Automatic Video Segmentation and Indexing

Youssef CHAHIR[†] and Liming CHEN[‡]

[†] *Laboratoire HEUDIASYC UMR CNRS 6599, Université de Technologie de Compiègne, Centre de Recherches de Royallieu, BP 20529, 60205 Compiègne Cedex, France*

[‡] *Département Mathématiques et Informatique, Ecole Centrale de Lyon, 36, avenue Guy de Collongue BP 163 - 69131 ECULLY Cedex, France*

E-mail: ychahir@hds.utc.fr , liming.chen@ec-lyon.fr

ABSTRACT

Indexing is an important aspect of video database management. Video indexing involves the analysis of video sequences, which is a computationally intensive process. However, effective management of digital video requires robust indexing techniques. The main purpose of our proposed video segmentation is twofold. Firstly, we develop an algorithm that identifies camera shot boundary The approach is based on the use of combination of color histograms and block-based technique. Next, each temporal segment is represented by a color reference frame which specifies the shot similarities and which is used in the constitution of scenes. Experimental results using a variety of videos selected in the corpus of the French Audiovisual National Institute (INA) are presented to demonstrate the effectiveness of performing shot detection, the content characterization of shots and the scene constitution.

Keywords: Video indexing, Content-based video retrieval, Shot similarity,

1. INTRODUCTION

Fast and efficient indexing, browsing and retrieval of video are a necessary for the development of various multimedia database applications. Video data has several different features compared with traditional data such as text, digit, and image. It is composed of image, audio, and text and is a large amount of unstructured data. National or corporate archives store millions of hours of video, e.g. at the French Institut National de l'Audiovisuel (INA). Automatic content-based access to such archives is clearly needed in order to publish these documents. For efficient management of video data , it is necessary to build a video database system., which must be able to supply not only the conventional functions of the database system but also the special functions for content-based retrieval [1,2] to an application. In order to access video by content, a major step is to structure the video representation into different story units [3,4].such shots, which are separated by cuts, and into clusters of contiguous shots (called "scenes", or "sequences") [5]. As it is seen in figure 1 , a video document is composed of one or more sequences. Each sequence is a group of scenes linked together by a definable common thread of action. It is a continuous story, while a scene consists of one or more shots. Generally, it is a continuous chain of actions, which take place in the same place at the same time.. The primary task of video segmentation is the identification of the start and end points of each shot in order to create a set of bookmarks, allowing the access to the video information. The automatic recognition of the beginning and end of each shot can be summarized as solving two problems: (i) avoiding incorrect identification of shot changes due to rapid motion or sudden lighting change in the scene; and (ii) identification of cuts . In the following, we propose new highly performing metrics for automatic detection of these effects, taking into account the information contained in the chromatic component. Experimental evidence of the algorithm's performance is given through an extensive test on about 1h of movies, selected in the corpus of INA, showing different technical characteristics, from the point of view of lighting condition, scene motion, and editing frequency.

The remainder of the paper is organized as follows. Section 2 provides a brief review of existing work on shot detection. Our proposed shot boundary detection scheme is described in section 3. Section 4 presents methods used for color reference frame computing and algorithms of constitution of the scenes. The experimental results for shot boundary detection and scene constitution are presented in section 5. The paper concludes with a summary of the work and future directions in section 6.

Part of the SPIE Conference on Intelligent Robots and Computer Vision XVIII: Algorithms, Techniques, and Active Vision • Boston, Massachusetts • September 1999 SPIE Vol. 3837 • 0277-786X/99/$10.00

2. RELATED WORK

A number of temporal video segmentation methods that employ different similarity metrics have been suggested for both uncompressed and compressed video[6]. These methods can be divided into three classes: pixel/block comparison methods, intensity/color histogram comparison methods, and methods using DCT coefficients in MPEG encoded video sequences[7]. The pixel-based methods detect dissimilarities between two video frames by comparing the differences in intensity/color values of corresponding pixels in the two frames. The number of the pixels changed are counted and a camera break is declared if the percentage of the total number of pixels changed exceeds a certain threshold [8]. These methods produce several false alarms, because camera movement, and moving objects have the effect of a large number of pixel changes, and hence a wrong segment will be detected. To eliminate this difficulty, methods perform the similarity comparison on video blocks rather than pixels have been proposed. A likelihood ratio approach, which divides the video, frames into blocks and the compares the corresponding blocks on the basis of the statistical characteristics of their intensity/color levels[9,8]. However, it is possible that two corresponding blocks can have the same statistics even though their content is different. A number of temporal segmentation methods have also been proposed in the literature in compressed-domain [10,11]. These methods considerably reduce the amount of data to be processed when the video sequence is stored in compressed format. But, their fundamental drawback is that they do not allow automatic management of the content of input video.

Our work is based on the use of combination of intensity/color histograms and block-based technique for comparison is the more favored approach, since the histogram takes into account the global intensity/color characteristics of each frame, thus it is more robust to noise and object/camera motion.

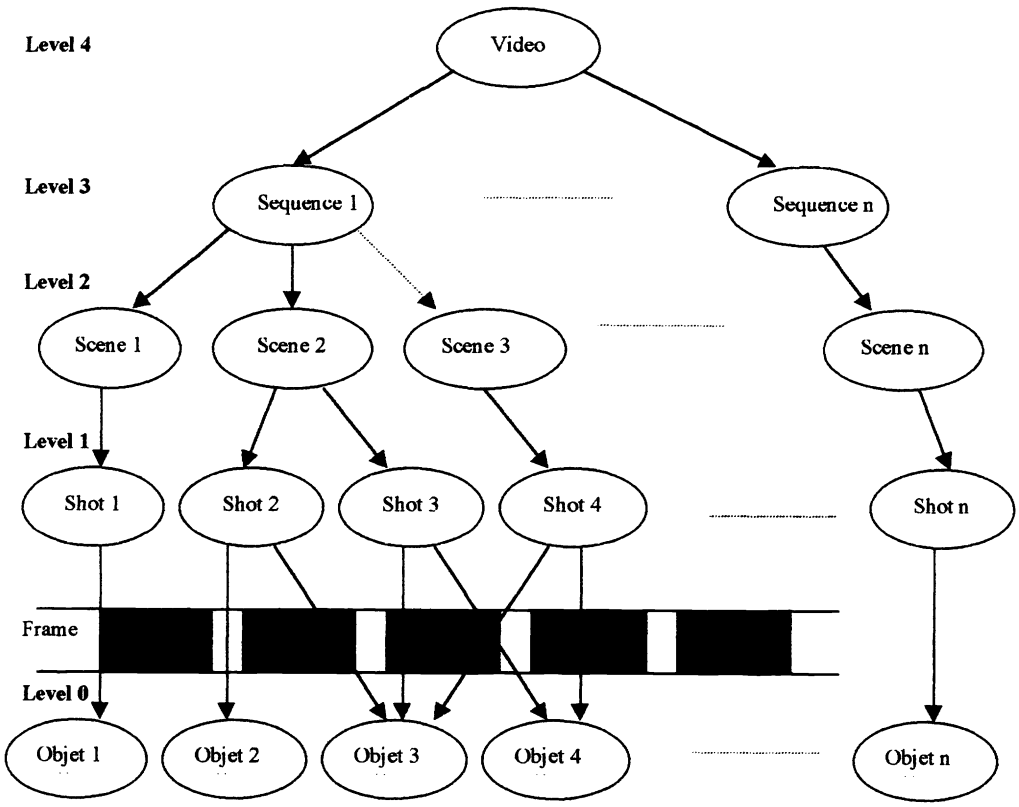


Figure 1 Hierarchy of video data

3. LOCAL FEATURES BASED SHOT BOUNDARY DETECTION

A number of color space representation schemes (e.g., RGB, HSV, and YIQ) have been reported in the literature. We have chosen the CIE L*u*v* color space representation , which separates the brightness feature from the hue because it is close to human perception of colors.

$$\begin{cases} L^* = 25[100Y/Y_0]^{1/3} - 16 \\ u^* = 13\,L^*(u'-u_0) \\ v^* = 13\,L^*(v'-v_0) \end{cases} \quad \text{with} \quad \begin{cases} X = 0.607R + 0.174G + 0.201B \\ Y = 0.299R + 0.587G + 0.114B \\ Z = 0.000R + 0.066G + 1.117B \end{cases}$$

$$\text{and } u' = \frac{4X}{X + 15Y + 3Z};\ v' = \frac{9Y}{X + 15Y + 3Z};$$

$$(Y_0, u_0, v_0) = (1.000, 0.201, 0.461)$$

We use block-based technique which use local attributes to reduce the effect of noise and camera flashes [12]. Here, as in reference [13], each frame is partitioned into a set of blocks called subframes , but with a higher partitioning rate in the horizontal (h) than in the vertical direction (v), because the first is statistically more frequent. In our case, each frame has a resolution of 352 x 288 pixels, h equal to 6 and v equal to 4. We combine the block-based technique with the intensity/color histograms. The histogram takes into account the global intensity/color characteristics of each frame. This combination makes the technique more robust to noise and object/camera motion.

Then, instead of comparing two whole frames, we compare every pair of subframes between two frames to obtain 24 difference values. The difference between two corresponding blocks of consecutive color frames has been derived considering color histograms, in the L*u*v* color space. We compute a histogram on each subframe of the frame and use a histogram difference metric to determine local cuts. .The number of local cuts is counted and a global cut is declared valid if the percentage of the total number of local cuts exceeds a certain threshold. The figure 2 illustrates such process.

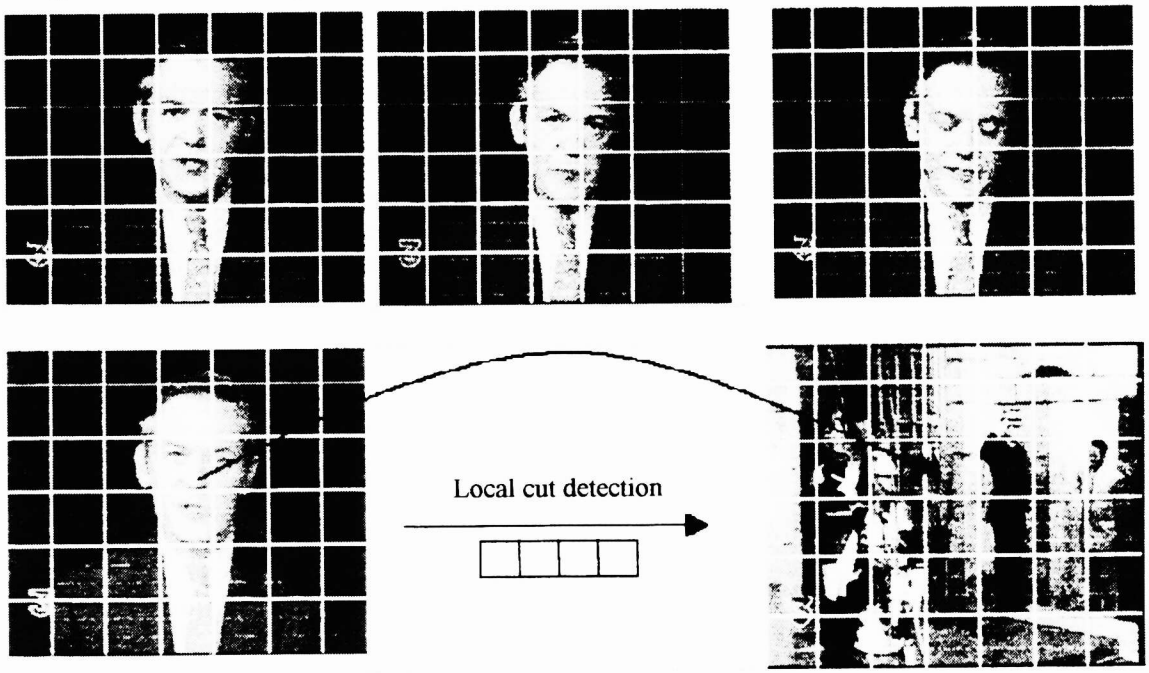


Figure 2 : Combination of color histograms and block-based technique

The choice of the thresholds is a crucial. We use a dynamic threshold, which could vary through the video sequence as in reference [14]. Separately, we compute the visual characteristics of the frame, and use the sum of the weighted distances as a measure of total frame.

Regardless of the color space, color information in an image can be represented either by a single 3-D histogram or three separate 1-D histograms. A histogram is essentially invariant under rotation and translation of an image. A suitable normalization of the histogram also provides scale invariance. Euclidean distance is used as a metric to compute the distance between the feature vectors.

Let $H_i^{L^*}$, $H_i^{u^*}$ and $H_i^{v^*}$ be the normalized color histograms of the first frame f_i, and $H_j^{L^*}$, $H_j^{u^*}$ and $H_j^{v^*}$ be the normalized color histograms of the second frame f_j. The similarity between the two consecutive frames is given by the following equation [15]:

$$\Delta_H = d(H_{f_i}, H_{f_j}) = \frac{\Sigma_L * \min(H_{f_i}^{L^*}, H_{f_j}^{L^*}) + \Sigma_u * \min(H_{f_i}^{u^*}, H_{f_j}^{u^*}) + \Sigma_v * \min(H_{f_i}^{v^*}, H_{f_j}^{L^*})}{\left|H_{f_i}\right| * 3}$$

Note that the value of Δ_H lies in the interval [0,1]. If the frames are identical the $\Delta_H = 1$. In this metric, the intersection measure is incremented by the number of pixels, which are common between the target frame and the query frame.

The intersection histogram is initialized to the histogram of the first frame in the shot, and progressively updated. We also represent a color distribution by its central moments (first four moments), and these are taken to be the texture attributes :

Central Moments: $\mu^n = \sum_{i=1}^{N} (i - \mu_1)^n H[i]$

with $\mu_1 = \sum_{i=1}^{N} iH[i] = Mean$

Variance: $\sigma^2 = \mu^2 = (\sum_{i=1}^{N} (i - \mu)^2 H[i])$

Two other central moment descriptors that are commonly used are the skewness and kurtosis, which are defined from μ^n for n=3 and 4, respectively.

The fundamental variables, mean and variance, provide an indication of how uniform or regular a region is. Skewness is a measure of how much the outliers in the histogram favor one side or another of the mean. It is an indication of symmetry. Finally, kurtosis measures the effect of the outliers on the peak of the distribution, that is, the degree of peakedness. These moments presented above give also a global description of the histogram shape. In order to keep the visual characteristics of the histogram into account , we introduce these first moment in the distance computation. For comparing image similarity between two frame f_1 and f_2 using this feature set, weighted Euclidean distance is used.

$$D(f_1, f_2) = \frac{w_0 \Delta_H + w_1 \Delta_{\mu 1} + w_2 \Delta_{\mu 2} + w_3 \Delta_{\mu 3} + w_4 \Delta_{\mu 4}}{\sum_{i=0}^{4} w_i}$$

where the w_i are the weights assigned to the visual similarity. Experimental evidences has shown that this measure is more robust in matching color images than color histograms [16], and thus is used as one of the color similarity measures in key-frame based retrieval.

4. COLOR REFERENCE FRAME COMPUTING AND SCENE CONSTITUTION

Choice of a reference frames of the shots is a crucial task in automatic video indexing, since their visual features are essential for shot indexing and retrieval. Reference frames must capture the low-level semantics of the shot, in this sense they must allow a description as precise and complete as possible [17].

The computed reference frame is characterized quantitatively by average local histograms. The average local histogram concerns a block, which is computed by accumulating the block histograms. The extraction of shot reference frame scheme is shown in figure 3. In fact, the computed reference image is computed from the average of the totality of the shot. The L*, u*, and v* components of mean color histogram of the first k frames for the i^{th} bin is defined as:

$$H_M^{L*} = \frac{1}{k}\sum_f^k H_f^{L*}(i) \;\; ; \;\; H_M^{u*} = \frac{1}{k}\sum_f^k H_f^{u*}(i) \;\; ; \;\; H_M^{v*} = \frac{1}{k}\sum_f^k H_f^{v*}(i)$$

The scheme allows us to give a measure of the content within the shot, which is deducted from accumulated frame differences.

The representative frame of the shot is the frame of the shot, which is the closest by their visual characteristics to the computed reference frame. For this reason, we compute the difference between the color histogram of each frame of the shot and the mean color histogram. This comparison is done by the chi-square test χ^2, which is the most accepted test for determining whether two distributions are from the same source, or not. Then the distance between a frame f_i and mean frame M can be defined as:

$$\chi^2 = D(H_{f_i}, H_M) = \Sigma L * \frac{(H_{f_i}^{L*} - H_M^{L*})^2}{(H_{f_i}^{L*} + H_M^{L*})^2} + \Sigma u * \frac{(H_{f_i}^{u*} - H_M^{u*})^2}{(H_{f_i}^{u*} + H_M^{u*})^2} + \Sigma v * \frac{(H_{f_i}^{v*} - H_M^{L*})^2}{(H_{f_i}^{v*} + H_M^{L*})^2}$$

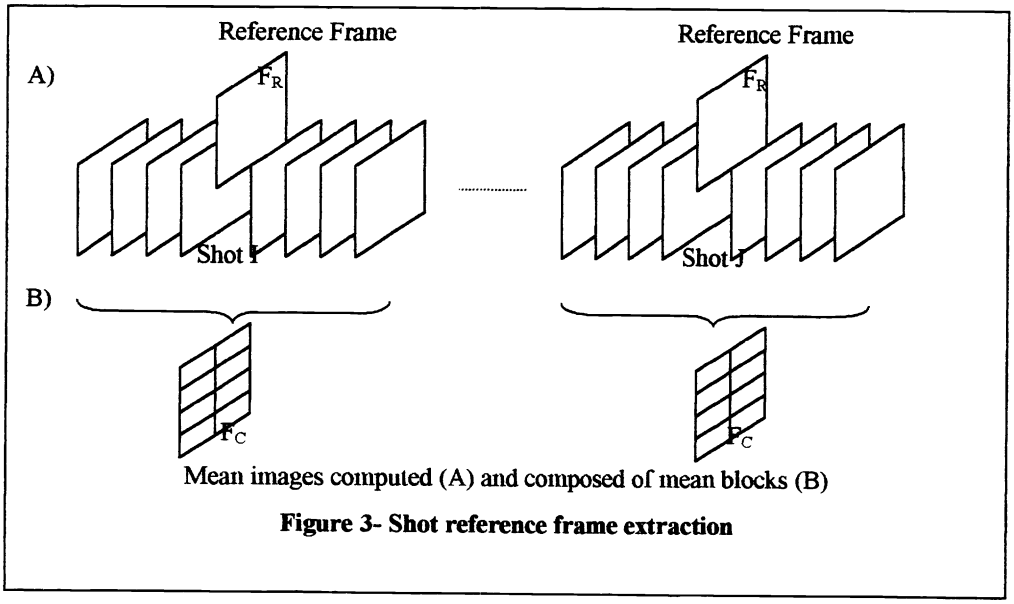
One advantage to use reference frame in browsing is that we are able to browse the video content without necessarily storing the entire video.

The next natural step after shot boundary detection and representative frame selection is to carry out shot similarity based on their visual characteristics. The goal of shot similarity is to enable shot comparison, for video retrieval and browsing, based on visual content and to constitute a scene .

We can define video shot similarity based on the similarities between two shots ,denoted as S_i and S_j, composed of the two key-frame sets $K_i = \{f_{i,m}, m=1,\ldots,M\}$ and $K_j = \{f_{j,n}, n=1,\ldots,N\}$ as:

$$D(S_i, S_j) = \frac{1}{M}\sum_{m=1}^{M} \underset{k=1}{\overset{N}{Max}} [D(f_{i,m}, f_{j,k})]$$

where D is a similarity metric between two images. This definition state that the similarity between two shots is the sum of the most similar pairs of key_frames.



Mean images computed (A) and composed of mean blocks (B)

Figure 3- Shot reference frame extraction

Sometimes, a producer films a scene in its totality and proceeds to a cutting in different shots following the senses that he wants to give to the film (suspense, rhythm,. etc). In the following, we are interesting to the reconstitution of the scene before the cutting and propose an algorithm to do it. We estimate that a scene is a continuous chain of actions represented arbitrarily in shots. In our system, we represent each shot by three frames: the first of the shot, the end of the shot and the color reference frame. The comparison of content of shots based on their color reference frame allows us to deduce the neighborhood between shots and consequently constitute an heuristic on crossing of shots. Thus, we build a sorted vector of close shots for each shot., and we examine each pair of shots comparing the end of the first shot with the start of the second. We estimate that two

shots that follow not lie to the same scene. Otherwise, these two shots would be merged in an alone shot. For each shot, we match the corresponding sorted vector in the decreasing order of similarity.

The proposed algorithm of scene constitution is illustrated in the following steps:

Step 1: initialization of number k of scene, and all shots to unvisited

Step 2: For each shot i, we form a scene cluster (k) by comparing the limit shot of i with the first one of a close shot which is bellow a certain threshold. The computing of the closest shots is given in taking account the shot similarity between i and the other shots. The choice of D is described in section 4. This comparison is done by the chi-square test χ^2 .

```
Input :
        begin : array of start frames of shots
        end : array of limit frames of shots
        close_shot: a matrix of sorted vector of close shots based on sorted similarity between
        shots on the basis on their color reference frame.
        k = close_shot(i,j) means that k is the jth shot which is close to shot i

Step 1:
        For i=0 to n do close_shot(i,j) is unvisited ;
        k=0;
Step 2:
        For i=0 to n do
                j = -1;
                shot i is visited ;
                do
                        begin
                                j = j + 1;
                        end
                while ( ( D (end(i) , begin(close_shot(i,j)) ) > threshold )
                        AND (j<n)
                        AND (shot close_shot(i,j) is unvvisited)) ;

                if ( j<n) then
                        scene(k) = scene(k) U {shot j} ;
                        shot close_shot(i,j) is visited ;
                        k = k +1;
                end
        End
Output :  k scenes constitution
```

5. EXPERIMENTAL RESULTS

In this section, we present the experimental results of our techniques conducted on a Pentium-Pro 300 MHz.. All frames are at resolution 288 * 352 with each pixel represented by 32 bits of data (16M colors). Results are obtained in L*u*v* color space on a 56 minutes video data including recorded TV programs, such as sitcom and commercials. Figure 4 shows a simple comparison of results of global shot detection obtained in RGB space and in the L*u*v* space by using the intersection distance. We observe that artifacts in the L*u*v* space are more clearer than in the RGB space. Also, we see that the cut changes in the RGB space is more sensitive to change illumination than L*u*v*.

The experimental result using the intersection metric applied in the L*u*v* space on each block of frames is shown in figure 5. We have computed local histograms of each block, and we have used the intersection histogram difference metric to determine local cuts. As we can see in figure 5-b, the number of local cuts of the shot 100 which are counted correspond to 20/ 24 . A global cut is then declared valid because the percentage of the total number of local cuts exceeds a 60 %. While in figure 5-a the cut is not very evident. As it is seen, all desirable artifacts appear locally.

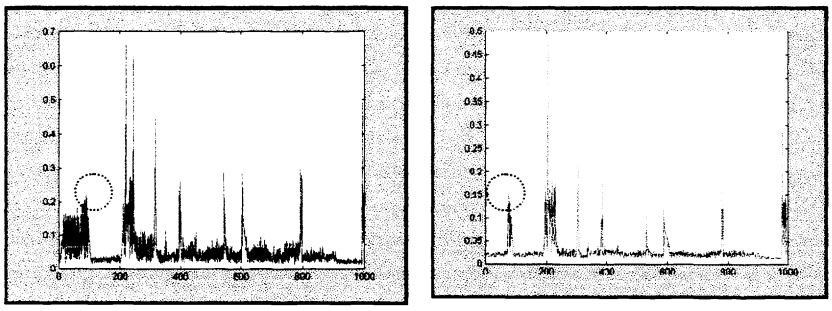
The performance of our shot detection purpose is shown in table 1. Table 1 lists the best performance that was obtained for each of the four videos. This performance is given in terms of precision and recall parameters. Let n_c, n_m, and n_f, respectively, be the number of correct, missed, and false cuts that are generated by a shot detection scheme. The precision recall are then defined as: Precision = n_c/n_c+n_f and recall =n_c/n_c+n_m

To provide an idea about the different shot boundaries, we show in figure 6 a partial set of detected video shots by showing the first frame of each detected clip.

In this section, concerning similarity measures of shots and the scene constitution, we present the experimental results of our techniques applied to 14:43 minutes video sequence corpus selected from *"AIM1MB02"*. It has 369 shots (22082 frames).

For comparison purposes, both the similarity measures of shots based on pairs of key-frames corresponding to table 2 and similarity measures of shots based on reference frames corresponding to table 3 are sorted by decreasing order and recalled in table 4. As we can see, these results are acceptable since only the shot 4 contains shots in the different order. We observe that for the shot 4, the two first close shots are the same (i.e. 2 and 6). On the other hand, the shot 5 is ranked at the 3rd place and the others shots are shifted to the right keeping the same order.

According to tests that we have drawn, we can deduce that the comparison between all pairs of images of the two shots, is equivalent to the comparison of representative images of shots. Figure 7 illustrates an example of detected video scenes. The process is conducted on the 91 first shots of AIM1MB02.



RGB **L*u*v***

Figure 4: Comparison of cut detection based on histogram difference intersection in the RGB and L*u*v* spaces

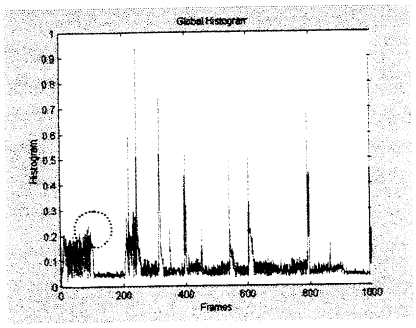


Figure 5-a: Shot boundary detection based on global histograms

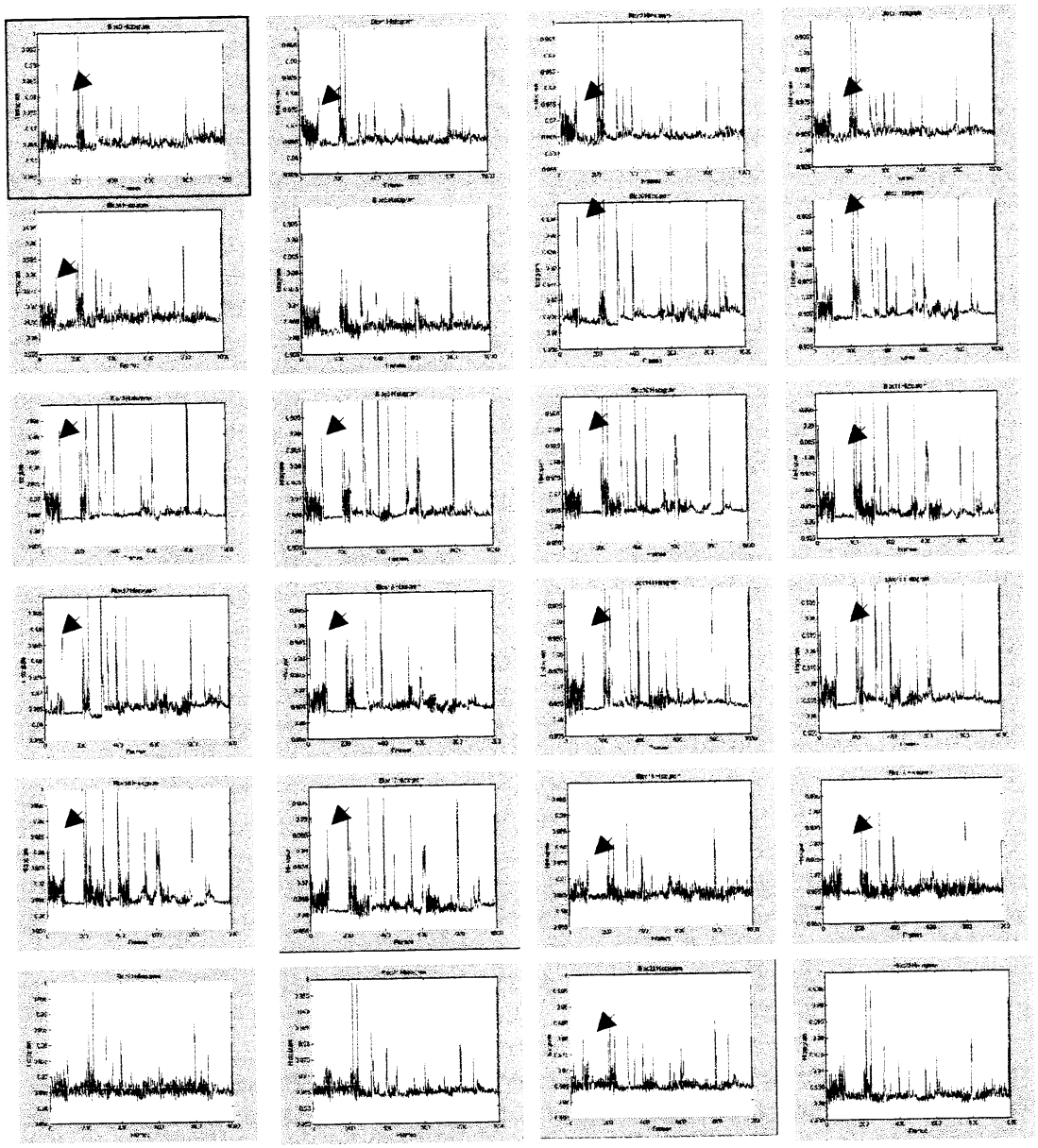


Figure 5-b: Shot boundary detection based on local histograms difference intersection

Video	Total frames	Total Time	Total camera breaks	Correct n_c	False n_f	Missed n_m
AIM1MB02	22082	14:43	369	335	13	21
AIM1MB03	18572	12:22	118	108	7	3
AIM1MB04	19928	13:17	188	170	9	9
AIM1MB05	23144	15:25	166	150	11	5

Table 1. Performance of shot detection on different selections from videos

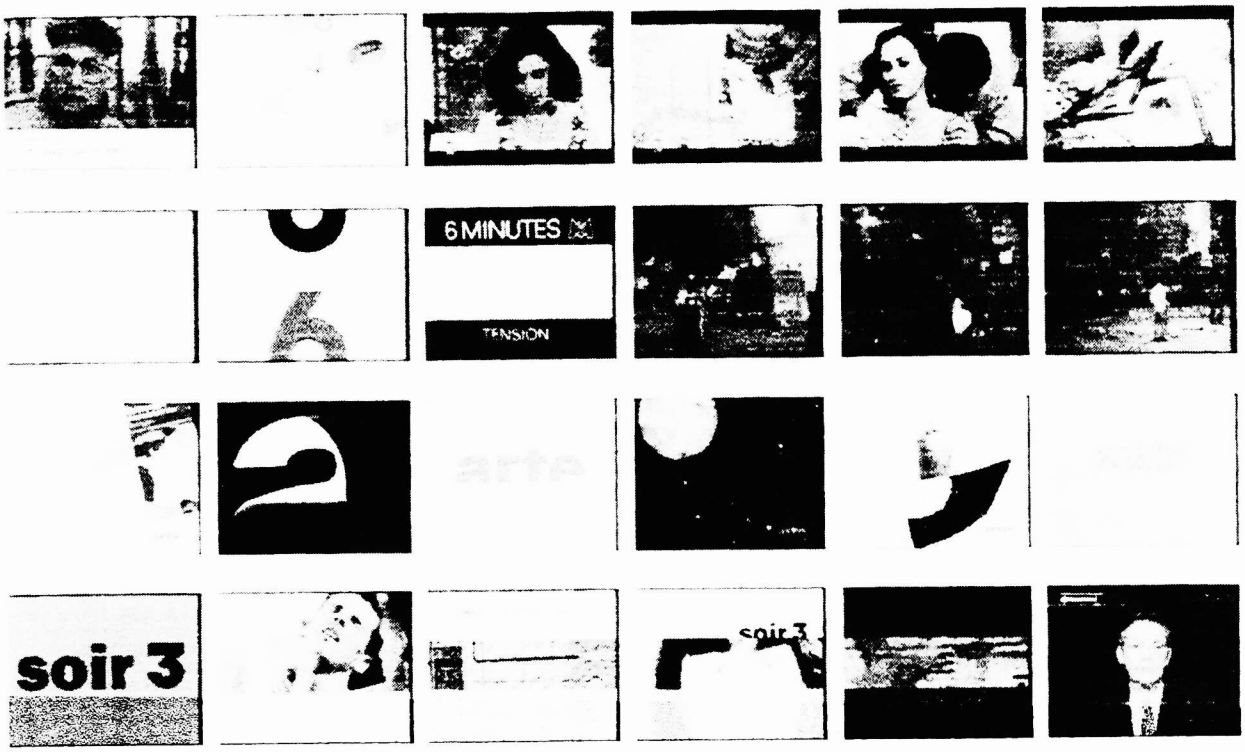


Figure 6: AIM1MB01, (2) AIM1MB02, (3) AIM1MB03, (4) AIM1MB04, and (5) AIM1MB05

AIM02	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
S1	-	0.0769	0.0519	0.0291	0.0334	0.0603	0.0955	0.0551	0.0589	0.0598
S2	0.0769	-	0.0613	0.0332	0.0375	0.0744	0.0673	0.0625	0.0629	0.0672
S3	0.0519	0.0613	-	0.0177	0.0186	0.0348	0.0316	0.0299	0.0301	0.0318
S4	0.0291	0.0332	0.0177	-	0.0190	0.0321	0.0291	0.0272	0.0277	0.0288
S5	0.0334	0.0375	0.0186	0.0190	-	0.0305	0.0264	0.0244	0.0248	0.0271
S6	0.0603	0.0744	0.0348	0.0321	0.0305	-	0.0216	0.199	0.0199	0.214
S7	0.0955	0.0673	0.0316	0.0291	0.0264	0.0216	-	0.185	0.0178	0.192
S8	0.0551	0.0625	0.0299	0.0272	0.0244	0.199	0.185	-	0.0163	0.166
S9	0.0589	0.0629	0.0301	0.0277	0.0248	0.0199	0.0178	0.0163	-	0.153
S10	0.0598	0.0672	0.0318	0.0288	0.0271	0.214	0.192	0.166	0.153	-

Table 2. Similarity measures of shots based on pairs of key-frames

AIM02	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
S1	-	0.564	0.300	0.228	0.298	0.432	0.785	0.312	0.384	0.401
S2	0.564	-	0.439	0.399	0.409	0.503	0.498	0.468	0.479	0.482
S3	0.300	0,439	-	0.153	0.159	0.390	0.228	0.261	0.300	0.238
S4	0.228	0.399	0.153	-	0.259	0.381	0.219	0.795	0.295	0.208
S5	0.298	0.409	0.159	0.259	-	0.291	0.278	0.190	0.267	0.207
S6	0.432	0.503	0.390	0.381	0.291	-	0.287	0.209	0.222	0.801
S7	0.785	0.498	0.228	0,219	0.278	0.287	-	0.781	0.218	0.789
S8	0.312	0.468	0.261	0.795	0.190	0.209	0.781	-	0.178	0.645
S9	0.384	0.479	0.300	0.295	0.267	0.222	0.218	0.187	-	0.543
S10	0.401	0.482	0.238	0.208	0.207	0.801	0.789	0.645	0.543	-

Table 3. Similarity measures of shots based on reference frames

Shots	**Sorted close shots in the decreasing order**	
	pairs of key-frame	**Reference frame**
S1:	7,2,6,10,9,8,3,5,4	7,2,6,10,9,8,3,5,4
S2:	1,6,7,10,9,8,3,5,4	1,6,7,10,9,8,3,5,4
S3:	2,1,6,10,7,9,8,5,4	2,1,6,10,7,9,8,5,4
S4:	*2,6,1,7,10,9,8,5,3*	*2,6,5,1,7,10,9,8,3*
S5:	2,1,6,10,7,9,8,4,3	2,1,6,10,7,9,8,4,3
S6:	10,8,2,1,3,4,5,7,9	10,8,2,1,3,4,5,7,9
S7:	10,8,1,2,3,4,5,6,9	10,8,1,2,3,4,5,6,9
S8:	6,7,10,2,1,3,4,5,9	6,7,10,2,1,3,4,5,9
S9:	10,2,1,3,4,5,6,7,8	10,2,1,3,4,5,6,7,8
S10	6,7,8,9,2,1,3,4,5	6,7,8,9,2,1,3,4,5

Table 4. Comparison of results of close shots from the two similarity measures

Video	Total Scenes	Correctly Classified	Incorrectly Classified	Missed	Successful %
AIM1MB02	32	29	1	2	91%

Table 5. Performance of Scenes constitution on AIM1MB02

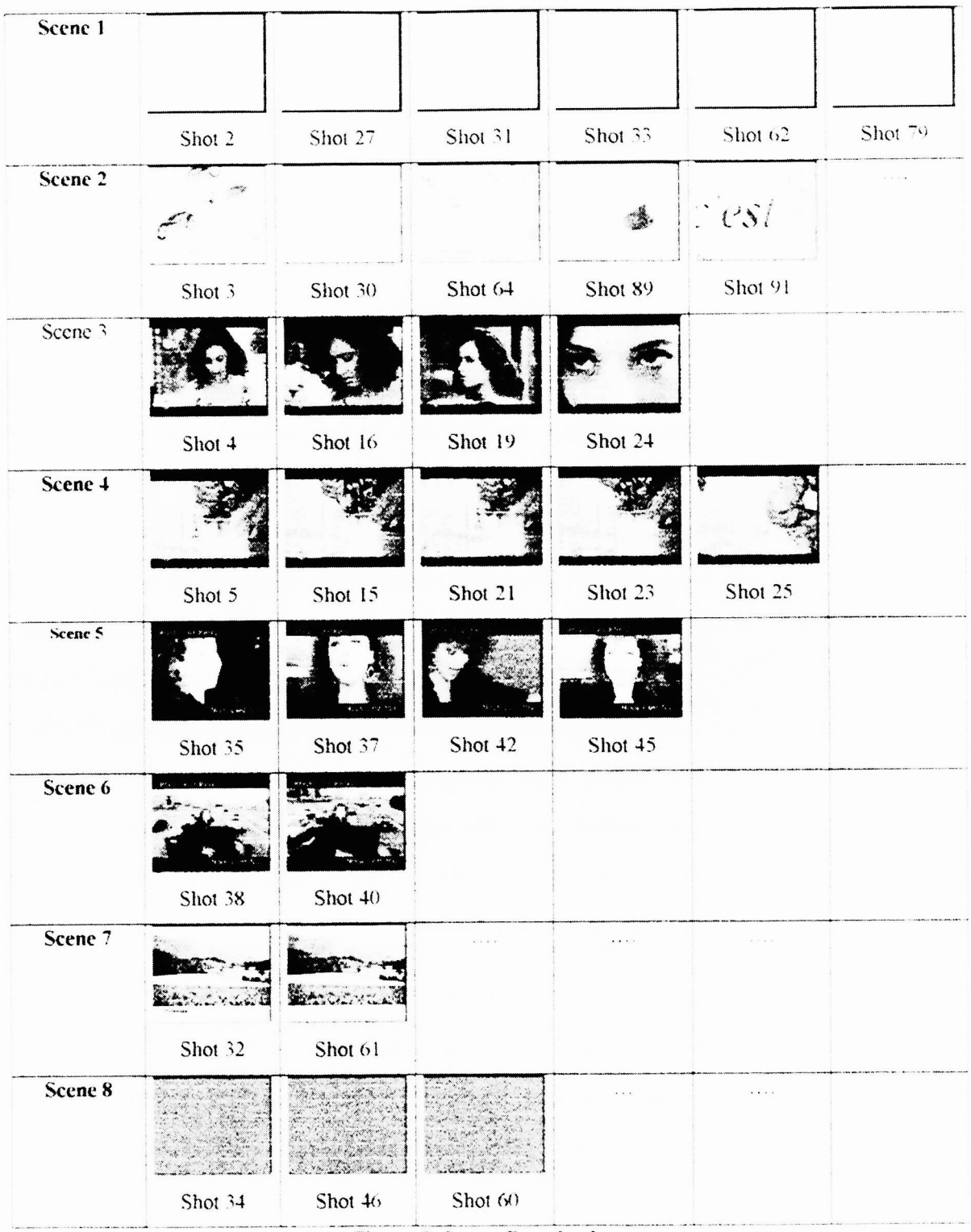


Figure 7: Scene Constitution

6. CONCLUSION AND FUTUR WORK

We have presented a method to detect camera shot boundaries on the basis of local features in automatic way. We have also described the extraction of the color reference frame and our algorithms for the scene constitution.. We concluded that the combined color histograms and block-based shot characterization method is successful and thus can form a good basis for color-based indexing and retrieval of video clips.

In our future work, we aim to classify shots into classes based on color and motion characteristics. Indeed, we try to determine the seven basic camera operations which are fixed, panning (horizontal rotation), tracking (horizontal transverse movement), tilting (vertical rotation), booming (vertical transverse movement), zooming (varying the focusing distance), and dollying (horizontal lateral movement) [6]. We must improve our algorithms for video segmentation. Performance improvements are also necessary. We are also working on JAVA based GUI to the VDBMS which allows the information to be easily distributed over the Internet and World Wide Web.

REFERENCES

1 S.W. Smoliar and H. Shang, "Content-Based Video Indexing and Retrieval", IEEE Multimedia, pp. 62-71, 1994

2 Tzi-cker Chiueh, "Content-Based Image Indexing", Proc. Of the 20 th VLDB Conference, Santiago, Chile, pp. 582-593, 1994

3 G. Davenport et al., "Cinematic primitives for multimedia", IEEE Conf. on Computer Graphics Appl., July 1991

4 D. Swanberg, C. Shu and R. Jain, "Knowledge guided parsing in video databases", Proc. SPIE, San Jose, January 1993

5 L. Chen et al., "Multi-channel video segmentation", Proc. Of SPIE, Multimedia Storage and Archiving Systems II, pp. 252-263, Boston 1996

6 F. Idris and S. Panchanathan, "Review of Image and Video Indexing Techniques", Journal of Visual Communication and Image Representation, Vol.8, N°2, pp.146-166, June 1997

7 B. Günsel et al. "Hierarchical Temporal Video Segmentation and Content Characterization", Proc. Of SPIE, Multimedia Storage and Archiving Systems II, pp.46-56, Dallas 1997

8 H.J.Zhang et al. "Automatic partitioning of full-motion video", ACM/Springer Multimedia Systems, 1(1):10-28, 1993

9 D.Swanberg, C.-F Shu, and R.Jain, "Knowledge guided parsing in video databases", Proc. SPIE 1908, pp.13-24,1993

10 J. Meng et al. "Scene change detection in a MPEG compressed video sequence". In Proc. SPIE, vol. 2419, pp14-25, 1995

11 B. Yeeo and B. Liu, "Rapid scene analysis on compressed video", IEEE Trans. On Circuits and Systems for Video Technology, 5:553-544,Dec.1995

12 S. Shahraray, "Scene change detection and content-based sampling of video sequences", Digital Video Compression: Algorithms Tech. 2419, 1995, 2-13

13 A. Nagasaka and T. Tanaka, "Automatic video Indexing and Full video search for object appearances", in IFIP Trans. Visual Database Systems II, pp. 113-128 (1992)

14 M. Ardebilian ,X. Tu and L Chen ., "Improvement of shot detection methods based on dynamic threshold selection", Proc. Of SPIE, Multimedia Storage and Archiving Systems II, pp. 14-22, Dallas 1997

15 Anil K. Jain and A. Vailaye, "Image Retrieval using Color and Shape", 1995

16 M. Stricker and M. M. Orengo, "Similarity of color images", Proc. IS and SPIE. Storage and Retrieval for Image and Video Databases III, San Jose ,1995

17 E. Ardizzone , M. La Gascia, "Multifeature Image and Video Content-based Storage and Retrieval", Proc. Of SPIE, Multimedia Storage and Archiving Systems II, pp. 265-276, Boston 1996