INDEXING AND RETRIEVAL OF THE MOST CHARACTERISTIC FRAMES / SCENES IN VIDEO DATABASES

Anastasios D. Doulamis, Yannis S. Avrithis, Nikolaos D. Doulamis and Stefanos D. Kollias

Department of Electrical and Computer Engineering
National Technical University of Athens
Heroon Polytechneiou 9, 157 73 Zographou, Greece
e-mail:adoulam@image.ntua.gr

Abstract. An integrated framework for automatic extraction of the most characteristic frames or scenes of a video sequence is presented in this paper. This is accomplished by extracting a collection of a small number of frames or scenes that provide sufficient information about the video sequence. The scene/frame selection mechanism is based on a transformation from the image to a feature domain, which is more suitable for image comparisons, queries and retrieval.

1 INTRODUCTION

The increasing development of advanced multimedia applications requires new technologies for the organization and content based retrieval of digital video databases. In this paper we present an integrated framework for automatic extraction of the most characteristic frames or scenes of a video sequence. This objective is accomplished by extracting a collection of a small number of frames or scenes that provide sufficient information about the video sequence. This can be very useful for multimedia interactive services, e.g., browsing of digital video databases on web pages, or automatic production of video trailers. Moreover, this objective is related to searching of frames or scenes, based on certain features such as motion, luminosity, color, shape and texture, with video indexing as a potential application.

Several approaches have been proposed in the recent literature [1-3], which mainly deal with scene cut detection based on abrupt frame changes [1], as well as with single frame extraction based on the frame properties [2]. Significant improvements on the above techniques are derived in this paper, such as extraction of collections of frames or scenes based on time variation of their properties which sufficiently characterize the video sequence.

In order to achieve extraction of the most characteristic frames or scenes of a video sequence, scene cuts are first detected. A multidimensional feature vector is then generated for each frame. The representation of each frame by a feature vector, apart from reducing storage requirements, transforms the image domain to another domain, more efficient for image queries and retrieval. Based on feature vectors of all frames within a scene, a scene feature vector is computed which characterizes the respective scene. The scene vector is then fed as input to a classifier mechanism which extracts the most characteristic scenes. Finally, the most characteristic frames of a given scene are extracted, based on fluctuation of frame vectors versus time.

2 SCENE DETECTION

The first stage of the proposed method includes the separation of a video stream into scenes, thus a scene cut detection technique is required. This can be achieved by computing the sum of the block motion estimation error over each frame and detecting frames for which this sum exceeds a certain threshold. In MPEG coded video streams scene cuts can be also detected by measuring the bit allocation in B and P frames [4].

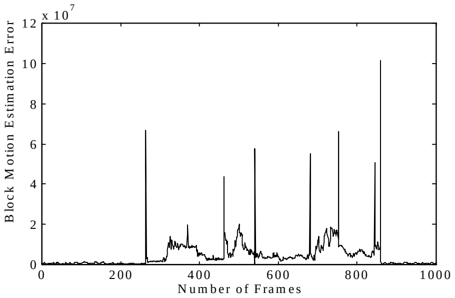


Figure 1. Block motion estimation error versus time (frame number).

A typical graph of the block motion estimation error versus time is shown in Figure 1. The corresponding video sequence is taken from a TV news program of total duration 40 seconds (1000 frames at 25 frames/second). It consists of a main studio scene, 6 scenes of field reporting, and another main studio scene. The scene cut points, as well as the higher motion of the field reporting scenes compared to that of the main studio scenes, are obvious from the graph.

3 FEATURE EXTRACTION

The most important task involved in the process of characterizing video scenes is the extraction of proper frame features such as motion, luminosity, color, shape and texture [3]. For this purpose, a feature vector for each frame of the video sequence is calculated. The feature vector is based on color and motion segmentation of each frame.

3.1 Color segmentation

The color segmentation algorithm splits each frame into regions according to spatial homogeneity criteria. Having formed the segments for each frame, many features are extracted for query or retrieval, such as the number of segments, as well as their location, size, and color. Images consisting of uniform areas will be characterized by large segments, while images containing many small objects or noise, by small and distributed segments.

Block resolution is used for the segmentation in order to reduce computational time and to exploit information of existing coding standards (MPEG). However, regardless of the adopted resolution, oversegmentation can make the feature vector less useful since similar frames may then be characterized by different segments. Hierarchical merging of similar segments is proposed in this paper, depending on segment size apart from spatial homogeneity, in order to eliminate small segments while preserving large ones.

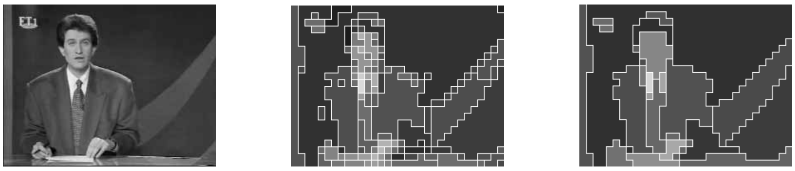
Let us denote by b_k, k=1,2,..,M the blocks of a frame where M is the total number of blocks of each frame. Then, we assume that in the initial state each segment consists of one block and consequently the initial number of segments is M. Thus, the initial segments are $S_k^{(0)} = \{b_k\}$, $k = 1,2,\ldots,M$, where $S_k^{(n)}$ denotes the final k-th segment at the n-th iteration. At every iteration, each segment is merged with its neighboring segments depending on their color and size. Let us assume that $S_k^{(n)}(m)$ is the k-th segment at n-th iteration after merging with m (m=1,2,…,) neighboring segments, $N(S)$ is the set of segments neighboring with segment S, $P(S)$ the number of blocks of S, $f_n(\,)$ a threshold function, $d(\mathbf{x},\mathbf{y})$ a distance metric between the vectors $\mathbf{x}$ and $\mathbf{y}$ and $\mathbf{c}(S)$ the mean color (three dimensional vector) for S. Then

$$S_k^{(n)}(m) = S_k^{(n)}(m-1) \cup S_l^{(n-1)}$$

for all $S_l^{(n-1)} \in N(S_k^{(n)}(m-1))$ such that

$$d(\mathbf{c}(S_l^{(n-1)}), \mathbf{c}(S_k^{(n)}(1)) < f_n(P(S_l^{(n-1)}), P(S_k^{(n)}(m-1))$$

The threshold function $f_n(\,)$ forces the small segments to merge with the large ones while preserving the large segments almost unchanged. Figure 2 depicts the color segmentation of a frame extracted from a TV news program.



(a) (b) (c)

Figure 2. (a) Original frame from TV news program (main studio). (b) First stage of color segmentation. (c) Final stage.

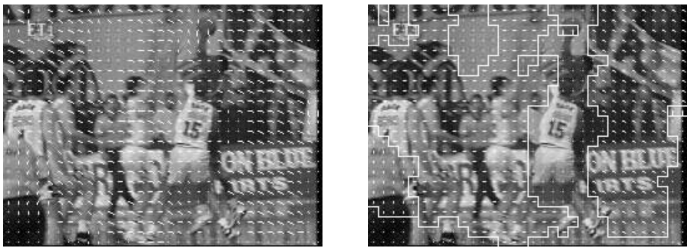
3.2 Motion Segmentation

A similar approach is carried out for the case of motion segmentation, which is an important task, since it gives the ability of indexing, query or categorization of frames according to motion characteristics. In this case, the number, size, and location of the motion segments, as well as the direction and magnitude of the corresponding motion vectors, are derived as motion features.

The proposed procedure is still at block resolution for exploiting properties of the MPEG bit stream. However, the motion vectors, that are computed using either exhaustive or logarithmic search, usually appear “noisy” due to luminosity fluctuations. For elimination of the “noisy” vectors one can use the technique proposed in [5]. Nevertheless, in this paper we also add a penalty term to the equation of motion vector calculation:

$$\hat{\mathbf{v}} = (\hat{v}_x, \hat{v}_y) = \underset{(v_x, v_y) \in A}{\arg\min} \sum_{k=8*i}^{8*i+7} \sum_{l=8*j}^{8*j+7} (u^{(n)}(k,l) - u^{(n-1)}(k - v_x, l - v_y))^2 + D(v_x, v_y)$$

for block (*i*,*j*), where $\hat{\mathbf{v}}$ is the motion vector and $A = \{-a,\ldots,a\} \times \{-a,\ldots,a\}$ is the search area. The penalty term forces the motion vectors to be close to the initial co-ordinates (0,0). To achieve smoothness of motion vectors within a moving area, a median filter is then used for eliminating “noise” while preserving “edges” between regions of different motion, similarly to image restoration applications [6].



(a) (b)

Figure 3. (a) Motion vectors after processing, (b) Final stage of motion segmentation.

After the appropriate motion vectors are extracted, we use a technique similar to the previous one to derive the motion segments. Figure 3 illustrates the motion segmentation of a frame extracted from a TV news program.

3.3 Feature Vector Formulation

All of the above features are gathered in order to form a multidimensional feature vector which is used for collection of information content for each frame. Properties of color or motion segments cannot be used directly as elements of the feature vector, since its size will differ between frames.

To overcome this problem, we classify color as well as motion segments into pre-determined categories, forming a multidimensional “histogram”. To eliminate the possibility of classifying two similar segments at different categories, causing erroneous comparisons, a degree of membership is allocated to each category, resulting in a fuzzy classification [7]. The feature vector is then constructed by calculating the sum, over all segments, of the corresponding degrees of membership, and gathering the above sums into the appropriate categories:

$$F(\mathbf{n}) = \sum_{i=1}^{K} \left\{ \prod_{j=1}^{L} \mu_{n_j}(f_j^{(i)}) \right\}$$

where *K* is the total number of color or motion segments, *L* is number of features (e.g., size, color or location) of segments that are taken into account for classification, $\mathbf{n} = [n_1 \ldots n_L]^T$ specifies the "category" into which a segment is classified, and $n_j \in \{0,1,\ldots,Q\}$ is an index for each feature, where Q is the number of regions into which each feature space is partitioned. The j-th feature, $f_j^{(i)}$, of the i-th segment, S_i, is the j-th element of the vector $[P(S_i)\, \mathbf{c}(S_i)^T\, \mathbf{l}(S_i)^T]^T$ for color segmentation, or $[P(S_i)\, \mathbf{v}(S_i)^T\, \mathbf{l}(S_i)^T]^T$ for motion segmentation, where P, $\mathbf{c}$, $\mathbf{v}$, and $\mathbf{l}$ denote the size, color, motion vector and location of each segment. Finally, $\mu_n(f)$ is the degree of membership of feature f in partition n. Triangular membership functions μ_n are used with 50% overlap between partitions. The feature vector is formed by gathering values $F(\mathbf{n})$ for all combinations of $n_1,\ldots,n_L$, for both color and motion segments.

4 SCENE / FRAME SELECTION

The graph of the feature vector for all frames within a scene indicates the way in which the frame properties fluctuate during a scene period. Consequently, a vector which characterizes the whole scene is constructed by calculating the mean value of feature vectors over the whole duration of a scene.

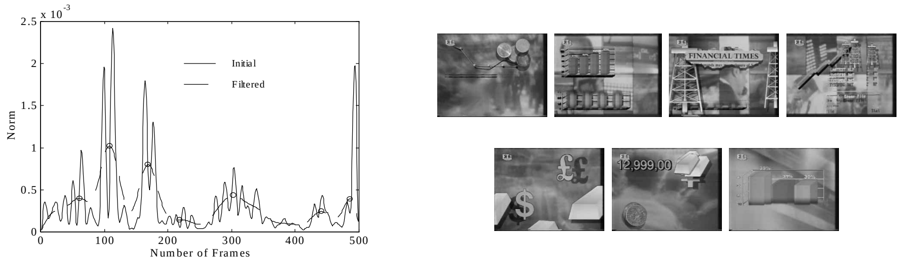
Scene Selection. In applications where extraction of the most characteristic scenes is required, the scene vector is used as input to a neural classifier. The classifier decides whether a particular vector, i.e., the corresponding scene, belongs to the set of the most characteristic scenes or not. The weights of this classifier can be either pre-determined by experts or adapted interactively according to user preferences.

Frame Selection. Other applications require the extraction of the most characteristic frames within a given scene. In this case, the decision mechanism is based on the detection of those frames whose feature vector resides in extreme locations of the graph. For this purpose, the magnitude of the second derivative of the feature vector was used as a curvature measure. Local maxima of this curve were then extracted as the locations (frame numbers) of the characteristic frames.

Video Queries. Once the feature vector is formed as a function of time, a video database can also be searched in order to retrieve frames that possess particular properties, such as dark frames, frames with a lot of motion and so on. The feature vector space is ideal for such comparisons, as it contains all essential frame properties, while its dimension is much less than that of the image space.

5 EXPERIMENTAL RESULTS - CONCLUSIONS

The proposed algorithms were integrated into a system that was tested using several video sequences from video databases. Figure 4(a) depicts a typical graph of the magnitude of the second derivative of the frame feature vector versus time for a scene of such a sequence, of total duration 20 seconds (500 frames). Due to the sensitivity of the segmentation procedure to frame changes, this measure is rather "noisy" (continuous line). For this reason it was first filtered (dashed line) and then its local maxima (small circles) were used for selecting characteristic frames of the sequence. Figure 4(b) illustrates these frames.



(a) (b)

Figure 4. (a) Graph of the magnitude of the second derivative of the frame feature vector versus time for a TV news program. (b) Selected frames.

Extraction of the most characteristic frames or scenes of video sequences taken from large video databases has been presented in this paper, and satisfactory results have been derived. However, several improvements are possible for the proposed system, such as more robust color or motion segmentation algorithms, enhancement of the frame selection mechanism, or inclusion of additional components (shape or texture information) in the feature vector.

6 REFERENCES

[1] Y. Ariki and Y. Saito, "Extraction of TV news articles based on scene cut detection using DCT clustering," Proceedings of ICIP, Sept. 1996, Switzerland.

[2] G. Iyerngar and A.B. Lippman, "Videobook: An experiment in characterization of Video," Proceedings of ICIP, Sept. 1996, Switzerland.

[3] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele and P. Yanker, "Query by image and video content: the QBIC system," IEEE Computer Magazine, pp. 23-32, Sept. 1995.

[4] N. Doulamis, A. Doulamis, G. Konstantoulakis and G. Stassinopoulos, "Performance models for multiplexed VBR MPEG video sources," Proceedings of ICC, Montreal, June 1997 (to appear).

[5] N. Doulamis, A. Tsiodras, A. Doulamis and S. Kollias, "Low bit-rate coding of image sequences using ROI and Neural Networks," Proceedings of IWISP, Manchester, November 1996.

[6] X. You et al, “Robust adaptive estimator for filtering noise in images”, IEEE Trans. Image Processing, pp. 693-699, May 1995.

[7] B. Kosko, “Neural networks and fuzzy systems: a dynamical systems approach to machine intelligence,” Prentice Hall, 1992.