstract. Accurate detection of shot transitions plays an important role on automatic analysis of digital video contents, and it is a key issue for video indexing and summarization, amongst other tasks. This work presents in more detail a novel strategy, based on the concept of visual rhythm, to automatically detect sharp transitions or cuts in arbitrary videos. The central part of the work is a comparative evaluation of this strategy versus three other very competitive approaches for video cut detection: one based on the visual rhythm concept, other based on pixel differentiation and a last one based on color histograms. The evaluation carried out demonstrated that the proposed method achieves, on average, higher recall rates at a cost of a slightly lower precision.

Keywords: video cut detection, visual rhythm, pixel differentiation, color histograms, video summarization.

1 Introduction

Digital video applications, such as digital libraries, interactive TV, and multimedia information systems in general, are growing fast due to the advances in multimedia encoding and decoding technologies, increase in computing power and the ever-expanding internet [1]. This has stimulated research in the areas video indexing, retrieval and summarization. While digital videos can be seen as formed by a concatenation of 2-D image samples (frames) of a scene, shots can be seen as a basic functional unit of a video. Shots are defined as uninterrupted sequences of video frames with graphic, spatial and temporal configurations [3]. The automatic detection of shots or the transition between two consecutive shots is an essential part of most video content analysis algorithms.

Gradual and sharp transitions are the two most known types of video transitions [4]. In this paper, we focus on the problem of detecting sharp transitions (or cuts), which is usually taken as a simpler problem than that of gradual transition detection. However the state of the art, as indicated in our literature review, reveals there is still room for improvements in the accuracy of cut detection techniques. In a previous work [13], we proposed an algorithm for video cut detection based on the concept of visual rhythm and compared this algorithm with a previous approach, based on the same principle (the work by Lu et al. [10]). The visual rhythm concept [4] is

L. Rueda, D. Mery, and J. Kittler (Eds.): CIARP 2007, LNCS 4756, pp. 311–320, 2007.

explained in more detail in Section 2. In the present paper, the focus was to extend the previous evaluation by incorporating two other widely used and well established techniques based on pixel differentiation (inspired on a measure of motion saliency as given in Wildes [16]), and one based on color histogram (combining the strategies presented by Lienhart [9], and Yeo and Liu [17]).

Next section presents a bibliographic review of related work on video transition detection, including all the competing approaches evaluated in this paper. Section 3 presents technical details of the principal visual rhythm approach. The comparison has been made using a set of arbitrary videos, collected from a public video database [12], which is described in more detail in Section 4. The experimental evaluation and results are also in Section 4. Final considerations and comments on future works are presented in Section 5.

2 Bibliographic Review

There is a variety of methods and techniques proposed to perform the automatic detection of video transitions. Gargi et al. [3], Lienhart [8,9] and Hanjalic [5] developed comparative studies for some of the most representative approaches.

Methods that do not use computed features of compressed videos (e.g. the motion vectors of an *mpeg* video) rely on the assumption that frames from the same shots present a certain visual consistency, whereas frames in the vicinity of video transitions present important variation. Color histograms, pixel differences, edge variation and motion are very popular amongst the kinds of features that have been utilized in the characterization of such variation.

Yeo and Liu [17] and Zhang et al. [19] proposed methods for cut detection designed to distinguish between sharp transitions and sharp illumination variations. Yeo and Liu [17] detected peaks generated by each sharp illumination variation (one at the beginning and another at the end of the variation). Zhang et al. [19], on the other hand, considered models of ideal cut and flashlight detection. Other methods that are based on edge detection, usually demand high computation resources and are sensitive to fast object and camera motion. In order to address these questions, Jun and colleagues [6], proposed to apply a median filter to the features extracted from the video and to compare it to the original signal. A similar work has been done by Leszczuk et al. [7], who implemented a differential motion factor. Other works [15,18] employed a local adaptive threshold, instead of a fixed threshold, to classify an inter-frame variation as sharp or not.

In order to reduce the sensitivity of fast object and camera motion Zheng et al. [20] performed feature extraction (color histogram, pixel differences, standard deviation, mean deviation and motion vectors), in either compressed or uncompressed domains. From the analysis of the variation of these features, decisions (such as the choice for global or local thresholds) are taken and the transition detection is performed.

A popular method for cut detection is based on the concept of visual rhythm [4,10, 11,13]. The visual rhythm is a simplification of a video into a 2-D image [4]. A video sequence is typically seen as having three dimensions: one temporal (corresponding to the frame sequence) and two spatial (corresponding to the XY dimensions of each frame). The visual rhythm approach samples each video frame in such a way that it is

represented by a single 1-D line of pixels. These 1-D lines are, in turn, concatenated to form a 2-D image. Thus, a simplified video representation is obtained with only two dimensions: one temporal (typically the horizontal direction) and another spatial (typically the vertical direction). The 1-D lines that form this new 2-D representation can either be the main diagonal, the central row or the central column of pixels of a video frame. Chung et al. [2] considered the main diagonals as the most interesting lines to use because they contain information from both the lines and the columns of the frame. However, Lu et al. [10] opted to use the horizontal central line, because, according to them, when recording a video, the camera normally moves in the horizontal plane and the camera operator usually locates the interesting objects in the center of the field of view.

After creating the visual rhythm signal, sharp transitions or cuts in a video can be detected using bi-dimensional image processing algorithms. Firstly, different patterns in the signal are identified, and then an association between each video transition event and a pattern in the visual rhythm signal is made. The video cut detection task is therefore to look in the signal for a pattern corresponding to a sharp transition (usually a distinctive vertical line that separates two homogeneous patterns, one to the left and another one to the right of the visual rhythm signal).

One drawback of this approach is that the association between the sharp transition and the line that separates two video shot patterns is not one-to-one. Every sharp transition in the video creates a distinctive vertical line separating two patterns in the visual rhythm; however the contrary is not necessarily true, since this line can represent different video events. An attempt to minimize this problem, presented by Guimarães et al. [4], was to perform a search for vertical lines that separate patterns in different visual rhythms of a same video (obtained from different video samplings).

Other authors consider the problem of cut detection as a problem already solved. However, the results of current approaches are not yet close to perfect detection, so there is still room for improvement. Moreover, due to difficulties in annotating video datasets, most methods are evaluated using only a small number of videos. Existing methods would possibly have reduced performance for larger datasets [19].

3 Main Approach

This section details the main cut detection strategy evaluated in this work. A preliminary evaluation and additional bibliographic references related to this strategy can be found in an earlier paper [13] by the same authors. The strategy is based on the concept of visual rhythm as described in Section 2 and differs from previous works with regards to the computation of the visual rhythm signal and the cut detection rule.

3.1 Description

The approach starts with the following: for each pair of adjacent columns of a generated 2-D visual rhythm image, we compute the integral of the absolute differences of pixel intensities in all lines, according to Equation 1.

$$D_j = \sum_{i=1}^{N} \left| f(i, j) - f(i, j+1) \right| \tag{1}$$

where i and j are line and column indices; D_j is the integral of the absolute differences for columns j and (j+1); N is the total number of lines; $f(i, j)$ is the pixel intensity at line i and column j, and $f(i, j+1)$ is the pixel intensity at line i and column j+1.

The values of D_j for each pair of adjacent columns create a 1-D signal that indicates the abrupt changes between the columns of the image (i.e. the sharp transitions between the frames of the video). As an example, Fig. 1(a) presents a sample of one of the videos used in the experiments that are described in Section 4. Figure 1(b) presents a 2-D image (concatenation of diagonal stripes) derived from the video in Fig. 1(a). Finally, Fig. 2 presents the 1-D signal created from the Fig. 1(b).

(a) (b)

Fig. 1. (a) Sample of one of the videos used in the experiments; (b) 2-D image generated from the video using a concatenation of diagonal frame stripes

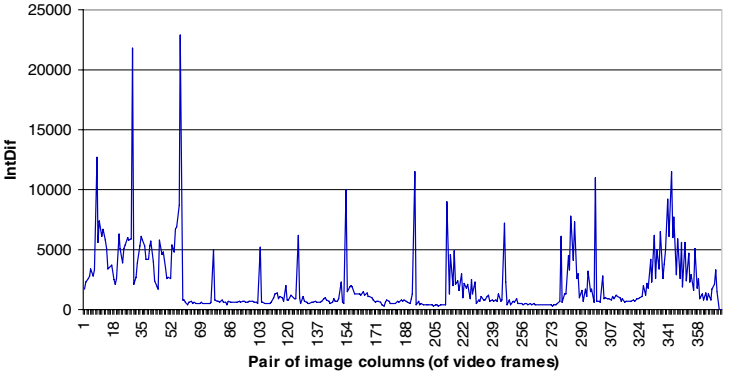

Fig. 2. 1-D signal generated from the visual rhythm image in Fig. 1(b)

The principle of the proposed method is that sharp peaks in the 1-D signal indicate the presence of sharp transitions in the video. Thus, the next step consists in automatically detecting these peaks and labeling the correspondent frames as cuts. Equation 2 formalizes this principle: a point j in the 1-D signal (D_j) is labeled as a sharp peak (indicating a cut or sharp transition) if it is greater than or equal the product between a pre-defined factor (k) and the average of the points inside a window (of size w) centered on the point (frame) j under analysis.

$$\textbf{if } D_j \geq k \times A \textbf{ then } D_j \text{ is a cut,} \tag{2}$$

where $A = \{(j - hw) + [(j - hw) + 1] + \ldots + j + \ldots + [(j + hw) - 1] + (j + hw)\} / w$; and $hw = \lfloor w/2 \rfloor$.

3.2 Finding Best Parameters

The procedure adopted to define the values for window size (w) and factor (k) is described as follows. It is based on an exhaustive search strategy, with the evaluation of possible combinations of window sizes and factors given a sampling interval. The window size w varied from 3 to 15, in steps of 2 (this way, we guaranteed that it is always an odd number, i.e. the window will always have a central point), and the factor k was varied from 1.1 to 3.0, in steps of 0.1. For each window-factor combination and each of the five videos (parameter fitting set), we calculate the precision $P^v_{w,k}$ and recall $R^v_{w,k}$ metrics for the parameter fitting set, where v is a video from the parameter fitting set; v = 1, 2, ... 5; w = 3, 5, ... 15; and k = 1.1, 1.2, ... 3.0. Then, we compute the arithmetic mean between the precision and the recall, which we named the correctness rate (C), presented in Equation 3.

$$C^v_{w,k} = \frac{\left(P^v_{w,k} + R^v_{w,k}\right)}{2} . \tag{3}$$

The window-factor combination which maximized, on average, the correctness rates for the five videos of the parameter fitting set was used to run the proposed algorithm on a new set of five videos (testing set). Precision $P^t_{w,k}$, recall $R^t_{w,k}$ and correctness $C^t_{w,k}$ rates were again computed for the testing set, where t is a video from testing set; t = 6, 7, ... 10; and (w, k) is the chosen best parameter pair.

Instead of finding the best window-factor combination from just a single pair of video sets (parameter fitting and testing), a cross-validation step was employed. The above procedure was repeated for all possible combinations of parameter fitting and testing sets taken from the complete set of 10 videos available. For each combination of parameter fitting and testing sets, we determined a window-factor pair which maximized the average correctness rates of the correspondent parameter fitting set. The several combinations of parameter fitting experiments were used to calculate the correctness rates of their correspondent testing sets. The averages of the several correctness rates obtained (for the different testing sets) were compared, and the window-factor pair associated with the testing set that presented the highest average correctness rate was defined as the ideal pair. The ideal window size (w) and factor (k) found as described above were 11 and 2.2, respectively.

The purpose of this cross-validation step was to find the best set of parameters to be used in experiments with new videos, taking into account all possible 5-element combinations of the available videos at this stage. A major drawback of this exhaustive search approach was that it was very time consuming. A computationally

more attractive alternative for this task would be to use a genetic algorithm optimization, but this was left as future work.

The videos used in the cross-validation described in this section were random and very diverse in nature, containing samples of indoor and outdoor sports, documentaries, TV series, and advertisements, among others. There was no special concern about selecting videos with static camera. The majority of the videos had camera movements, some especially intense. Video sizes were all of 320×240 pixels, and they have been captured in *mpeg*2 or *avi* formats, and resampled to 12 frames per second to reduce computational processing costs. The approximate length of each video was 31 seconds, resulting in about 375 frames per video. Frames where sharp transitions occur in the videos have been manually annotated.

4 Database, Evaluation and Results

The evaluation described in this section considered five new videos, not used for finding the ideal values of window size and the factor, as described in the previous section. The five new videos were taken from a public video base, the Open Video Project [12]. TRECVID project webpage [14] was also consulted, where annotated transitions are available for some videos of the Open Video Project database. In the TRECVID project, the videos of Open Video Project base are identified by *ids*. The videos selected for the evaluation performed in this paper are listed on Table 1. This table associates each *id* to its corresponding name in the Open Video Project [12].

Table 1. Videos utilized in the evaluation

id	Original Name (Open Video Project [12])	Duration	# of Frames (at 12 fps)
150	Three Smart Daughters (Singer Screen Ad)	1 min 21 s	970
160	Trip, The	48 s	604
169	Wonderful New World of Fords, A (1960 Ford Spot)	3 min	2184
269	Roads to Romance: The Santa Cruz Trail and Land of the Giant Cactus (in Arizona)	3 min	2163
272	She Caught on Quick (Singer Screen Ad)	1 min 15 s	957

These five videos have dimensions of 352×240 pixels, and are in *mpeg* format, sampled at 29.97 frames per second (fps). To reduce computation costs for the experimental evaluation, video frames were resampled to a smaller frame rate of 12 fps (using the Adobe Premiere Pro 1.5 software). This conversion incurred in a few frame drops, but considered unimportant for video cut detection. Besides computational time reduction, there was no other special reason for this chosen frame rate. For cut annotations, we started with the annotations available in TRECVID project webpage [14]. However, a manual revision of the original annotations was needed in order to correct a number of inconsistencies (this has been done in a similar way to the labeling performed for the ten videos used in the cross-validation, as commented in the previous section). Moreover, as the focus of this work is cut detection, gradual transitions (e.g. dissolves or wipes) were not considered.

Finally, cuts were annotated in the present work as frame numbers. This was done simply for convenience, since the raw information processed by our algorithm is at frame level. Time-based annotations can be easily derived from frame-based annotations and vice-versa. In the TRECVID project [14] webpage there is a rule that can help with these conversions. Table 2 presents the reviewed manual annotations for the five videos used in the evaluation. Each video is identified by its respective *id* presented in TRECVID webpage [14].

Table 2. Annotated cuts (indicated by the starting frame number) for the videos used in the experiment. Each cut is exactly 1 frame long

id 150	id 160	id 169	id 269	id 272
56	7	149	387	57
199	44	264	730	112
291	146	284	786	197
315	208	360	969	234
447	250	384	1006	356
686	284	421	1314	377
753	358	484	1442	401
800	422	732	1506	477
	487	925	1666	681
	596	1075	1695	736
		1283	1861	892
		1355	1927	
		1393		
		1552		
		1735		
		1804		
		1837		
		1870		

In Table 3, the results of the application of the main visual rhythm approach are presented on the five new videos obtained from the Open Video Project [12], using a window of size 11, and a factor of 2.2 (ideal window and factor values, determined through the cross-validation experiment explained in Section 3). The columns named "Cuts Detected", "Correct" and "Ground Truth" indicate, respectively, the total number of cuts detected, the number of cuts correctly detected by the approach, and the number of manually annotated cuts in the video. The column named "Precision" was calculated as the number of cuts correctly detected divided by the total number of cuts. The column named "Recall" was calculated as the ratio between the correct detections and the ground truth counts. The lines named "Mean" and "STD" indicate the mean value and standard deviations, respectively. Table 4 summarizes the results for Lu et al.'s visual rhythm approach [10] on the same set of videos. Both statistics in Tables 3 and 4 have been obtained in a previous work of the same authors [13].

In order to better characterize the above results and extend the initial evaluation with other important video cut detection techniques, a method based on pixel differentiation [16], and an approach based on the very popular concept of color histograms [9,17] have been implemented and tested. The goal was to evaluate and discuss precision and recall rates between the four approaches. All algorithms were

Table 3. Results for the main approach [13]

id	Cuts Detected	Correct	Ground Truth	Precision (%)	Recall (%)
150	13	8	8	61.54	100.00
160	13	9	10	69.23	90.00
169	23	17	18	73.91	94.44
269	21	12	12	57.14	100.00
272	18	11	11	61.11	100.00
Mean	17.60	11.40	11.80	64.59	96.89
STD	4.08	3.14	3.37	6.09	4.06

Table 4. Results for the approach by Lu et al. [10]

id	Cuts Detected	Correct	Ground Truth	Precision (%)	Recall (%)
150	1	1	8	100.00	12.50
160	2	1	10	50.00	10.00
169	2	1	18	50.00	5.56
269	2	1	12	50.00	8.33
272	3	3	11	100.00	27.27
Mean	2.00	1.40	11.80	70.00	12.73
STD	0.63	0.80	3.37	24.49	7.61

Table 5. Results for the pixel differentiation approach [16]

id	Cuts Detected	Correct	Ground Truth	Precision (%)	Recall (%)
150	7	3	8	42.86	37.50
160	5	5	10	100.00	50.00
169	12	8	18	66.67	44.44
269	10	6	12	60.00	50.00
272	13	9	11	69.23	81.82
Mean	9.40	6.20	11.80	67.75	52.75
STD	3.01	2.14	3.37	18.56	15.24

Table 6. Resulting statistics for the color histogram approach [9,17]

Id	Cuts Detected	Correct	Ground Truth	Precision (%)	Recall (%)
150	11	5	8	45.45	62.50
160	11	9	10	81.82	90.00
169	12	10	18	83.33	55.56
269	12	4	12	33.33	33.33
272	12	9	11	75.00	81.82
Mean	11.60	7.40	11.80	63.79	64.64
STD	0.49	2.42	3.37	20.48	20.03

applied to the same set of videos. Tables 5 and 6 present the results for pixel differentiation and color histogram approaches, respectively.

Tables 3, 4, 5 and 6 indicate that the novel visual rhythm approach presents recall rates higher than the ones presented by any of the compared approaches. On the other hand, the average precision rates presented by Lu et al.'s algorithm and by the pixel differentiation approach are a little higher. The standard deviations presented by the novel visual rhythm approach are lower for precision and recall when compared to

Lu et al.'s approach, and much lower when compared to the other two. This indicates better regularity in cut detection results, mainly regarding precision. A last consideration to be made upon the two approaches based on the visual rhythm concept is that Lu et al.'s algorithm [10], in general, is more restrictive about classifying frames as cuts. Thus, Lu et al.'s algorithm performs fewer cut detections, which can greatly favor precision, though always in detriment of recall. The novel approach revisited in this paper, in turn, detects more cuts, but in an efficient way, yielding higher recall rates and a small reduction in its average precision, relatively to Lu et al.'s algorithm.

5 Final Considerations

Accurate detection of sharp transitions is foremost important to automatic analysis of digital video contents. This work evaluated some of the most important techniques for video cut detection and presented in more detail a novel strategy based on the concept of visual rhythm to automatically detect sharp transitions or cuts in arbitrary videos.

The novel strategy based on visual rhythm is a simple, yet computationally attractive and of promising performance approach. When compared to three other competing techniques in the literature (based on visual rhythm (by Lu et al. [10]), one based on pixel differentiation (inspired on the work by Wildes [16]) and one based on color histogram [9, 17]), it presented very high recall and average precision rates. However, Lu et al.'s and pixel differentiation approaches performed a little better regarding average precision.

Special care has been taken to perform the evaluation utilizing videos publicly available. Moreover, video cut annotations have been clearly presented in Section 4, so that other groups can build on or confirm the obtained experimental results.

As future work, we intend to improve on this aspect by adding new constraints to the decision we make based on the integral of the absolute differences of pixel intensities for each line and each pair of adjacent columns of the 2-D signal that represents the visual rhythm of the video. Since the cross-validation process to find optimized parameters is very computationally intensive, an optimization process, possibly using genetic algorithms, will be an interesting approach to be investigated.

Another future work is to extend the novel strategy, with the accumulation of other evidences, aiming at: (i) gradual transition (as wipes and dissolves) detection; (ii) the detection of shots within shots (there are situations when a shot is interrupted by other shot and, after some time, the original shot continues from the point where it had stopped; in situations like this, the different pieces of the shot should be considered as only one shot; however, the proposed approach considers each piece as a distinct shot); and (iii) the development of techniques to perform video characterization/indexing/summarization based on the detected shots.

References

1. Bordwell, D., Thompson, K.: Film art: an introduction. Random House, New York (1986)
2. Chung, M.G., Lee, J., Kim, H., Song, S.M.-H., Kim, W.M.: Automatic video segmentation based on spatio-temporal features. Korea Telecom Journal 4(1), 4–14 (1999)

3. Gargi, U., Kasturi, R., Strayer, S.H.: Performance characterization of video-shot-change detection methods. IEEE Trans. on Circuits and Systems for Video Tech. 10, 1–13 (2000)
4. Guimarães, S.J.F., Couprie, M.: Video segmentation based on 2d image analysis. Pattern Recognition Letters 24(7), 947–957 (2003)
5. Hanjalic, A.: Shot boundary detection: unraveled and resolved? IEEE Trans. on Circuits and Systems for VideoTechnology 12(2), 90–105 (2002)
6. Jun, S.-C., Park, S.-H.: An automatic cut detection algorithm using median filter and neural network. Computers and Communications 2, 1049–1052 (2000)
7. Leszczuk, M., Papir, Z.: Accuracy vs. speed trade-off in detecting of shots in video content for abstracting digital video libraries. In: Boavida, F., Monteiro, E., Orvalho, J. (eds.) IDMS 2002 and PROMS 2002. LNCS, vol. 2515, pp. 176–189. Springer, Heidelberg (2002)
8. Lienhart, R.: Reliable transition detection in videos: a survey and practitioner's guide. Int. Journal of Image and Graphics 1(3), 469–486 (2001)
9. Lienhart, R.: Comparison of automatic shot boundary detection algorithms. In: Proc. SPIE Image and Video Processing, pp. 290–301 (1999)
10. Lu, S., King, I., Lyu, M.R.: A novel video summarization framework for document preparation and archival applications, In: IEEE Aerospace Conf. CDROM: IEEEAC paper #1415, pp. 1–10 (2005)
11. Ngo, C.-W., Pong, T.-C., Chin, R.T.: Video partitioning by temporal slice coherency. IEEE Trans. on Circuits and Systems for Video Technology 11(8), 941–953 (2001)
12. Open Video Project, http://www.open-video.org
13. Santos, S.M., Gomes, H.M., Borges, D.L.: A Novel cut detection strategy based on visual rhythm. In: IASTED Int. Conf. on Computational Intelligence, pp. 303—308 (2006)
14. TRECVID – TREC Video Retrieval Evaluation, http://www-nlpir.nist.gov/projects/trecvid/
15. Truong, B.T., Dorai, C., Venkatesh, S.: New enhancements to cut, fade, and dissolve detection processes in video segmentation. In: ACM Int. Conf. on Multimedia, pp. 219–227 (2000)
16. Wildes, R.P.: A measure of motion salience for surveillance applications. In: IEEE Int. Conf. on Image Processing, pp. 183–187 (1998)
17. Yeo, B.-L., Liu, B.: Rapid scene analysis on compressed video. IEEE Trans. on Circuit and Systems for Video Technology 5(6), 533–544 (1995)
18. Yusoff, Y., Christmas, W., Kittler, J.: Video shot cut detection using adaptive thresholding. In: British Machine Video Conf. pp. 362–371 (2000)
19. Zhang, D., Qi, W., Zhang, H.-J.: A new shot boundary detection algorithm. In: IEEE Pacific Rim Conf. on Multimedia, pp. 63–70 (2001)
20. Zheng, W., Yuan, J., Wang, H., Lin, F., Zhang, B.: A novel shot boundary detection framework. In: SPIE Visual Communications and Image Processing, pp. 410–420 (2005)