Improved Cut Detection for the Segmentation of Animation Movies

Bogdan Ionescu[1,2], Patrick Lambert[2], Didier Coquin[2], Vasile Buzuloiu[1]

[1] LAPI, University "Politehnica" Bucharest, 061071 Romania
bionescu,buzuloiu@alpha.imag.pub.ro,
[2] LISTIC, University de Savoie, B.P. 806, 74016 Annecy, France
patrick.lambert,didier.coquin@univ-savoie.fr

Abstract. In this paper a new cut detection algorithm, adapted to the segmentation of animation movies, is proposed. A cut is a direct concatenation of two different shots (fundamental video units) that produces a temporal visual discontinuity in the video stream. As color is a major feature of animation movies (each movie has its own particular color distribution) the proposed algorithm uses second order derivatives and Euclidean distances between color histograms of frames in order to detect the cuts. For frame classification, an automatic threshold estimation is proposed. Also, in order to reduce false detections, we propose an algorithm to detect an effect specific to animation movies, named "short color change" (i.e. thunders, lightening). The resulting method achieved better results compared to the classical histogram–based and motion–discontinuity based approaches, as shown by tests conducted on several animation movies.

1 Introduction

As multimedia content became increasingly accessible owing to the development of new video coding techniques and of devices with bigger storage capabilities, the interest on video databases indexation has drastically increased. Thanks to the "The International Animated Film Festival" [1], which takes place every year at Annecy, France since 1960, a very large database of animation movies is available. Managing thousands of videos is a tedious task; therefore an automatic content analysis would be more than welcome.

Detecting the video shots boundaries, that is, recovering the elementary video units, provides the ground for nearly all existing video abstraction and high–level video segmentation algorithms [2]. In order to assemble the movie, shots are linked together using video transitions. Video transitions are divided into sharp transitions, known as *cuts* (also the most frequently used), which result from the direct concatenation of two different shots without using any visual effects, and gradual transitions, such as *fades*, *dissolves*, *mattes*, *wipes* etc. that imply the use of special optical effects (for a literature survey see [3]).

In this paper, a new, improved, cut detection technique is proposed to cope with the issues raised by the peculiarity of animation movies. Animation movies are different from natural ones in that

- the events do not follow a natural way (objects or characters emerge and vanish without respecting any physical rules, movements are not continuous);
- the camera motion is very complex (usually 3D);
- the characters usually are not human and could have any shape;
- a lot of visual effects are used (color effects, special effects);
- every animation movie has its own particular color distribution;
- artistic concepts are used (i.e. painting concepts);
- various animation techniques are used (3D, cartoons, animated objects, etc.).

1.1 Previous Work

Existing cut detection algorithms differ in the features used to measure the visual discontinuity. Cut detection techniques can be classified as: intensity/color based, edge/contour based and motion based [2]. A comparison of all of the three approaches listed above is presented in [2][4]. In the class of intensity/color–based methods, the histogram–based methods are the most frequently used ones thanks to their invariance to some geometrical transformation in frames, and achieve better results comparing to the other methods [5]. Various approaches to histogram–based cut detection were proposed: color based histogram intersection metric using $C_b - C_r$ and $r - b$ spaces [6], histogram difference in the YUV space [7], YUV color histograms [8], sub–window based histograms to minimize the influence of local changes in illumination and motion [9], and a multi–level Hausdorff distance histogram [10].

1.2 The proposed method

The proposed cut detection algorithm exploits the color feature, as being the major feature of animation movies. The frames are color reduced and divided into four quadrants in order to reduce the influence of entering objects (as the predominant motion in animation movies is the object motion [15]). A study on the influence of objects size on the global color histogram is proposed. By using second order derivatives computed on a temporal mean of the Euclidean histogram distances, the influence of the repetitive camera motion is reduced and cuts better emphasized.

Also, false detections are reduced by detecting an effect specific to animation movies, which is called “short color change” or SCC (i.e. thunders, lightning, explosions) which is usually (wrongly) detected as a cut. The proposed algorithm uses a modified flash–detector. For frame classification, an automatic threshold estimation is proposed using statistical measures computed on the obtained distance vectors. The results obtained by the proposed method are better compared to the classical histogram–based and motion discontinuity–based cut detection approaches.

The remainder of the article is organized as following: Section 2 describes the proposed cut detection algorithm, in Section 3 we present some experimental results and Section 4 contains final considerations and proposes future improvements.

2 Cut detection

2.1 Video subsampling and color reduction

In order to reduce computational complexity, the video sequence is first temporally subsampled: only one frame in n is retained for further processing. As the original frames resolution corresponds to the PAL video standard, a spatial subsampling is required as well: only one pixel for each block of 4×4 pixels is retained (see Section 3 for more details on this matter).

In order to compute color histograms, one has to reduce the number of colors, as the original frames are represented in true colors. Several color reduction techniques have been tested and a study on the influence of the color reduction on the cut detection in animation movies has been proposed in [14]. Here, we chose to achieve color reduction using the Floyd–Steinberg error diffusion algorithm run in the XYZ color space [11], by using a standard color palette (216 web safe color palette [12]). The advantages of using the proposed palette are presented in [18]).

2.2 Color histogram computation

As stated before, animation movies have some peculiarities that must be specifically addressed. One of the most important issues when dealing with the color histogram is the movement of objects in the scene, as large–sized moving objects may produce noticeable differences in the histograms of successive frames. (According to [15], the predominant type of motion in animation movies is the object motion.) In order to reduce the method's sensitivity to emerging (or vanishing) objects, frames are divided into 4 regions (see Figure 6). A study conducted on the influence of the objects' size on the global color histogram has shown that only the objects of the size of an image quadrant or higher will significantly change the global color histogram (for more details see Section 3).

For each retained frame f_t^c (where ^{c} denotes the subsampled version and t the time index) four color histograms $H_t^j = H^j(f_t^c, i)$ (with i indexing colors) are computed, each one corresponding to an image quadrant (indexed by j). Then four Euclidean distances $D_j(t)$ between each color histogram in the frame f_t^c and the corresponding one in the next temporal frame f_{t+1}^c are computed (see Figure 1).

$$D_j(t) = \left(\sum_{i=1}^{N_c} \left[H^j(f_{t+1}^c, i) - H^j(f_t^c, i) \right]^2 \right)^{1/2} \tag{1}$$

where $N_c = 216$ represents the number of colors of the chosen palette.

Eventually, an average histogram distance between consecutive frames is computed as the arithmetic mean of the four $D_j(t)$ differences to stand as the basis for the cut detection procedure we shall explain in the sequel:

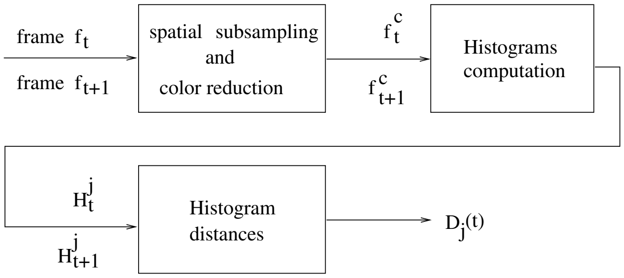


Fig. 1. Current frame analysis: f_t is the current analyzed frame, f_{t+1} is its next neighbor frame, f_t^c and f_{t+1}^c are the spatially subsampled and color reduced frames, H_t^j and H_{t+1}^j are their corresponding color histograms, $D_j(t)$ are the histogram distances for $j \in \{1, 2, 3, 4\}$.

$$D_{mean}(t) = \frac{1}{4} \sum_{j=1}^{4} D_j(t). \tag{2}$$

The advantage of using a single frame difference measure instead of four is twofold. It firstly leads to the simplification of the cut/non–cut decision. Secondly, thorough tests have shown that, provided that an accurate threshold selection procedure is devised, the method based on $D_{mean}(t)$ measure leads to improved results. More details on automatic threshold computation are presented in Section 2.4.

The use of the $D_{mean}(t)$ measure for cut detection is presented in the following paragraph.

2.3 Second order derivative

A cut means a strong color dissimilarity between two consecutive frames (i.e., a high value of the $D_{mean}(t)$ value) that continues with a strong color similarity between frames (i.e., a low value of $D_{mean}(t)$). This observation leads us to the observation that a simple temporal gradient computed on the $D_{mean}(t)$ sequence would lead to many false positives, as it does not take into account more that a pair of neighbors. This further leads us to the use of higher–order derivatives on the $D_{mean}(t)$ sequence for accurate cut detection.

Tests performed using n–order derivatives have shown that the higher the n value used, the lower the obtained cut detection rate. This can be explained by the decline of the distance values as n increases. The use the second order derivative, i.e., $n = 2$, has led to the best compromise between detection rate and false positive incidence. Moreover, the second order derivative is the best match for our problem, which consists in detecting the "low–high–low" pattern in the D_{mean} sequence (see Figure 2). Thus, the second order derivative of is computed as:

$$\ddot{D}_{mean}(t) = \frac{\partial^2 D_{mean}(t)}{\partial t^2} \tag{3}$$

where t is the frame index. Negative values are set to 0 as they contain redundant information. Cuts are then sought as local maxima of the $\ddot{D}_{mean}$ sequence.

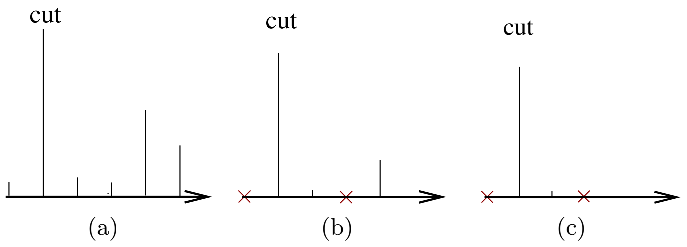


Fig. 2. Cuts emphasizing using second order derivatives: (a) D_{mean}, (b) $\dot{D}_{mean}$, (c) $\ddot{D}_{mean}$. The oX axis correspond to time, negative values (marked with red ×) are set to 0.

The advantage of using the second order derivative is illustrated in Figure 3 on a sample of an animation movie containing repetitive camera motion. The repetitive camera motion is the most frequent cause for false positives as it induces significant differences between histograms of consecutive frames. However, by using our second–derivative–based approach, the influence of repetitive camera motion is drastically reduced and real cuts are better emphasized (more details on this matter are presented in Section 3).

2.4 Automatic threshold estimation

For the final frame classification, color histogram mean distances are compared to a certain threshold s. Using a global pre–fixed threshold s is practically impossible, as each animation movie has its own color distribution [13]. Thus, an adaptive threshold computation is required. Several approaches to the estimation of the classification threshold are discussed in [2]. The most frequently used threshold determination is based on the average $\ddot{D}_{mean}$ value, computed as:

$$m_{\ddot{D}} = \frac{1}{N}\sum_{t=0}^{N} \ddot{D}_{mean}(t) \tag{4}$$

where N is the number of the retained frames. As cuts occurrence is reduced, taking the threshold as $m_{\ddot{D}}$ itself would lead to a high false detection ratio (the threshold is too low, see Figure 4). This is the reason why the threshold is set

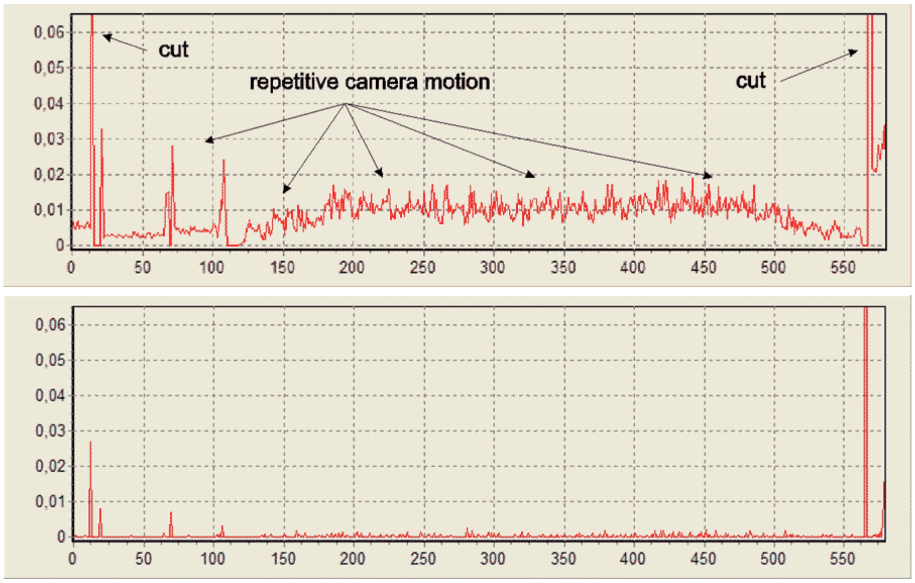


Fig. 3. Improvements to the cuts delimitation by using the second derivative: the $D_{mean}(t)$ sequence (top) and the $\ddot{D}_{mean}(t)$ sequence (down). One can remark the robustness of the method against camera movement.

to a higher value than $m_{\ddot{D}}$ that is computed by adding to the latter a fraction of the standard deviation of $\ddot{D}_{mean}$.

Our approach to threshold determination is different. The proposed threshold is determined by detecting all the significant local maxima of the $\ddot{D}_{mean}(t)$ sequence by identifying the following configuration:

$$\begin{aligned} &\ddot{D}_{mean}(t) > m_{\ddot{D}}, \\ \text{AND} \quad &\ddot{D}_{mean}(t-1) < \ddot{D}_{mean}(t), \\ \text{AND} \quad &\ddot{D}_{mean}(t+1) < \ddot{D}_{mean}(t) \end{aligned} \tag{5}$$

with $t = 1, \ldots N-1$

Then the threshold s, is set to the average value of the retained peaks of the $\ddot{D}_{mean}(t)$ sequence (see Figure 4). Experimental result have proven that this choice leads to a very good detection rate, as presented in Section 3.

Thus, the cut detection is performed by thresholding the $\ddot{D}_{mean}$ sequence with the proposed threshold s. More precisely, a cut is detected whenever:

$$\ddot{D}_{mean}(t) > s \quad \text{AND} \quad \ddot{D}_{mean}(t-1) < s \tag{6}$$

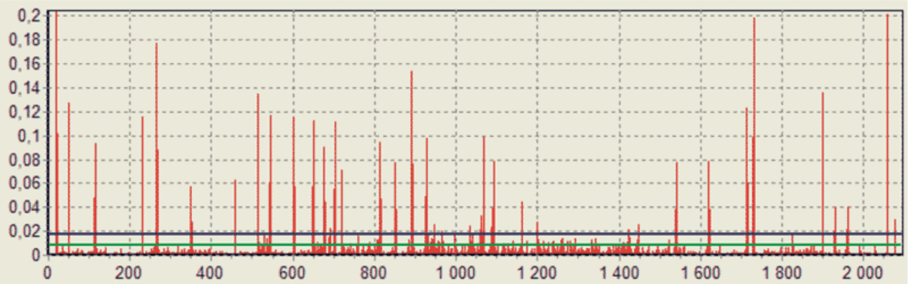


Fig. 4. Threshold estimation example: local maxima correspond to cuts, the proposed threshold s is depicted with the blue line, the green line correspond to s_{mean}.

2.5 False detection reduction

Because the animation movies contain a lot of special visual effects, the occurrence of false cuts are very likely. In order to reduce the false detection rate, every detected cut according to Eq. (6), is additionally checked in view of detecting a color effect specific to animation movies called "short color change" or SCC (see Figure 5), effect that is responsible for a part of the false positives.

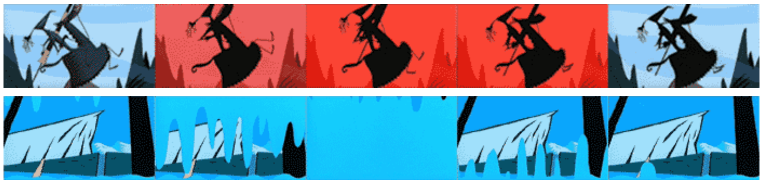


Fig. 5. Examples of "short color changes" (from "Francois le Vaillant" movie[1]).

The proposed algorithm is inspired from the flashlight detection in natural movies [16]. An SCC starts with an important change in color and ends with almost the same frame as the starting one. The dissimilarity between frames is transformed in Euclidean distances between global color histograms. Instead of using an accurate but slow color reduction algorithm (like the error diffusion), a fast uniform quantization of the RGB color cube into 125 colors is used.

Distances between histograms $H(f_t^c)$ and $H(f_{t+1+l}^c)$ (with f_t^c the retained frame at time index t, ^{c} denotes the subsampled version, t is fixed and $l = 1, 2, 3, \dots L_{\max}$) are successively computed. Here, $L_{\max}$ is the maximal admitted length of the SCC effect (a reasonable value is $L_{\max} = 10$). If, for a given l, the distance between the two histograms is lower than s (with s being the threshold used in cut detection, defined in Section 2.4) the detected cut is classified as SCC, and gets, thus, discarded.

3 Experimental results

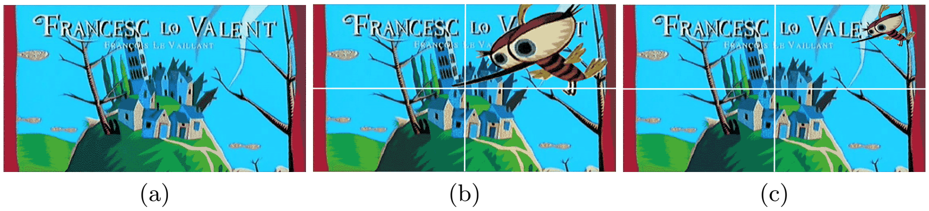
In this section, the choice of the used parameters is discussed, and experimental results are presented.

3.1 Choice of the method's parameters

As we already mentioned, in order to reduce computational complexity, the frames are both temporally and spatially subsampled. For the choice of the *temporal* subsampling step n, in order to keep the accuracy of the detection high enough, we used $n = 2$.

Several tests were performed for different values of n ($n \in \{1, .., 10\}$). Also, an adaptive subsampling proposed in [17] was tested. The adaptive subsampling uses a "divide et impera" based algorithm: the sequence is first subsampled using a high step value and if a cut occurred the step is divided progressively in order to localize the cut precisely. However, the procedure proved too time consuming, especially for movies with short shots (i.e., with many cuts). For this reason, it was eventually dropped and the uniform temporal subsampling was used.

In what concerns the splitting of the frame into subframes, in order to compute local histograms $H^j(f_t)$ (with f_t the current analysed frame, t the time index and j the image quadrant), a study on the influence of different object sizes on the global color histogram has been conducted. Results are presented in Figure 6 and Table 1. We finally chose to divide frames into four quadrants only because it has come up that only objects of the size of an image quadrant or higher change significantly the global color histogram, thus leading to false detections.



(a) (b) (c)

Fig. 6. Frames with different object sizes used to compute Table 1: (a) image$_{ref}$ represents the original frame, (b) image$_{1/4}$ is the frame with an object of 1/4 of the frame size, (c) image$_{1/16}$ is the frame with an object of 1/16 of frame size, (the image quadrants are delimited with the white line).

image used	image$_{ref}$	image$_{1/4}$	image$_{1/16}$
$d_E(H, H_{ref})$	0	**0.3**	0.04

Table 1. Euclidean distances between histograms for different sizes of emerging/vanishing objects: H_{ref} is image$_{ref}$ histogram, d_E is the Euclidean distance between current frame histogram and H_{ref}.

3.2 Detection results

The performance of the proposed methods are evaluated by using the precision/recall ratios, defined as:

$$\text{precision} = \frac{\text{GD}}{\text{GD} + \text{FD}}, \quad \text{recall} = \frac{\text{GD}}{\text{N}_{\text{total}}} \tag{7}$$

where GD is the number of good detections, FD is the number of false detections and N$_{\text{total}}$ is the total number of real transitions.

The "short color change" (SCC) detection was validated using 14 short–time animation movies from [1] (total amount of time of 101min47s and 120 SCC) and achieved an overall precision and recall of **93%** and **88.3%** respectively. The detailed report on GD and FD for each movie is presented in Table 2.

N$_{\text{total}}$	1	0	39	7	0	7	0	0	4	2	45	3	12	0
GD	1	0	38	6	0	5	0	0	4	1	40	2	9	0
FD	0	0	2	0	0	0	1	0	0	0	0	0	5	0

Table 2. SCC detection results for the 14 test sequences.

The proposed cut detection algorithm was tested on two long animation movies with a total time of 158min4s and 3166 cuts (movie1: 84min46s and 1597 cuts, movie2: 73min18s and 1569 cuts) and compared with two other cut detection methods. The first method proposed in [14] (referred to as *4histograms method*) uses directly the frames color histograms in order to detect the cuts while the second one (referred to as *motion method*) uses a motion discontinuity–based approach (the motion field is estimated by using a block–based approach).

In order to estimate the detection errors, cuts have been manually labeled using a specially–developed software. The obtained results are presented in Table 3 and Figure 7.

The obtained false detections are mostly owing to the very fast camera motion, while misdetections to color similarities between the cut's frames. Some examples of the most frequent error situations are illustrated in Figure 8.

The classical histogram–based method (*4histograms method*) leds to a lower recall, 88.63%, thus a lower recognition rate, while the motion–based method

Method	N_{total}	GD	FD	precision	recall
4histograms	3166	2806	199	**93.37%**	**88.63%**
motion	3166	2993	355	**89.39%**	**94.53%**
proposed	3166	2931	157	**94.92%**	**92.6%**
proposed (with SCC det.)	3166	2931	127	**95.97%**	**92.6%**

Table 3. The obtained cut detection errors.

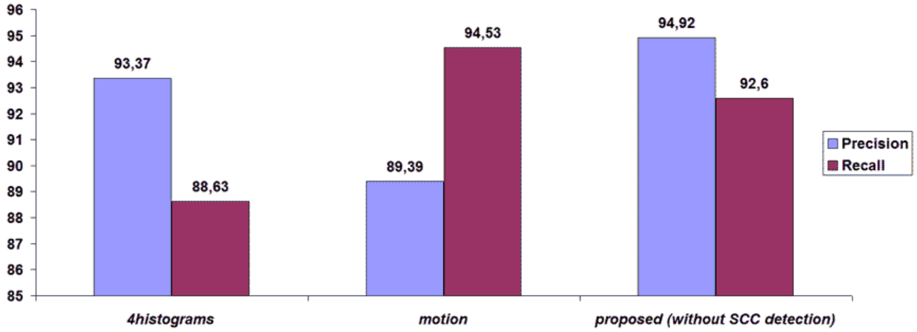
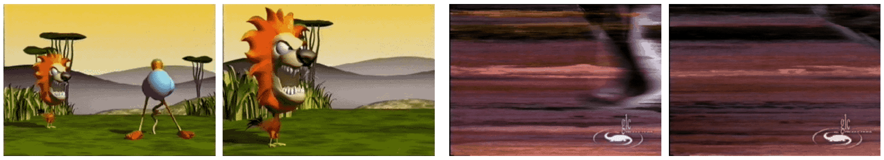


Fig. 7. The obtained cut detection precision and recall.

(*motion method*) achieved a better recognition but with a higher false detection ratio, precision 89.39%. Using the proposed method we have obtained both higher precision and higher recall ratios (above **92%**, see Figure 7). Also, using the proposed method in conjunction with the "short color change" detection procedure, the precision was improved of 1% (see Table 3).

The motion–based approach is less suited for animation movies as, for this type of movies, the motion is in general discontinuous and very fast, leading to a high false detection ratio.



(a) frames 2329,2328

(b) frames 4923,4924

Fig. 8. Cut misdetection/false detection examples: (a) color similarity (misdetection), "Gazoon" movie; (b) very fast motion (false detection), "The Buddy System" movie [1]).

4 Conclusions

In this paper a new improved cut detection technique adapted to the segmentation of animation movies is proposed. The visual discontinuities produced in the video stream by the cuts are converted into mean Euclidean distances between color histograms of successive frame regions. A number of improvements are proposed in order to manage the specificities of animation movies: color information is used (as each animation movie has its own particular color distribution), frames are divided into smaller regions, neighbor frames are analyzed, second order derivatives of mean distances are used, an special effect specific to animation movies is detected to decrease incidence of false positives. All of these procedures aim at reducing the influence of object/camera motion (which is predominant in animation movies) and of special visual effects. For the frame classification an automatic threshold estimation is proposed. Experimental results show a very good recognition ratio of the proposed method compared with the classical histogram–based and motion–based approaches. The obtained recall and precision are above 92% (that is, an improvement of around 3% with respect to classical techniques). Also, using the proposed method in conjonction with the "short color change" detection, the precision rate was improved of 1%.

The obtained false detections were related to the non–continuous very fast motion and color changes while the misdetections were related to color similarity of cut frames.

Future improvements consists on merging the detection results for the *proposed method* with the ones obtained with the *motion method*, all in order to reduce the misdetections thus to improve the recall ratio. A possible strategy is based on the use of a confidence measure in order to describe the quality of the detection for both the two methods.

Acknowledgments

We thank CICA and Folimage for the providing us with animation movies and for their support.

References

1. Centre International du Cinema d'Animation, "http://www.annecy.org".
2. R. Lienhart, "Reliable Transition Detection in Videos: A Survey and Practitioner's Guide" *International Journal of Image and Graphics*, Vol. 1(3), pp. 469-486, 2001.
3. A. D. Bimbo, *Visual Information Retrieval,* Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, pp. 202-259, 1999.
4. A. Dailianas, R. B. Allen, P. England, "Comparison of Automatic Video Segmentation Algorithms" *SPIE Integration Issues in Large Commercial Media Delivery Systems*, Vol. 2615, pp. 2-16, 1995.
5. U. Gargi, R. Kasturi, S. H. Strayer, "Performance Characterization and Comparison of Video-Shot-Change Detection Methods" *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 10(1), 2000.

6. M. S. Drew, Z. N. Li, X. Zhong, “Video Dissolve and Wipe Detection via Spatio-Temporal Images of Chromatic Histogram Differences” *Proceedings of IEEE International Conference on Image Processing*, Vol. 3, pp. 929-932, 2000.
7. S. Nayaga, S. Seki, R. Oka, “Temporal Video Segmentation using Unsupervised Clustering and Semantic Object Tracking” *SPIE Journal of Electronic Imaging*, Vol. 7(3), pp. 592-604, 1998.
8. S. H. Kim, R. H. Park, “Robust Video Indexing for Video Sequences with Complex Brightness Variations” *Proceedings of IASTED International Conference on Signal and Image Processing*, pp. 410-414, 2002.
9. A. Nagasaka, Y. Tanaka, “Automatic Video Indexing and Full-Video Search for Object Appearances” *Proceedings of IFIP TC2/WG2.6 Second Working Conference on Visual Database Systems*, pp. 113-127, 1991.
10. B. Shen, “HDH Based Compressed Video Cut Detection” *Proceedings of Visual97 San Diego CA*, pp. 149-156, 1997.
11. N. D. Venkata, B. L. Evans, V. Monga, “Color Error Diffusion Halftoning” *IEEE Signal Processing Magazine*, 20(4), pp. 51-58, 2003.
12. Worldnet User’s Reference Desk, “http://www.wurd.com/pwp_color.php”.
13. R. Lienhart, “Comparison of Automatic Shot Boundary Detection Algorithms” *SPIE Storage and Retrieval for Still Image and Video Databases VII*, 3656, pp. 290-301, 1999.
14. B. Ionescu, D. Coquin, P. Lambert, V. Buzuloiu, ”The Influence of the Color Reduction on Cut Detection in Animation Movies”, *Proceedings of* 20^{eme} *Colloque GRETSI sur le Traitement et l’Analyse du Signal et d’Image*, Louvain-la-Neuve, Belgique, september 2005.
15. Cees G.M. Snoek, M. Worring, ”Multimodal Video Indexing: A Review of the State-of-the-art”, *Multimedia Tools and Applications*, in press, 2005.
16. W.J. Heng, K.N. Ngan, ”Post shot boundary detection technique: flashlight scene determination”, *Proc. of the Fifth International Symposium on Signal Processing and Its Applications*, pp. 447-450, 1999.
17. M.-S. Lee, Y.-M. Yang, S.-W. Lee, ”Automatic video parsing using shot boundary detection and camera operation analysis”, *Pattern Recognition*, Vol. 34, pp.711-719, 2001.
18. B. Ionescu, D. Coquin, P. Lambert, L. Dârlea, ”Color-Based Semantic Characterization of Cartoons”, *IEEE ISSCS - International Symposium on Signals, Circuits and Systems*, Iasi, Romania, july 2005.