U.P.B. Sci. Bull., Series C, Vol. 67, No. 2, 2005

AN APPROACH TO SCENE DETECTION IN ANIMATION MOVIES AND ITS APPLICATIONS

B. IONESCU, D. COQUIN, P. LAMBERT, V. BUZULOIU*

Acest articol propune o metodă imbunatatită de detectare a scenelor video in secvenţe de animaţie. Filmele sunt considerate a fi deja segmentate in plane video. O scenă video este definită ca fiind un ansamblu de plane similare ce au proprietatea de unitate spaţială, temporală, a locaţiei şi a acţiunii. Metoda propusă foloseşte şi compară utilizarea histogramelor medii color normale si ponderate pentru a măsura similarităţile dintre planele video. Diverse aplicaţii ale detecţiei de scene video sunt discutate: corecţia detecţiei de plane video, detecţia tehnicii de filmare "shot-reverse-shot", reprezentarea ierarhică a conţinutului secvenţelor video. Sunt prezentate rezultate experimentale.

In this paper an improved method for detecting scenes in animation movies is proposed. The movies are considered to be already divided into the fundamental video units or shots. A scene is defined as an ensemble of similar neighbor shots which have the properties of unity of space, place, time and action. The proposed approach uses mean color histograms in order to measure the similarities between shots. Two kinds of histograms are proposed and compared: normal histograms and weighted histograms. Various applications of the scene detection are discussed: feedback for cut detection algorithms, detection of the "shot-reverse-shot" camera technique, multi-scale hierarchical representation of the animation movie content. Experimental results are presented.

Keywords: video segmentation, scene detection, color reduction, edge detection, "shot-reverse-shot", video indexing, video abstraction.

Introduction

Increasing networking development and computational power make the accessibility of the video documents very easy through every day desktop PC's. However, current video analysis applications do not provide the desired functions for manipulation of large video databases, as quick content-based search, analysis

* PhD student, The Image Processing and Analysis Laboratory, University "POLITEHNICA" of Bucharest, Romania; Associate Prof., Associate Prof., Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance, Université de Savoie, France; Prof., The Image Processing and Analysis Laboratory, University "POLITEHNICA" of Bucharest, Romania

and comparison of video structure and techniques. Within the last years the video analysis focuses on the semantic interpretation of the video documents.

Thanks to the International Festival of Animation Movies [1], which takes place yearly at Annecy-France since 1960, a very large database of animation movies is available. Managing hundreds of thousands of videos is a very difficult task. An analyzing system which permits browsing and content interpretation is required.

Detecting the video shot boundaries, thus means recovering those elementary video units, provide the ground for nearly all existing video abstraction and high-level video segmentation algorithms [2]. A robust system for the manipulation of video content will necessarily have to deal with segmentation of the visual information into its syntactic units, which implies that an inverse process of the editing stage have to be performed. A video camera produces a temporal video stream of frames organized into two levels of syntactic and semantic structures: **the shot** and **the scene**. A shot is a stream of frames recorded between the time in which the video camera is turned on and turned off. The sequence obtained with these constraints has the characteristic of continuity [3]. In order to build the video sequence the shots are link together using video transitions such as: cuts, fades, dissolves (for more information on video transitions see [2][4][5]).

A scene is a sequence of shots related by semantic features (macro segment). The content of a single scene must have the four Aristotelian properties of unity of space, place, time and action. All the shots sharing these properties are part of the same scene [3]. A study of a rule set for the identification of the macro segments in video documents was proposed in [6]. Rules taken into account are: the way in which gradual transitions are inserted between shots; the distance at which two similar shots are repeated in the video stream where the similarity between frames is checked by considering pointwise differences between low-resolution normalized luminance images; the similarity between contiguous video shots using the mean and standard deviation of hue and saturation for each pixel on a set of selected frames called key-frames; the editing rhythm; the presence of music after silence; the same type of camera motion. A discussion on the video segmentation into shot aggregates (i.e. scenes, episodes) is presented in [7].

In this paper a method for detecting video scenes in animation movies is proposed. Animation movies are more difficult to analyze than natural ones, as the events do not follow a natural way (objects or personages emerge and vanish without respecting any physical rules, the 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 color distribution, artistically

concepts are used (i.e. painting concepts), various animation techniques are specific to one movie (3D, cartoons, animated objects etc).

The movies are considered to be already divided into the fundamental video units or shots. Two approaches are proposed and compared: the first one uses mean normal color histograms while the second approach uses mean weighted color histograms. The color histograms are computed for each shot using a uniformly distributed set of frames.

For the histogram computation, frames are to be sub sampled and color reduced. The weights used for the computation of the weighted color histograms are derived from edge amplitude maps of the retained frames. The similarities between shots are converted into Euclidian distances between their mean color histograms.

Various applications of the proposed scene detection algorithm are discussed: feedback for cut detection algorithms, detection of the "shot-reverse-shot" camera technique and multi-scale hierarchical representation of the animation movies.

1. The scene detection algorithm

As mentioned in the Introduction in order to perform the scene detection the video sequence is already divided into video shots. Each shot will be processed as described in Fig. 1.

A uniformly distributed percent of frames is retained from each video shot. Longer shots are more important in terms of semantic information than shorter ones and they will be represented by more frames.

Each retained frame is sub sampled: for each $n \times n$ pixels frame sub-block only one pixel is retained (n=2..4 depending on the original frame size, for the tests we have used 182x82 pixel frames, with n=4). The colors are then reduced to a manageable number using an error diffusion method performed on the XYZ color space (see Section 2). In parallel a Sobel edge amplitude map is computed (see Section 3).

After the retained frames are subsampled, color reduced and the edge map is computed, the color information is represented by two kinds of histograms: normal color histograms and weighted color histograms. Two different approaches are proposed and compared. The first one summarizes the color information from each shot with the mean normal color histogram of all the retained frames while the second one uses a mean weighted color histogram of all the retained frames.

Histograms are statistical measures of the pixel color distribution and they are invariant to some geometrical transformations in the image. As for the animation movies color is a major feature (each animation movie has its own

particular color distribution) color histograms are an ideal way of representing the general color properties.

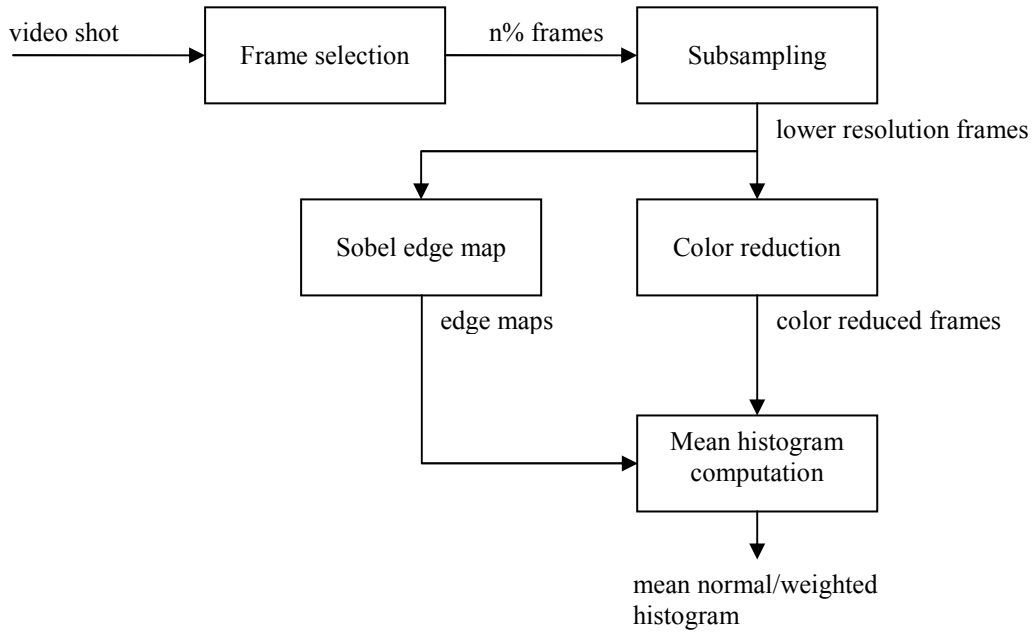


Fig.1. Shot processing for scene detection.

The normal color histogram is defined as:

$$h(i) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \delta(i - f(m,n)) \tag{1}$$

where *f()* is the indexed color image (obtained after the color reduction) of size $M \times N$ pixels, L is the number of colors, $\delta(x - y)$ is 1 if $x = y$ and 0 otherwise, $i = 0 \ldots (L-1)$ and *h()* is a vector of size L.

One major drawback of this kind of histogram is that does not take into account the spatial information, for example two different frames with the same amount and number of colors have the same histogram. In order to add some spatial information the proposed weighted histogram uses a map of frame edges thus giving more importance to contour pixels. It is defined as following:

$$h(i) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \delta(i - f(m,n)) \cdot g(m,n) \tag{2}$$

where *f()* is the indexed color image (obtained after the color reduction) of size $M \times N$ pixels, *g(m,n)* is a gray level image of size $M \times N$ containing the *f()* edges amplitude map (values from 0 to 255, where 255 corresponds to the

strongest edge, see Section 3), L is the number of colors, $\delta(x-y)$ is 1 if $x = y$ and 0 otherwise, $i = 0\ldots(L-1)$ and $h()$ is a vector of size L.

Having each shot of the movie represented either by the mean normal color histogram or by the mean weighted color histogram the shot classification is performed by computing the Euclidian distances between each shot's mean histogram and the others. If a distance between two shots is less than a certain threshold then they are considered to be similar and to belong to the same scene. Only neighbor shots could belong to the same scene (a 10 shots window was used for the search of color similar shots). The classification threshold was determined empirically by experimental tests.

The weighted histograms make the distinction between different textures in the frames, by giving more importance to edge pixels. For example images with the same color distribution but with different texture will lead to identical normal color histograms but to very different weighted color histograms. Experimental results are presented in Section 5.

2. Color reduction

In order to compute the color histogram for a frame, colors have to be reduced to a smaller number (the frames are true color images thus use a 16 million color palette).

The color reduction techniques aim at reducing the number of colors without or with minimal visual loss on the images. They are based on the fact that humans do not distinguish small variations of color hue, therefore some colors can be replaced without producing major changes in the perception of the image. There are a large number of methods proposed to produce the highest quality quantized images. They exploit the excellent algorithms or theoretical issues, such as fuzzy logic, neural networks and genetic algorithms [8]. Depending on the application a compromise between the visual quality and execution time is taken.

In our case the desired property of the color reduction method is to offer a good visual representation of the colors without any color distortion (apparition of new colors in images). In general, color image quantization involves two steps: palette design and pixel mapping. There are two general classes of quantization methods: fixed or universal palette and adaptive or custom palette. Fixed quantization is very fast, but sacrifices the quantization quality while the adaptive quantization determines an optimal set of representative colors. A literature survey on color reduction techniques is presented in [8].

Determining the optimal palette for a frame is time consuming and will lead to a particular color palette for each frame. Comparing different color histograms (computed on different color palettes) is a difficult task and computational time consuming [9]. The proposed color reduction method uses the

error diffusion on the XYZ color space with the selection of colors in the Lab color space from an apriori fixed color palette.

The fixed palette has 216 colors (see Fig. 2.c) and it is proposed in [10] as a web safe color palette designed to avoid dithering of color on various internet browsers. The implemented error diffusion algorithm uses the Floyd and Stenberg filter [11] on the XYZ color space (for an overview of error diffusion techniques see [12]):

$$\frac{1}{16}\begin{bmatrix} 0 & 0 & 0 \\ 0 & -16 & 7 \\ 3 & 5 & 1 \end{bmatrix} \tag{3}$$

where the middle value correspond to the current pixel and the rest of the values to the 8 neighbor pixels.

For the error diffusion the image is scanned from top to bottom and from left to right. For each current pixel its color C is changed with the color within the fixed color palette with the minimum Euclidian distance, $C_{\min}$, computed in the Lab color space. Then, the color approximation error, $E = |C - C_{\min}|$ is diffused using the Floyd-Stenberg filter mask on the XYZ color space. The neighbor pixel values are changes according to the coefficients in the mask, i.e. if the error is E then the south neighbor value V_{South} will change to $V_{South} + \frac{5}{16} \cdot E$, and so on. A color reduction example is presented in Fig. 2*.

Fig. 2.a. Original image

Fig. 2.b. Color reduced image

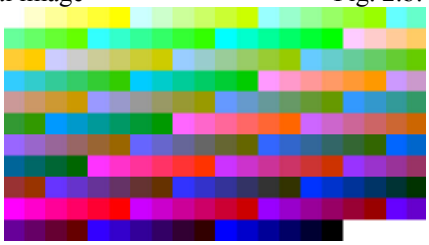


Fig. 2.c. 216 colors safe color palette

* for the full color version see http://alpha.imag.pub.ro, publications section.

3. Edge detection

In order to compute the weighted color histograms an edge map of the frame is required (see Section 1). The edge detection will be performed using the Sobel derivative masks [13]:

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \; S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{4}$$

where S_x is the horizontal derivative mask, S_y is the vertical derivative mask, the middle coefficient correspond to the current analyzed pixel and the others coefficients to its 8 neighbors.

In order to perform the edge detection firstly the frames are reduced to gray level images using a modified luminance value from the YCbCr color space (video standard) adapted to computer color representations [14]:

$$Y = 0.257 \cdot R + 0.504 \cdot G + 0.098 \cdot B + 16 \tag{5}$$

Appling the two Sobel masks on the gray level image, two new images $G_x()$ and $G_y()$ (containing the pixels gradient values) are obtained. The edges amplitude map is defined as following:

$$G_{ampl}(m,n) = \sqrt{G_x^2(m,n) + G_y^2(m,n)} \tag{6}$$

where $G_{ampl}()$ is the edge amplitude image of size $M \times N$ pixels, $m = 1 \ldots M$, $n = 1 \ldots N$.

The desired edge map is obtained after a quantification of the $G_{ampl}()$ image into 256 levels which represent a grey level image of the frame edges map. An example is presented in Fig. 3.

Fig. 3.a. Original image

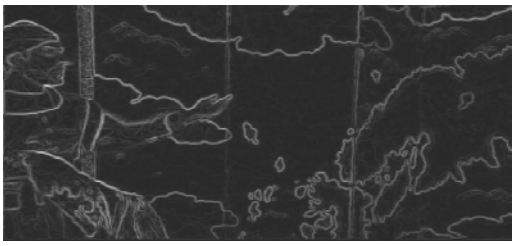


Fig. 3.b. Edge map

4. Scene detection applications

The detection of the shot similarities can be used for various purposes starting with the scene detection. The shot detection algorithms usually do not detect properly all the video transitions and lead to false detections. A false detection consists in the detection of non-existing video transition which splits a

shot into several artificial shots. The proposed scene detection algorithm can be used as a feedback for the shot detection and thus to correct the video transitions false detection. For example if two consecutive shots are found to be similar they belong to the same shot and they were artificially divided.

Also, the scene detection can be used as a detector for a special camera technique called "shot-reverse-shot" [15] (the camera starts filming in a certain scene, it changes the scene and then it returns to the starting scene). This camera effect is specific to pool sport events where the camera focuses alternatively on the players and on the table. In animation movies it contains important semantical information. For example a dialog between two characters produces a repetitive change of camera view between the two. This camera effect is detected by finding similar shots separated by other non similar shots, or finding a shifting between similar and non similar shots.

Another application of the scene detection is the video abstraction which is a compact representation of the video content. An overview of various abstraction techniques is presented in [16]. Almost all abstraction techniques are based on the summarizing the video shots by selecting for each one some representative frames, called key-frames. This kind of abstraction is not scalable. Using the scene detection one can achieve a multi scale abstraction and representation of the video content as presented in Fig. 4.

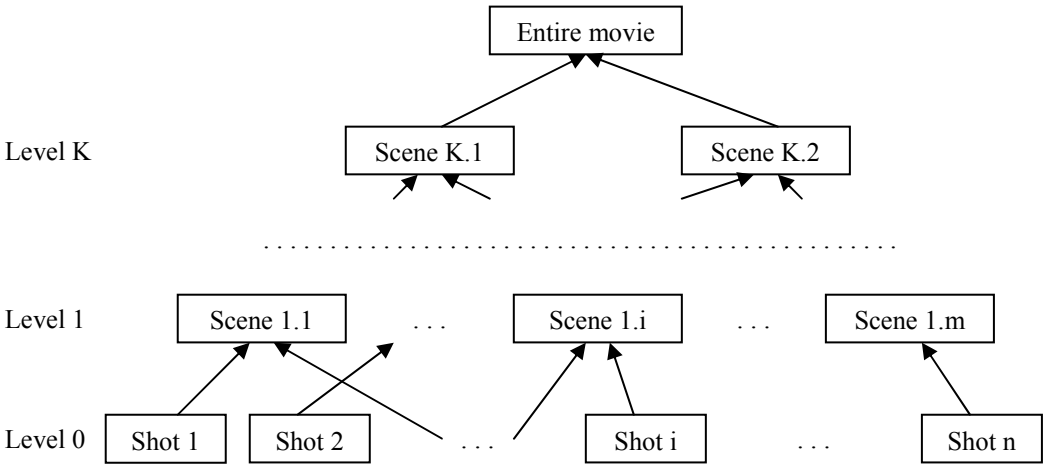


Fig. 4. Multi-scale representation of the video content

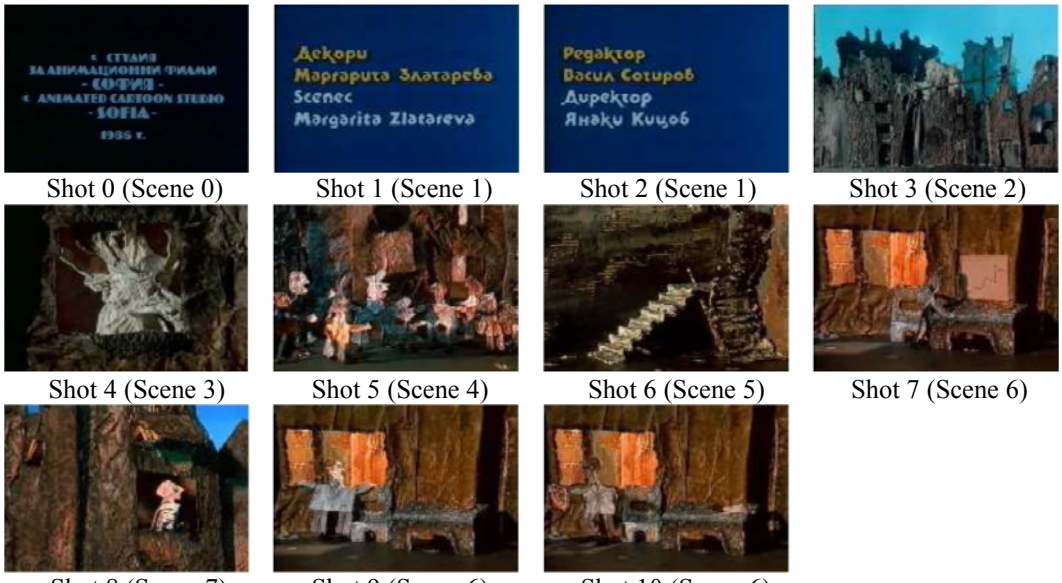
Depending on the application and on the abstract duration constraints one can choose between different levels of detail and abstract lengths, ranging from the highest level of detail but the longest abstract (shot-level or Level 0) and ending with the lowest level of detail but the most compact one (certain key-frames for the entire movie). The video abstraction is obtained by selecting

representative frames for each shot/scene/movie. Various approaches were proposed for determining the key-frames for a shot/scene: selecting the middle frame, selecting a set of uniformly distributed number of frames, selecting the most different frames, selecting the most similar ones etc.

5. Experimental results

The detection of the real scenes is a very difficult task as the concept relay on high level semantic interpretation of the video shots. At this moment the available technology is unable to offer real scene detection and the developed techniques rely on the similarities between various shot characteristics and not on their semantic signification. Also the scene concept itself is interpretable. One may consider some shots belonging to a certain scene while others will not.

As shot similarity criteria we propose the color distribution similarity, because the color is an important feature in the animation movies. Using animation movies the detection is more difficult as they are different of natural ones. Also depending on the producing animation technique (3D, cartoons, glass painting, dummies see [1]) animation movies will share specific textures.



Shot 0 (Scene 0) Shot 1 (Scene 1) Shot 2 (Scene 1) Shot 3 (Scene 2)

Shot 4 (Scene 3) Shot 5 (Scene 4) Shot 6 (Scene 5) Shot 7 (Scene 6)

Shot 8 (Scene 7) Shot 9 (Scene 6) Shot 10 (Scene 6)

Fig. 5. Shots distribution for the test sequence*.

The proposed histogram based scene detection methods were tested on various animation movies. An example is presented below. The test sequence has

* for the full color version see http://alpha.imag.pub.ro, publications section.

a total time of 2min22s at a frame rate of 25frames/s and was manually divided into 11 shots. Its summary is illustrated in Fig. 5 (each shot was represented by its middle frame). The real shots were artificially divided in order to test the detection method (i.e. shot 1 and shot 2 belong to the same shot). The shot lengths are presented in table 1.

Table 1.

Shot lengths (number of frames) for the test animation sequence

Shot 0	Shot 1	Shot 2	Shot 3	Shot 4	Shot 5	Shot 6	Shot 7	Shot 8	Shot 9	Shot 10
281	300	265	253	297	568	489	203	255	300	340

As discussed in Chapter 1 the scene detection is performed by analyzing the shot mean histogram distances. The obtained distances for the test sequence are presented in table 2 and 3 (for each shot a percent of 20% frames were retained, the threshold was experimentally estimated to 0.1, the winning distances are marked with bolded characters).

Table 2.

The distances between shot mean normal color histograms

shot	0	1	2	3	4	5	6	7	8	9	10
0	0	0.749	0.754	0.662	0.421	0.497	0.493	0.691	0.688	0.690	0.689
1	0	0	**0.014**	0.539	0.577	0.551	0.541	0.525	0.548	0.524	0.526
2	0	0	0	0.542	0.581	0.554	0.545	0.528	0.550	0.527	0.529
3	0	0	0	0	0.299	0.263	0.282	0.320	0.199	0.313	0.313
4	0	0	0	0	0	0.130	0.117	0.343	0.292	0.335	0.333
5	0	0	0	0	0	0	**0.070**	0.250	0.217	0.247	0.244
6	0	0	0	0	0	0	0	0.244	0.227	0.234	0.232
7	0	0	0	0	0	0	0	0	0.208	**0.064**	**0.065**
8	0	0	0	0	0	0	0	0	0	0.200	0.199
9	0	0	0	0	0	0	0	0	0	0	**0.009**
10	0	0	0	0	0	0	0	0	0	0	0

Using the normal color histograms (table 2) the following shots were found to be similar: shots 1 and 2 with the distance 0.014, shots 5 and 6 with the distance 0.07, shots 7 and 9 with the distance 0.064, shots 7 and 10 with the distance 0.065, shots 9 and 10 with the distance 0.009. One false detection was obtained for the shots 5 and 6.

Using the mean weighted histograms (table 3) the following shots were found to be similar: shot 1 and 2 with the distance 0.012, shot 4 and 5 with the distance 0.091, shot 5 and 6 with the distance 0.063, shot 7 and 9 with the distance 0.059, shot 7 and 10 with the distance 0.059, shots 9 and 10 with the distance 0.009. Two false detections were obtained for the shot 4 and 5 and for the shot 5 and 6.

Table 3.

The distances between shots mean weighted color histograms

shot	0	1	2	3	4	5	6	7	8	9	10
0	0	0.719	0.719	0.633	0.397	0.467	0.461	0.650	0.648	0.649	0.649
1	0	0	**0.012**	0.511	0.552	0.530	0.523	0.503	0.528	0.503	0.505
2	0	0	0	0.511	0.552	0.531	0.523	0.504	0.528	0.504	0.505
3	0	0	0	0	0.288	0.248	0.267	0.281	0.180	0.276	0.276
4	0	0	0	0	0	**0.091**	0.130	0.315	0.273	0.309	0.307
5	0	0	0	0	0	0	**0.063**	0.236	0.203	0.233	0.230
6	0	0	0	0	0	0	0	0.229	0.216	0.222	0.220
7	0	0	0	0	0	0	0	0	0.190	**0.059**	**0.059**
8	0	0	0	0	0	0	0	0	0	0.185	0.183
9	0	0	0	0	0	0	0	0	0	0	**0.009**
10	0	0	0	0	0	0	0	0	0	0	0

The weighted histograms lead to smaller distance values. The best detection result was obtained with the normal color histograms. That is because the edge colors have the tendency to be the same as the entire video sequence uses the same texture. Using both the proposed methods the "shot-reverse-shot" camera effect (see Chapter 4 and Fig. 5, shots 7, 8 and 9) was successfully detected. Also the subsegmentation of the video sequence could be corrected as the shots 1,2 and 9,10 were found similar thus belonging to the same shot.

Conclusions

In this paper a scene detection method applied to animation movies is proposed. The proposed method relies on the color similarities between consecutive shots. Neighbor shots with the same color distribution are likely to belong to the same scene. For the detection the video sequence is considered to be apriori divided into shots. Each shot is summarized by a percent of frames (key-frames) which are then subsampled, color reduced and a mean color histogram is computed as the mean of all colors histograms for each key-frame. Two different approaches are tested and compared: the use of normal color histograms and weighted color histograms. Within the second approach the edge amplitude map of the frames was used as weights for the weighted histograms. The methods were tested on several movies and an example is presented. Also various applications of the scene detection are discussed: feedback for the cut detection, detection of the "shot-reverse-shot" camera effect, a multi-scale hierarchical representation of the movies. The scene detection is a very difficult task as it relies on the semantic video interpretation and on the human observer. Different observers may have different opinion about the classification of certain shots in scenes. The proposed methods are limited to the color information and adding other information such as motion information or texture analysis is future work. Because of the peculiarity

of the animation movies (they share the same texture from the start to the end) the weighted color histograms were less suited than the normal color histograms and lead to more false detections. Scenes provide important semantic information. A better semantic interpretation of a movie will rely more on the scenes than on the video shots.

Acknowledgments

We thank to the "Centre International du Cinema d'Animation" for the availability of the animation movies and for their support.

R E F E R E N C E S

1. *Centre International du Cinéma 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, **1**(3), 2001, pp 469-486
3. *J.M. Corridoni, A.D. Bimbo*: Film Semantic Analysis, in proceedings of Computer Architectures for Machine Perception, 1995, pp 202-209
4. *R. Brunelli, O. Mich, C.M. Modena*: A Survey on the Automatic Indexing of Video Data, Journal of Visual Communication and Image Representation, **10**, 1999, pp 78-112
5. *A. Dailianas, R.B. Allen, P. England*: Comparison of Automatic Video Segmentation Algorithms, SPIE Integration Issues in Large Commercial Media Delivery Systems, **2615**, 1995, pp 2-16
6. *P. Aigrain, P. Jolly, P. Leplain, V. Longueville*: Medium Knowledge-Based Macro-Segmentation into Sequences, Working Notes of IJCAI Workshop on Intelligent Multimedia Information Retrieval, Montreal, Canada, 1995, pp. 5-14
7. *A.D. Bimbo*: Visual Information Retrieval, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 1999, pp. 231-245
8. *K. Kanjanawanishkul, B. Uyyanonvara*: Novel Fast Color Reduction Algorithm for Time-Constrained Applications, Journal of Visual Communication and Image Representation, 2004, article in press
9. *Y. Rubner, L. Guibas, C. Tomasi*: The Earth's Mover Distance, Multi-Dimensional Scaling and Color-Based Image Retrieval, in Proceedings of the ARPA Image Understanding Workshop, 1997
10. *Worldnet User's Reference Desk*: http://www.wurd.com/pwp_color.php, 2004
11. *R.W. Floyd, L. Steinberg*: An adaptive algorithm for spatial grey scale, in Proc. SID Int. Symp. Digest of Technical Papers, New York, 1975, pp. 36–37
12. *I. Katsavounidis, C.-C.J. Kuo*: A Multiscale Error Diffusion Technique for Digital Halftoning, IEEE Transactions on Image Processing, **6**(3), 1997, pp. 483-490
13. *A.K. Jain*: Fundamentals of Digital Image Processing, Prentice-Hall, Englewood Cliffs, N.J.07632, 1989, pp. 347-357
14. *C. Poynton*: Frequently Asked Questions about Color, http://www.inforamp.net/~poynton, 1999
15. *J.M. Corridoni, A.D. Bimbo*: Structured Digital Video Indexing, in Proceedings of ICPR, 1996
16. *Y. Wang, Z. Liu, J.C. Huang*: Multimedia Content Analysis using both Audio and Visual Clues", IEEE Signal Processing Magazine, 2000, pp. 12-36