COMPARISON OF AUTOMATIC SHOT BOUNDARY DETECTION ALGORITHMS BASED ON COLOR, EDGES AND WAVELETS

Gjorgji Madzarov, Suzana Loskovska, Ivica Dimitrovski, Dejan Gjorgjevikj
Department of Computer Science
Faculty of Electrical Engineering and Information Technology
Karpos 2 b.b., 1000 Skopje, Macedonia
Tel: +389 2 3099159; fax: +389 2 3064262
e-mail: madzarovg@feit.ukim.edu.mk

ABSTRACT

Shot boundary detection is fundamental to video analysis since it segments a video into its basic components. This paper presents a comparison of several shot boundary detection techniques and their variations including color histogram, edge directions histogram and wavelet transformations statistics. The performance and ease of selecting good thresholds for these algorithms are evaluated based on a wide variety of video sequences with different object and camera motions. Threshold selection is performed using sliding window. We used TV news, sports and documentary, music, movie and nature video sequences to estimate the performance of the algorithms. The experimental results indicate that the algorithm based on color histograms is most suitable for shot boundary detection in film and documentary categories, but the algorithm based on wavelet is preferable for nature and sports categories.

1 INTRODUCTION

In multimedia information retrieval, shot boundary detection is a very active research topic [1], [2], [3]. Today, a typical end-user of a multimedia system is overwhelmed with video collections. Organizing these collections, so they are easily accessible, is a major problem. Thus, to enable efficient browsing of multimedia materials, it is necessary to design techniques and methods for indexing and retrieving this kind of data. We focus on video data, as it is one of the richest, but also most resource consuming part of multimedia content. Digital video information often consists of series of 25 frames or images per second and an associated and synchronized audio track. To develop any content-based manipulations on digital video information, the video information must be structured and broken down into components. Digital video can be described with four different levels of details: complete video, video scenes, video shots and frames (Figure 1). At the lowest level, the video consists of a set of frames. At the next, higher level, frames are grouped into shots. Consecutive shots are aggregated into scenes based on story-telling coherence. All scenes together compose the entire video sequence.

Shots are basic structural building blocks in video. A shot in video information may be defined as a sequence of continuous images (frames) from a single camera at a time. A shot boundary is the gap between two shots. Naturally, boundaries between shots need to be determined automatically [4]. After the boundaries are found, each shot can be represented with an appropriate key frame. Key frames are used to encapsulate the content of the video sequence, and to apply indexing and browsing.

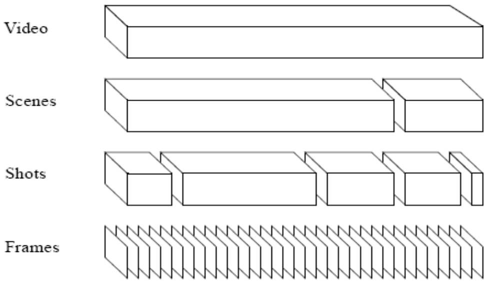

Figure 1: *Video structuring model*

A shot cut is a shot boundary where one shot abruptly changes to another. In shot cuts there is a sudden transition from one shot to another, i. e. one frame belongs to the first shot and the following frame belongs to the second shot. Other types of shot boundaries include fades, dissolves or wipes [5]. These shot boundaries types include gradual transition between two shots, i. e. there exists a sequence of frames that belongs to both the first and the second shot. "Detecting a cut" means precise positioning of the change of shots. In this paper we focus on detecting shot cuts, since they contribute roughly 90% of all shot boundaries present in video collections, as opposed to 10% presence of gradual transitions.

The remainder of the paper is organized as follows. Section 2 introduces several shot boundary algorithms; Section 3 describes the algorithm evaluation technique and the data used for testing methods. Section 4 presents the experimental results and Section 5 gives a conclusion of the paper.

2 SHOT BOUNDARY DETECTION ALGORITHMS

The task of any shot boundary detection method applied on video sequence is to detect the visual discontinuities along the time domain. During the detection process, it is crucial to extract the visual features that measure the degree of similarity between consecutive frames in a given shot.

2.1 Color histogram

The color histogram-based shot boundary detection is one of the most reliable variants of histogram-based detection algorithms [6]. It considers that color content does not change rapidly within, but across shots. Thus, shot cuts and also gradual transitions, can be detected as single peaks in the time series of the differences between color histograms of continuous frames. Often, digital images are represented in RGB color space. In our work, we used 24 bits/pixel images (8 bits for every color channel). The overall number of possible colors levels is 2^{24} bins. Due to the limited response of human visual system, we are not able to distinguish the whole levels of possible colors. A simple solution considers only the most significant bits of each RGB component (Figure 2). This solution reduces computational overhead and increases robustness toward simple camera and object motion.

R_7	R_6	R_5	R_4	R_3	R_2	R_1	R_0
G_7	G_6	G_5	G_4	G_3	G_2	G_1	G_0
B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0

Figure 2: *Color quantization*

With this quantization method all possible colors are grouped into 2^{12} different color levels in RGB space. This corresponds to 4096 colors.

2.2 Edge direction histogram

Edges characterize boundaries and therefore are a problem of fundamental importance in image processing. Edges in images are areas with strong intensity contrasts. Edge detecting an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image. There are many ways to perform edge detection [7]. The most convenient way to represent the distribution of edges in image is by using edge direction histogram. The edge direction histogram is composed of 72 bins corresponding to intervals of 2.5 degrees. Two Sobel filters are applied to obtain the gradient of the horizontal and the vertical edges of the luminance frame image [8]. These values are used to compute the gradient of each pixel. Those pixels that exhibit a gradient above a predefined threshold are taken to compute the gradient angle and then the histogram.

2.3 Multiresolution wavelet analysis

Multiresolution wavelet analysis provides representations of image data in which both spatial and frequency information are present [9]. In multiresolution wavelet analysis we have four bands for each level of resolution resulting from the application of two filters, a low-pass filter (L) and a high-pass filter (H). The filters are applied in pairs in the four combinations, LL, LH, HL and HH, and followed by a decimation phase that halves the resulting image size. The final image, of the same size as the original, contains a smoothed version of the original image (LL band) and three bands of details. Each band corresponds to a coefficient matrix that can be used to reconstruct the original image. These bands contain information about the content of the image in terms of general image layout (the LL band) and in terms of details (edges, textures, etc...). In our procedure the features are extracted from the luminance image using a three-step Daubechies multiresolution wavelet decomposition that uses 16 coefficients and producing ten sub-bands [10] (Figure 3). Two energy features, the mean and standard deviation of the coefficients, are then computed for each of the 10 sub-band obtained, resulting in a 20-valued descriptor.

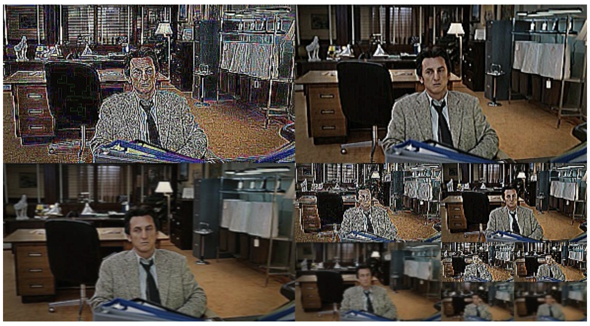

Figure 3: *Example image with applied wavelet transformation*

2.4 Threshold selection

The shot boundary detection method is based on the difference between histograms of frames belonging to a video sequence. This difference is computed using Manhatan distance:

$$d\ (H_i, H_{i-1}) = \sum_{j=1}^{M} \left| H_i(j) - H_{i-1}(j) \right|$$

or Euclidean distance:

$$d\ (H_i, H_{i-1}) = \sqrt{\sum_{j=1}^{M} \left(H_i(j) - H_{i-1}(j) \right)^2}$$

where H_i and H_{i-1} are the histograms for frame $F(i)$ and $F(i-1)$.

We use Manhatan distance for color histograms and Euclidian distance for edge direction histograms and wavelet statistic parameters.

Most of the existing approaches for shot boundary detection that are based on frame differences compare peaks in the histogram difference graph with a previously obtained threshold value. Differences that reach above the threshold value represent detected shot cuts. Figure 3 shows the result of computing the histogram difference for a given video sequence. In the figure a peak appears when a large discontinuity occurs between histograms. These peaks are usually associated to an abrupt transition. From all appearing peaks in Figure 4, a real shot cut is represented only with the peak appearing at frame 4090. All other peaks are caused by the intensive object movement in front of the camera. It is evident that selecting the threshold value is a problem of its own. Selecting too high threshold value increases the number of missed shot cuts. Using a lower threshold results in increasing the number of false alarms. A way to eliminate the peaks caused by the camera or objects motion, has to be derived.

Figure 4 shows that an abrupt scene transition produces only one peak value within a period of time. Therefore, we consider a sliding window of size $2n+1$ along the axis that covers frame transitions Dif[i-n], ..., Dif[i+n]. Next we compute the local mean-ratio within the sliding window, for each frame:

$$M_i = \frac{\sum_{j=i-n, j\neq i}^{j=i+n} Dif[j]}{2n}$$

Then we map the histogram difference curve into the local mean-ratio space. The histogram difference value at frame i is now equal to its original value Dif[i] divided by the mean M_i of the appropriate sliding window:

$$Dif^*[i] = \frac{M_i}{Dif[i]}$$

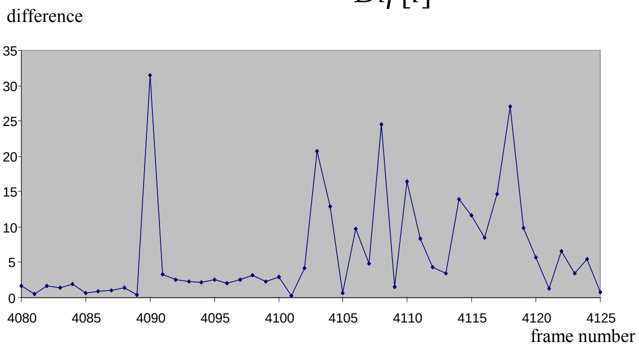

Figure 4: *Frame difference*

Figure 5 shows the transformed color histogram difference for a window of size n=5 applied on the same video sequence from the Figure 4.

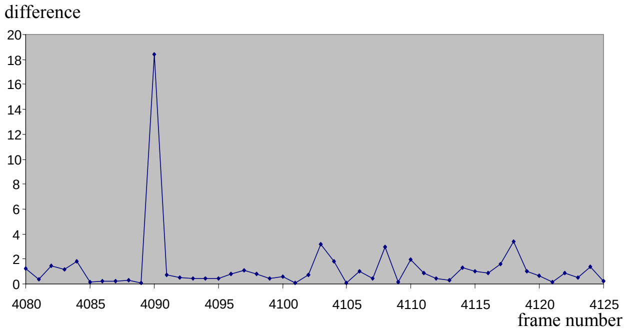

Figure 5: *Transformed frame difference*

It is evident that false alarms appearing in the original difference graph are eliminated. The only peak that appears in this curve is the actual shot cut at frame 4090. The chosen value of 5 for the half length of the sliding window n, is empirically derived from various experiments. Choosing a greater value than 5 increases the danger of including two shot cuts in a single sliding window.

3 QUALITY OF DETECTION

When evaluating our shot boundary detection methods we compared results with a listing of the actual shot cuts (when and where they occur). There are a number of parameters that should be considered when evaluating shot boundary detection methods, but the most important are:

- N_i – number of false shot boundaries detected by the method
- N_d – number of shot boundaries not detected by the method
- N_t – number of actual shot boundaries

Having these values, the measures can be calculated. In our work we used the following:

$$\text{Recall} = \frac{N_t - N_d}{N_t} \qquad \text{Precision} = \frac{N_t - N_d}{(N_t - N_d) + N_i}$$

The recall measure looks at the percentage of actual shot cuts that has been detected by the method, while the precision measure is a percentage showing how accurate the method is at detecting only the actual shot boundary.

4 EXPERIMENTAL RESULTS

To conduct a comprehensive test of the implemented algorithms, we selected a variety of video clips as test data. The categories of the selected videos are presented in Table 1. The locations of the actual shot boundaries in the test videos were determined by a manual visual analysis.

We conducted numerous experiments with a variety of video contents to compare the performance of automatic shot boundary detection algorithms based on color, edges and wavelet transformation. The videos are mainly with low resolution and contain the difficult aspects that challenge the scene change detection algorithms like camera motions, rapid moving objects, zooms, flickers, and often combinations of these. Our results are presented in Table 1. The shot boundary detection algorithm based on edge direction histogram gave the worst results. These results are expected because of the low resolution and quality of our test videos. The best value for the recall parameter is obtained with the algorithm based on wavelet transformation. The precision parameter is the best for the algorithm based on color histograms. The algorithm based on color histograms is most suitable for film and documentary. On the other hand, the algorithm based on wavelet is preferable for nature and sports categories. We obtained the worst results for the music category because of the fast transitions and fast camera movements. The results for news category are not satisfactory because of the low resolution and bad quality for the videos from this category.

Number of frames	Number of Shots	Type	Recall - %	Precision - %
edge direction histogram				
7900	35	News	85	56
3000	103	Music	78	97
6760	49	Film	89	80
5570	96	Documentary	87	79
2665	11	Nature	73	40
3000	29	Sport	90	53
color histogram				
7900	35	News	85	100
3000	103	Music	85	98
6760	49	Film	98	100
5570	96	Documentary	96	95
2665	11	Nature	91	83
3000	29	Sport	97	100
wavelet statistic parameters				
7900	35	News	97	79
3000	103	Music	87	99
6760	49	Film	96	66
5570	96	Documentary	95	98
2665	11	Nature	100	100
3000	29	Sport	97	100

Table 1: *Experimental results*

5 CONCLUSION

In this paper we present comparison results of shot boundary detection algorithms based on color, edges and wavelet transformation. It has been demonstrated that different algorithms performed differently for various video categories. The video quality is crucial for accurate shot boundary detection, especially for the algorithm based on edge direction histogram. Sport and nature categories are much easier for shot boundary detection compare to rest of the categories. This is because of the small number of shots and presence of long video sequences without fast transitions.

References

[1] M. Petkovic, R. van Zwol, H.E. Blok, W. Jonker, P.M.G. Apers, M. Windhouwer, M. Kersten, “Content-based Video Indexing for the Support of Digital Library Search”, *Proceedings of the 18th International Conference on Data Engineering (ICDE 2002)*

[2] K. Shirahama, K. Ideno and K. Uehara, “Video Data Mining: Mining Semantic Patterns with temporal constraints from Movies”, *Proceedings of the Seventh IEEE International Symposium on Multimedia (ISM 2005)*

[3] E. Bruno, D. Pellerin, “Video structuring, indexing and retrieval based on global motion wavelet coefficients”, *Proc. Int. Conf. of Pattern Recognition (ICPR)*, Quebec City, Canada, 2002.

[4] J. S. Boresczky and L. A. Rowe., “A Comparison of Video Shot Boundary Detection Techniques”, *Storage & Retrieval for Image and Video Databases IV, SPIE 2670*, pp 170-179, 1996.

[5] R. Lienhart, “Comparison of Automatic Shot Boundary Detection Algorithms”, *Proc. Storage and Retrieval for Image and Video Databases* (*SPIE*), Vol. 3656, pp 290-301, 1998

[6] R. A. Joyce, B. Liu, “Temporal Segmentation of Video Using Frame and Histogram Space”, *IEEE Transactions on multimedia*, vol. 8, no. 1, February 2006

[7] D. Ziou, S. Tabbone, ,”Edge Detection Techniques An Overview”, *International Journal of Pattern Recognition and Image Analysis*, 8(4), pp. 537-559, 1998.

[8] R. Gonzales and R. Woods, “Digital Image Processing”, Addison Wesley, 1992;414-428

[9] F. Idris, S. Panchanathan, “Storage and retrieval of compressed images using wavelet vector quantization”, *Journal of Visual Languages and Computing*, 1997;8:289-301.

[10] P. Scheunders, S. Livens, G. Van de Wouwer, P. Vautrot, D. Van Dyck, “Wavelet-based texture analysis”, *Journal Computer Science and Information management*, 998;1(2):22-34.