

Local Color Analysis for Scene Break Detection Applied to TV Commercials Recognition*

Juan María Sánchez, Xavier Binefa, Jordi Vitrià, and Petia Radeva

Computer Vision Center, Departament d'Informàtica
Universitat Autònoma de Barcelona, 08193 Bellaterra, Spain
{juanma,xavierb,jordi,petia}@cvc.uab.es

Abstract. TV commercials recognition is a need for advertisers in order to check the fulfillment of their contracts with TV stations. In this paper we present an approach to this problem based on compacting a representative frame of each shot by a PCA of its color histogram. We also present a new algorithm for scene break detection based on the analysis of local color variations in consecutive frames of some specific regions of the image.

1 Introduction

The recognition of known patterns in a given digital video is a task with lots of applications. Among these we have developed a system that recognizes the broadcast of TV commercials on-line and stores their broadcast time, as well as their length. This data can be used by advertisers to check the fulfillment of their contracts with TV stations, as well as to know the impact of their publicity on the viewers, together with audience statistics.

Some previous work has been done about the analysis of TV commercials. The aim of Lienhart et al. in [3] is the isolation and extraction of commercial blocks from TV broadcasts by detecting a set of features common to every commercial block. In the same paper, a recognition-based approach is also implemented to detect single commercials out of commercial blocks, using an approximate substring matching algorithm. Colombo et al. [1] obtain some semantic measures of video sequences from a set of perceptual features in order to characterize every commercial and perform a content-based retrieval on a video database. We have not found any previous work about auditing commercials broadcast. The problem we are dealing with consists on creating a suitable and compact representation to store a digital video sequence pattern, so it can be recognized in a given video. We assume that every video sequence is constituted by a set of shots, which can be represented by a single frame. This assumption forces us to implement an algorithm to correctly detect every transition between shots.

* This work was supported by the projects TAP97-0463 and TAP-0631 of Spanish Industry Ministry.

Different techniques have been used for this purpose, most of them based on color. The simplest one is the color histogram difference between consecutive frames [9]. Some improvements have been proposed by considering the spatial distribution of pixels in the image like the color coherence vectors (CCV) [6] that divide the pixels in each bin into coherent and non coherent, or the joint histograms [7] that build a multidimensional histogram considering a set of local features for each pixel. These algorithms detect almost every sharp transition, but gradual transitions, i.e. dissolves and fades, are not correctly detected due to the low global variation of color from one frame to the next. A completely different alternative is proposed by Zabih et al. in [12], where an algorithm is presented which is based on computing the number of edge pixels that appear and disappear in a given frame as a feature called edge change ratio (ECR). The main problem of this algorithm is dealing with affine transformations of the contents of the image at a low computational cost. The algorithm presented in this paper treats this problem by locally considering some specific regions of the image.

This scene break detection algorithm is introduced in section 2 and our TV commercial recognition system is presented in section 3, showing the way we use principal component analysis (PCA) to reduce the size of the representation of every shot and obtain a real time recognition system.

2 Local Color Analysis for Scene Break Detection

We introduce a sharp and gradual transition detection algorithm based on the analysis of color contents of specific regions in consecutive frames in order to get a measure of the global variation between them as a combination of local differences. Let R_j^i be region j of frame i of a video sequence. A measure of the variation of each region colors from frame i to frame $i + 1$ can be computed as a difference Δh_j^i of their associated color histograms h_j^i and h_j^{i+1} . A global difference measure between frames i and $i + 1$ can then be obtained as a combination of the local differences. The possible transformations of the contents of each region from one frame to the next must be considered in the way we define the pairs of regions (R_j^i, R_j^{i+1}) and how their color histograms are built and compared.

2.1 Regions Around Color Transitions

Following our approach, regions R_j^i are defined from the most significant color transitions of the image, detected as high values of its multispectral gradient. The main interest of these regions is that they can be divided into two sub-regions by the gradient creases. We consider that one of them (O_j^i) belongs to an object in the scene and the other one (B_j^i) belongs to a different object or to the background (or both). Each region R_j^i is defined as follows:

1. The multispectral gradient of the image is obtained using Sobel's approximation.

2. Only significant color transitions are considered applying a threshold, obtained by Otsu's algorithm [5], to the gradient image and removing its smaller connected regions using a queue based algorithm [11].
3. A region R_j^i is defined from each connected region of the gradient image by an adaptative window that contains all of its pixels.

If the content of at least one of these sub-regions does not significantly change from one frame to the next, we can say that the main content of the R_j^i is the same in R_j^{i+1} . This situation can be easily detected if h_j^i and h_j^{i+1} are built in such a way that they contain the same number of pixels from O_j^i and B_j^i . Using a queue based algorithm we know which pixels are at a distance less or equal to d from the border between both sub-regions. With this algorithm the color histogram is exactly the same in every orientation of the contents of the region. Suppose that we have a region in frames i and $i + 1$ which at least one of its sub-regions does not change. Therefore, the values of its corresponding colors in h_j^i and h_j^{i+1} are very close. If we sort their bins to get two vectors $h_j^i(n)$ and $h_j^{i+1}(n)$, $n = 0..N - 1$, such that $|h_j^i(r) - h_j^{i+1}(r)| < |h_j^i(s) - h_j^{i+1}(s)|$, $\forall r < s$, we will find the bins corresponding to the unchanged region in their K first positions, where $\sum_{k=0}^{K-1} h_j^i(k) \approx \sum_{k=0}^{K-1} h_j^{i+1}(k) \approx 0.5$, and the sum of the differences between these K elements of the vectors will be close to zero. Thus, we will know if a region has not continuity in any of its sub-regions when this sum of differences is high. These changed regions will be named Q_j^i .

2.2 Tolerance to Transformations of Regions

Before applying this difference measure between regions, we must find the pairs of regions (R_j^i, R_j^{i+1}) that will be compared, considering the possible transformations of their contents, mainly translations, scalings due to camera zooms and rotations. The window that defines R_j^i on the gradient image is used to find the displacement of this region correlating with its neighbourhood in the gradient image of frame $i + 1$. Changes in the scale of R_j^i contents can mislead the correlation, as shown in figure 1 where it is mistaken as a displacement and the content of the window that defines R_j^{i+1} differs from R_j^i 's. To solve this situation, each connected region of the gradient image of frame $i + 1$ is divided into smaller pieces with a maximum of p pixels and a region R_j^i is defined for each piece, so all of them can be correctly located in frame $i + 1$ as shown in Fig. 1.

Although rotations are mainly considered in the way we compute the color histograms of the regions as we have previously said, when the orientation of the contents of R_j^i changes, the shape of the window that defines it may change as well. Figure 2 shows this situation. We assume that between two consecutive frames, the rotation angle of an element of the scene is small, and so it is the change in the shape of the window that defines R_j^i . This change can be overcome by increasing the window size.

If an R_j^{i+1} can not be found for a given R_j^i , i.e. the maximum of the correlation with its surroundings is close to zero, we know that the contents of R_j^i have



Fig. 1. Tolerance to zooms dividing the connected regions of the gradient image into smaller pieces.



Fig. 2. Change in shape of the window that defines an R_j^i due to a rotation.

completely changed from one frame to the next, with no need of computing the difference between h_j^i and h_j^{i+1} . These regions will be named P_j^i .

2.3 Detecting and Classifying Scene Breaks

Once we know which regions R_j^i do not have continuity in frame $i+1$, we compute a measure of the global variation of the scene between two consecutive frames. The variation of a large region is more significant than a smaller one, so we weigh up each region contribution to the global measure by the number of pixels it contains. Then, this measure is computed as:

$$V(i, i + 1) = \frac{\sum_{j=0}^{L-1} |P_j^i| + \sum_{j=0}^{M-1} |Q_j^i|}{\sum_{j=0}^{N-1} |R_j^i|} \tag{1}$$

where $|R_j^i|$ is the number of pixels in R_j^i . This measure lets us detect and classify different kinds of scene breaks. Cuts are characterized by high values of $V(i, i+1)$ caused by the sudden change of the scene content. Dissolves are mainly due to the P_j^i 's because new edges gradually appear in frame $i + 1$ far from the location of the ones in the previous frames. In a fade-out, every R_j^i turns out to be a P_j^i because all the gradient creases disappear, but a fade-in can not be detected using $V(i, i + 1)$. As gradient creases gradually appear, $V(i + 1, i)$ must be used instead.

As far as we are forced to compute $V(i, i + 1)$ and $V(i + 1, i)$ to detect both kinds of fades, we can also improve the detection of cuts and dissolves, because both of them are high in these transitions, so their sum is a more robust measure than only one of them. Figure 3 shows their values corresponding to each different kind of scene break.

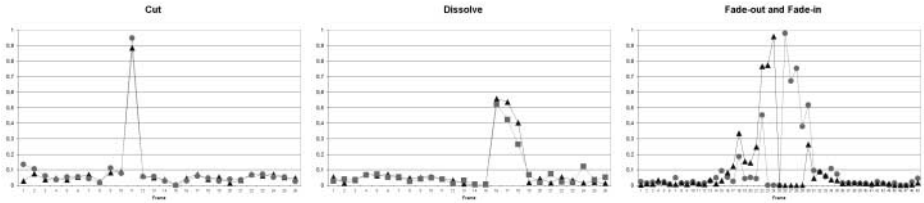


Fig. 3. Values of $V(i, i+1)$ (Δ) and $V(i+1, i)$ (\circ) during a cut (left), a dissolve (center) and both kinds of fades (right).

2.4 Experimental Results

We have tested our algorithm on a video sequence of Spanish TV commercials acquired at 25 frames per second with a 160×120 pixels resolution. This sequence contained several scene breaks including 65 cuts, 13 dissolves, 3 fades-out and 2 fades-in. All of them were correctly detected by our algorithm. 15 false positives were reported during this sequence mainly caused by frame sequences where there are no significant regions in the gradient image. In this situation, regions R_j^i can not be defined and the algorithm behaviour is unexpected. Only 4 of them were due to fast motion of large objects in the scene. For our application to commercial recognition, false positives turn out to be redundancy in their representation, which may be good for the recognition process, while the detection of every scene break is a must.

3 Recognition of TV Commercials

In this section, we introduce our approach to the recognition of previously stored patterns of commercials in a given TV broadcast. The main drawback of commercial recognition is the change of their length in different broadcasts in order to reduce their cost for advertisers. This reduction is achieved removing shots and/or shortening their length. Our representation of a commercial assumes that every video sequence can be divided into shots. The information contained in each shot can be represented by a single image. For efficiency, we have chosen the first frame of a shot as its representative. Then, a commercial is stored as the sequence of representative frames of its shots.

The recognition process is achieved detecting the transitions between shots, getting their first frames and searching the stored database for the matching ones. The database is looked over in such a way that the elements with the highest matching probability are first compared. Let's suppose that we have previously detected and identified a shot of the input video sequence, so we know which commercial is being broadcasted. When we detect the next shot, it is compared with every shot representative of the current commercial to know if it is still being broadcasted. If it finished, the new shot can very probably be the first one of another commercial, so its representative frame is compared with

every first shot representative of the commercials in the database. Even if none of them matched it, it still must be compared with the rest of shot representatives of every stored commercial because the first shot of a commercial can be removed.

The comparison of shot representatives is based on their color histograms in order to achieve some invariance to small changes in the representative frame of the shot, as can be caused by removing its first few frames. Color histograms are affected by color intensity variations in different TV stations broadcasts, which can be modeled as a shift in each color channel histogram. Every representative frame color channel is normalized depending on its average intensity value before searching the database.

3.1 Dimensional Reduction Using PCA

If we want to work with a large commercial database in real time, the size of the elements to compare should be reduced to very few dimensions. This is achieved applying PCA to the color histograms of shot representatives. This statistical technique has been used extensively for object recognition tasks (Turk and Pentland [10], Nayar et al. [4], Huttenlocher et al. [2], Seales et al. [8]).

Let h be the color histogram of a frame represented as a column vector. Given a set of learning histograms h_m , $m = 1..M$, we define the matrix $X = [h_1 - c, \dots, h_M - c]$, where c is the average of the h_m 's. The average histogram is subtracted from each h_m so that the predominant eigenvectors of XX^T will capture the maximal variation of the original set of histograms. The eigenvectors of XX^T are an orthogonal basis in terms of which the h_m 's can be rewritten. Let e_i , $i = 1..N$, denote each of its eigenvectors and let E be the matrix $[e_1, \dots, e_N]$. Each h_m is rewritten in terms of the orthogonal basis defined by the eigenvectors of XX^T as $g_m = E^T(h_m - c)$. As distances are preserved under an orthonormal change of basis, it can be shown that $\|h_m - h_n\|^2 = \|g_m - g_n\|^2$ [8].

The most important fact is that h_m can be approximated using just those eigenvectors corresponding to the k largest eigenvalues, rather than all N eigenvectors, where $k \ll N$. This low-dimensional representation is intended to capture the important characteristics of the set of learning images. Let f_m be the vector of coefficients of g_m corresponding to the k largest eigenvalues. Then, the sum of squared differences, $\|h_m - h_n\|^2$, is approximated as $\|f_m - f_n\|^2$.

This size reduction of the elements to compare allows us to speed up the recognition process by computing difference measures of the elements in the k -dimensional subspace instead of in the original N -dimensional space, while the maximal variation of the original set is kept.

3.2 Real Application Results

We have tested our system on real prime time Spanish TV broadcasts, using a PC and acquiring the video sequence directly with a Matrox Rainbow Runner video capture board. Although our scene break detection algorithm could not be fully used in real time because it has not been optimized yet, we have implemented it as a helper to a simple color histogram difference algorithm. When this measure



(a) Representative frames of the 10 first shots of a “Little Maidness” commercial.



(b) Representative frames of the 5 shots of a different “Little Maidness” commercial.

Fig. 4. Two different commercials of the same product.

gets over a low threshold (which reports many false positives), our new algorithm is asked to confirm the detection of the transition.

We have learned 23 commercials in our system, with a total of 399 shot representatives. Some of these commercials were broadcasted more than once during the test sequence. The learned commercials appeared a total of 81 times during the 9 hours the system was running. All of them were recognized and their lengths were exactly reported, including those broadcasted by a different TV station of the learning one. The test sequence also contained different commercials of the same product with some similar shots like the ones shown in Fig. 4. Some cases were reported correctly, but some were not because the learned key frames did not correspond to the ones found during the broadcast. Although some of them are similar, the projections of their color histograms do not match. A better key frame selection would let us detect these commercials. However, as far as they are actually different commercials, both of them should be learned if they must be recognized by the system.

Only 4 shots were reported as wrong commercials during this test. They can easily be detected by the user if one frame is stored for every detected commercial.

4 Conclusions

This paper has introduced two contributions to digital video analysis. First of all, we have developed an application of pattern recognition in digital video

sequences based on their segmentation into shots, and on the dimensional reduction of their representative frames, using PCA of their color histograms, to speed up the recognition process and achieve the goal of commercial recognition during TV broadcasts in real time. Our system has shown to be very accurate detecting the broadcast of commercials as well as their length.

On the other hand, this approach forces us to correctly detect transitions between the shots of the sequence, including cuts, dissolves and the two kinds of fades. We have introduced a new scene break detection algorithm based on the local analysis of color variation between consecutive frames in some specific regions of the image. We have chosen the regions around color transitions due to its high perceptual content that lets us interpret what is happening in each one of them. The experimental results have been very satisfying. Future work has to be done optimizing the algorithm in order to fully use it in real time applications.

References

1. C. Colombo, A. Del Bimbo, and P. Pala. Retrieval of commercials by video semantics. In *Proc. Computer Vision and Pattern Recognition*, pages 572–577, 1998. 237
2. D. P. Huttenlocher, R. H. Lilien, and C. F. Olson. Object recognition using subspace methods. In *Proc. of the 4th European Conference on Computer Vision*, volume I, pages 536–545, April 1996. 242
3. R. Lienhart, C. Kuhmünch, and W. Effelsberg. On the detection and recognition of television commercials. In *Proc. IEEE Conf. on Multimedia Computing and Systems*, pages 509–516, Ottawa, Canada, June 1997. 237
4. S. K. Nayar, H. Murase, and S. A. Nene. Parametric appearance representation. In *Early Visual Learning*, pages 131–160. Oxford University Press, 1996. 242
5. N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics, SMC*, 9(1):62–66, January 1979. 239
6. G. Pass and R. Zabih. Histogram refinement for content-based image retrieval. In *Proc. of the 3rd Workshop on Applications of Computer Vision*, Sarasota, Florida, December 1996. 238
7. G. Pass and R. Zabih. Comparing images using joint histograms. *ACM Journal of Multimedia Systems*, 1998. (to appear). 238
8. W. B. Seales, C. J. Yuan, W. Hu, and M. D. Cuts. Object recognition in compressed imagery. *Image and Vision Computing*, 16:337–352, 1998. 242
9. M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991. 238
10. M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991. 242
11. L. Vincent. Algorithmes morphologiques à base de files d’attente et de lacets. Extension aux graphes, Ph. D. dissertation, Ecole Nationale Supérieure des Mines de Paris, France, 1990. 239
12. R. Zabih, J. Miller, and K. Mai. A feature-based algorithm for detecting and classifying scene breaks. In *ACM Conference on Multimedia*, San Francisco, California, November 1995. 238