Scene Detection Methods for MPEG – Encoded Video Signals

Georgios Akrivas, Nikolaos. D. Doulamis, Anastasios. D. Doulamis and Stefanos. D. Kollias

Abstract—**This paper evaluates the performance of three scene cut detection algorithms for MPEG-coded video data. A large amount of data taken from actual video sequences is presented to evaluate the scene results. For each method, the percentage of false, as well as failed, scene detection is used as indication of the method performance. The effect of the selected MPEG bitrate and resolution on the performance of the methods are further investigated, to evaluate the methods' robustness, especially in low bit rate applications. Conclusions on the appropriateness of the methodologies used in the framework of the targeted applications are finally derived.**

Index terms—**scene detection, MPEG compressed domain, dc coefficients.**

I. INTRODUCTION

With the rapid progress in computer technologies, there has been an explosion in the amount of visual information being generated, stored, accessed, transmitted and analyzed. Traditionally, video is represented as a sequence of numerous consecutive frames, each of which corresponds to a constant time interval [1]. Although such a linear representation is appropriate for viewing an image sequence in a movie mode, it has a number of limitations for the new emerging multimedia applications, such as content-based indexing and retrieval, video browsing and summarization. This is due to the fact that it is time consuming and tedious to sequentially (linearly) scan a video frame by frame to locate content of interest.

Scene detection can be considered as the first stage of a non-sequential (non-linear) video representation [1]. This is due to the fact that a scene corresponds to a continuous action of a single camera operation and thus application of a scene detection algorithm partitions the video into "meaningful" time entities. For this reason, scene cut detection algorithms is first applied to video indexing and retrieval systems to extract characteristics frames (*key-frames*) and shots (*key-shots*) on which video queries can be applied [2]. Furthermore, efficient detection of scenes from an image sequence is also useful for coding purposes, since different coding methods can be used according to the scene content.

For this reason, scene detection algorithms have attracted a great research interest recently, especially in the framework of the MPEG-4 and MPEG-7 standards and several algorithms have been reported in the literature dealing with the detection of cut, fading or dissolve changes either in the compressed or uncompressed domain. The purpose of this paper is to examine the performance of three different methods for scene detection, all directly applied to MPEG-coded video sequences thus making full MPEG decoding unnecessary. The performance is measured on large amounts of real – life video data, and under a variety of parameters. Only abrupt scene changes are examined, due to difficulty in finding large amounts of real – life gradual scene changes.

II. SCENE DETECTION METHODS

Two of the examined methods are presented in the paper of Yeo and Liu [3]. The basic idea of both techniques is that the dc coefficients on an 8x8 block of a frame can form a sufficient representation of the image content as far as the scene cut detection is concerned. Furthermore, the use of this spatially reduced image (called "dc image" in [3]) can increase the detection performance due to its smoothing effect, which partially removes the motion impact. A fast approximation for estimating the dc coefficient, or equivalently the dc image, in the case of P- and B- frames is also presented in this paper, based on the weighted mean of four overlapping blocks [3]. Motion vectors are used for the estimation of the weights involved in the previous computation. The advance of this approach is that it avoids motion compensation/estimation required for calculating the dc images in case of P and B frames, which is computationally expensive.

Using this approximation and comparing each two subsequent images, with an appropriate metric, a sequence of differences is created. Abrupt scene changes manifest themselves as sharp peaks at this metric sequence. The algorithm should detect these peaks against the signal

This work is partially funded by the Greek Secretariat of Research and Technology (Project PENED 99ED 478).
The Author are with the national technical university of Athens (NTUA), department of electrical and computer engineering, 9 Heroon Polytechniou Str., 15773 Zografou, Athens, Greece.
Corresponding author email: gakrivas@image.ntua.gr

noise. According to the method used for constructing the metric sequence, two different approaches can be discriminated.

The first uses the absolute difference of the dc images as metric for peak detection (*Method A*). As the authors mention in [3], this scheme is sensitive to motion complexity for not spatially reduced images (full frames). However, use of dc images compensates possible instabilities of motion to a large extent, as explaine d in section IV, where evaluation of the obtained results is performed.

The second approach, which is actually insensitive to the "motion noise" has been also reported in [3] and based on histograms of dc images. However, this approach has the risk that two consecutive scenes may appear similar dc histograms since it is highly probable to be produces in the same environment. We call this method *Method B* in the rest of this paper.

Then, peak detection is performed for both methods by a simultaneous satisfaction of two conditions. The first detects peaks in the metric sequence as points of maximum value within an interval of m frame width, centered around the examined point. To reduce the probability, however, of detecting noisy peaks, a second condition has been also applied in [3]. According to this, the peak points should be also n times greater than the second largest value within the aforementioned interval. As can be seen, the first condition ensures the peak existence, while the second enforces the sharpness of the peak.

To further increase, the robustness of peak detection, we have included another third condition, which exploits the absolute difference between two successive frames, and excludes short peaks that come as a result of errors of inter frames. More specifically, if this difference exceeds a pre-determined threshold peaks are detected. In our case, the threshold equals $d \times M \times N$, where *M*x*N* are the dimensions of the frame and *d* is real-value parameter.

The need of the third condition is depicted in Figure 1, which shows the absolute difference among consecutive dc images versus the frame number for a real video sequence. Actual scene changes happen only at frame numbers 12 and 99, while the rest are artifacts caused by the prediction errors.

Besides choosing the right combination of above mentioned parameters, the scene detection performance can be increased by exploiting information of the chrominance data. More specifically,

- Scene changes are detected as peaks in one of the three chrominance channels.
- Scene changes are detected as peaks in all the three chrominance channels.
- Scene changes are detected based on a linear combination of the difference sequences over the three chrominance channels. In this case, the metric sequence of the absolute difference is obtained as $d = (1-c)d_L + cd_c$ where d_L is the corresponding difference for the luminance channel, while d_c is the difference for the chrominance channel. Furthermore, c is an appropriate parameter.

The third option was found to be the most efficient in our experiments since it increases the signal to noise ratio (SNR).

Finally, the method, presented in the paper of Ariki and Saito [4], has been also evaluated for scene detection in our paper as third examined technique (*Method C*). In particular, the algorithm forms a cluster by taking a group of p subsequent frames and computing mean values μ_i and standard deviations σ_i for the dc coefficients and the zig-zag scanned. The subscript i corresponds to the ith DCT coefficient. Thus, i=0 denotes the dc while i=1 the first ac. Each coefficient x_i is considered outside the cluster if $|x_i - \mu_i| > \varphi\sigma_i$, where φ is a real parameter. The frame is considered to fall outside the cluster, if the number of DCT coefficients that are outside of the interval exceeds a threshold. The number of consecutive frames that fall outside the cluster is measured. When the number reaches p, a new scene is detected, and a new cluster is computed. Otherwise, the properties of the cluster are updated. In our implementation, we use the dc coefficients for both luminance and chrominance, and the ac coefficients for luminance only. Furthermore, we have used the approximation of Yeo & Liu to perform the computational of the ac coefficients.

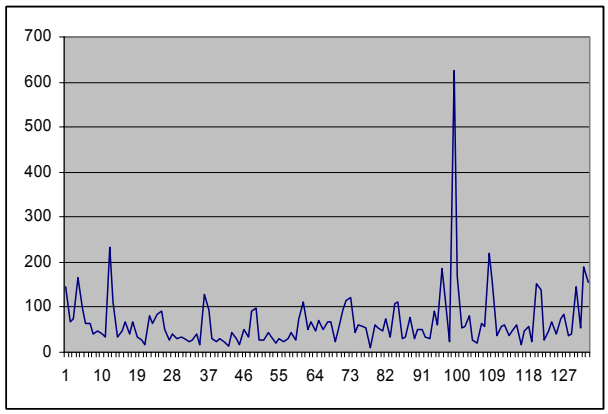

Figure 1: Absolute difference of consecutive dc images.

III. EVALUATION OF THE SCENE DETECTION ALGORITHMS

Twenty (20) minutes of MPEG coded video data, taken from two movies, consisting of total 231 scenes was used in this paper to evaluate the performance of the three scene cut detection algorithms, mentioned in the previous section.

The video sequences were captured using VHS tapes and have been coded based on the Ligos LSX 2.51 MPEG Encoder. Two different resolutions were used, SIF (frames of 352x288 pixels) and QSIF (frames of 176x144 pixels), while the bitrates were ranged from 600 kbps to 1800 kbps for SIF and 150 kbps to 450 kbps for QSIF.

To perform the experiment all parameters involved, have been regulated using a video sequence of small duration, so that the minimum false alarm (wrong scene detection) and failed detection (scenes which have no detected) was reached. It should be mentioned that these errors are in general contradictory. Furthermore, the optimal combination of luminance and chrominance coefficients was investigated. Then, for the examined sequences, two measurements were calculated

- The number of false scenes detected
- The number of failed scene detection

Method A (SIF Images)				
Bitrate (kbps)	Absolute Error		Relative Error (%)	
600	4	2	1,7	0,9
900	4	2	1,7	0,9
1200	4	2	1,7	0,9
1500	3	2	1,3	0,9
1800	3	2	1,3	0,9

Table I: Scene cut detection performance of method A at different bitrates in case of SIF images.

Method A (QSIF Images)				
Bitrate (kbps)	Absolute Error		Relative Error (%)	
150	6	4	2,6	1,7
225	6	4	2,6	1,7
300	6	4	2,6	1,7
375	6	4	2,6	1,7
450	6	4	2,6	1,7

Table II: Scene cut detection performance of method A at different bitrates in case of QSIF images.

Method B (SIF Images)				
Bitrate (kbps)	Absolute Error		Relative Error (%)	
600	12	40	5,2	17
900	33	32	14	14
1200	36	32	16	14
1500	22	37	9,5	16
1800	17	38	7,4	16

Table III: Scene cut detection performance of method B at different bitrates in case of SIF images.

Tables I and II present the results obtained using the method A in case of SIF and QSIF images. In these tables, both the absolute and the relative errors are presented for false alarms and failed detection.

Similarly, the results of the method B are depicted in Tables III and IV for SIF and QSIF images respectively. Finally, the scene detection performance for the third method is presented in Table V and VI.

A more clearly comparison of the performance for the three examined methods (Method A, b and C) is depicted in Figure 2 where the relative prediction error both for false alarms and failed detection is presented in case of bitrate equal to 1200 kbps for SIF image resolution.

Method B (QSIF Images)				
Bitrate (kbps)	Absolute		Relative (%)	
150	98	45	42	19
225	101	46	44	20
300	108	48	47	21
375	99	47	43	20
450	94	45	40	19

Table IV: Scene cut detection performance of method B at different bitrates in case of QSIF images.

Method C (QSIF Images)				
Bitrate (kbps)	Absolute Error		Relative Error (%)	
600	20	6	8,7	2,6
900	20	6	8,7	2,6
1200	20	6	8,7	2,6
1500	19	6	8,0	2,6
1800	19	6	8,2	2,6

Table V: Scene cut detection performance of method C at different bitrates in case of SIF images.

Method C (QSIF Images)				
Bitrate (kbps)	Absolute Error		Relative Error (%)	
150	18	7	7,8	3,0
225	19	5	8,2	2,2
300	17	6	7,4	2,6
375	16	6	6,9	2,6
450	16	6	6,9	2,6

Table VI: Scene cut detection performance of method C at different bitrates in case of QSIF images.

IV. DISCUSSION ON THE RESULTS OBTAINED

The first and most obvious conclusion one can make is that the method of the absolute difference (Method A) is superior to that of comparing histograms (Method B). As it has been mentioned above, when comparing histograms, there is the risk of misclassifying consecutive scene, which

belong to almost the same environment, and hence present very similar histograms. This is indeed quite often in commercial films as the material for the experiments came from.

The method of DCT clustering (Method C) stands between the above two. One should note, however, that this method is also able to classify scene of similar content, while is adequate for gradual scene change detection, whereas the methods A and B fail.

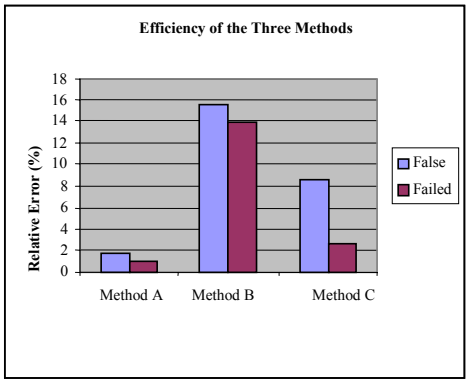

Figure 2: Comparison of the relative performance error in case of 1200 kbps and SIF image resolution.

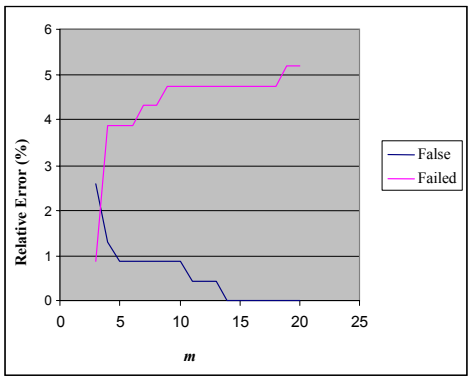

Figure 3: Relative error versus parameter *m* both in case of 1200 kbps and SIF image resolution.

A second conclusion one can make is that all methods are relatively insensitive to bitrate. We believe that the reason for this is that all methods use information stored in the dc coefficients and the MPEG encoding procedure transmits them with little probability of error.

As far as the image resolution is concerned, it can be seen a deterioration of the scene detection performance at lower resolution levels for the methods A and B. Especially, for the second case, the performance become truly unacceptable since the relative error dramatically increases. For the third method, however, there is a slightly improvement of results.

Furthermore, the effect of false alarms and failed detection for various values of parameters, *m* and *n* is presented in Figures 3-4. It can be seen that in all cases, decrease of false alarms leads to an increase of failed detection and vice versa. For this reason, as mentioned above the parameters are selected so that both errors are minimized, i.e., close to the intersection of the two plots. The optimal parameter *m*, which yields the best performance seems to be smaller than that proposed in [3] since the increase of failed detection cannot justify the milder decrease of the false ones. The same conclusions are drawn for parameter *n*.

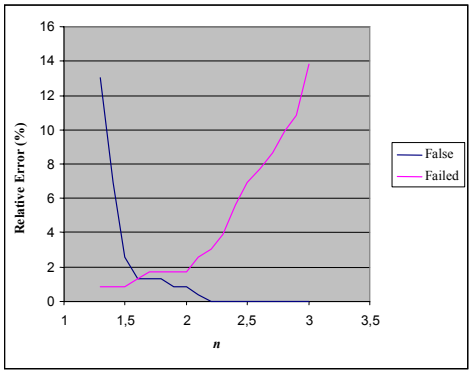

Figure 4: Relative error versus parameter *n* in case of 1200 kbps and SIF image resolution.

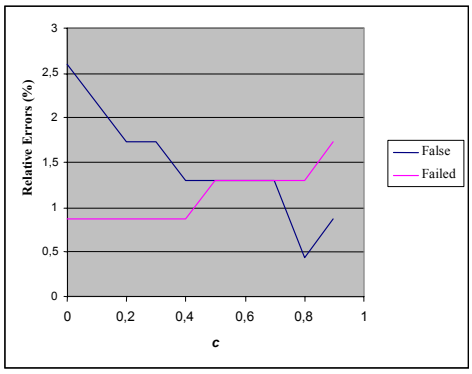

Figure 5: Relative error versus parameter *c* in case of 1200 kbps and SIF image resolution.

Finally, we conclude that taking chrominance into account indeed improves the performance. Figure 5 shows the results when performing peak detection on $d = (1-c)d_L + cd_c$, for various values of c, d_L, d_c being the difference sequences for Luminance and Chrominance, respectively.

V. REFERENCES

[1] Y. Avrithis, A. Doulamis, N. Doulamis and S. Kollias, "A Stochastic Framework for Optimal Key Frame Extraction from MPEG Video Databases," *Computer Vision and Image Understanding*, Vol. 75, No. 1/2, pp. 3-24, July/August 1999.

[2] A. Pentland, R. W. Picard and S. Sclaroff, "Photobook: Content-Based Manipulation of Image Databases," *Int. J. Comput. Vision*, Vol. 18, No. 3, pp. 233-254, 1996.

[3] Boon-Lock Yeo and Bede Liu, Fellow, *IEEE, CSVT*, "Rapid Scene Analysis on Compressed Video," Vol. 5, No. 6, Dec 1995.

[4] Y. Ariki and Y. Saito, "Extraction of TV News Articles Based on Scene Cut Detection Using DCT Clustering", *IEEE Inter. Conf,. on Image Processing (ICIP)*, Lausanne, Switzerland 1996.