A HIGH-PERFORMANCE SHOT BOUNDARY DETECTION ALGORITHM USING MULTIPLE CUES

M. R. Naphade† R. Mehrotra§ A. M. Ferman‡ J. Warnick§ T. S. Huang† A. M. Tekalp‡

†Dept. of Electrical & Computer Engineering
University of Illinois at Urbana Champaign. Urbana, IL 61801, USA
{milind,huang}@ifp.uiuc.edu
‡Dept. of Electrical Engineering & Center for Electronic Imaging Systems
University of Rochester, Rochester, NY 14627
{ferman,tekalp}@ee.rochester.edu
§Imaging Science Technology Lab
Eastman Kodak Company, Rochester, NY 14650-1816
{mehrotra,warnick}@image.kodak.com

ABSTRACT

A central step in content-based video retrieval is the temporal segmentation of video. An application independent approach to video segmentation is to detect temporally contiguous segments without significant content change between successive frames. Each such segment is termed as a shot. A high-performance shot boundary detection-based video segmentation algorithm is proposed. The technique uses unsupervised clustering on a multiple feature input space, followed by a heuristic elimination process to detect, with almost perfect accuracy, shot boundaries in the video. With an extremely high accuracy coupled with a very small number of false positives, this algorithm outperforms most of the existing techniques.

1. INTRODUCTION

With rapid increase in digital video, designing systems for efficient browsing and content based retrieval become crucial. This necessitates the creation of a Table of Contents (TOC) and an index as in the case of a book. To create the TOC and index, a central step is the meaningful temporal segmentation of video [1] - [3]. We define a shot as a contiguous video segment, without any significant content change between pairs of successive frames. The content is determined by the background, objects, their motion and their respective position(s). The focus of this paper is to integrate multiple cues for quantifying the content change and to report a high-performance algorithm for shot boundary detection. The paper proposes an algorithm that performs an unsupervised clustering over the content change descriptors in a 2-feature input space. One of the clusters, that represents the possible shot boundaries is then passed through an elimination process. A heuristic assumption in the elimination process leads to a very high accuracy and a very small number of false positives. Section 2 reviews existing techniques and their drawbacks and lays a foundation for the need of the proposed algorithm. The algorithm is presented in Section 3. Experimental results are presented in Section 4, followed by Conclusions in Section 5.

2. EXISTING METHODS

Most of the existing methods detect shot boundaries by employing metrics, that measure reliably the frame-to-frame content change. Frame pairs with a high content change are termed as shot boundaries [3],[4]. A predefined threshold for the value of these metrics is mostly used to detect shot boundaries. The most commonly used metric is the Histogram difference. Gray-scale or Color histograms of successive frames are computed and then their bin-wise difference is obtained. A linear combination or straightforward summations of these bin-wise differences is then used as a measure of the total content change across frames [3]. We term this and other variations as histogram-based methods. The histogram-based metric is robust to camera as well as object motion. A class of methods make use of the less popular pixel-wise frame difference. The intensity/color in the difference frame is then summed to produce a pixel-wise difference based metric, which is then thresholded to detect scene changes. We label all the approaches using pixel-wise differencing and its variations as spatial difference based metrics. While these metrics take into account the spatial changes, they are extremely sensitive to object and camera motion [3],[4]. Irrespective of the choice of the metric, thresholding process introduces another problem. Hard thresholds cannot perform equally well for all videos.

0-8186-8821-1/98 $10.00 © 1998 IEEE

The technique reported in [9] makes use of adaptive thresholding in the compressed domain shot boundary detection algorithm, but the thresholding is based on observations of compressed domain bit stream parameters for I, P and B frames. Our observation is that compressed domain algorithms [5],[6],[8],[9] perform well below acceptable levels of accuracy for high performance detection. Gunsel et al. [7] propose the use of unsupervised clustering to overcome this problem. However, the algorithm [7] doesn't perform with near perfect accuracy as it uses only a histogram difference based metric. The above discussion leads us to the idea of using multiple metrics to quantify content change, use unsupervised clustering for avoiding hard thresholds, and introduce an elimination process, to get rid of false positives by using a heuristic observation regarding the nature of false positives. It is our firm belief, that a single metric is not sufficient to obtain high performance.

3. APPROACH

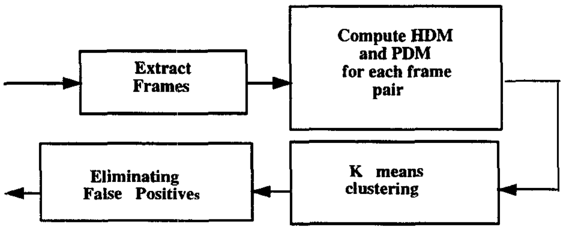
Extract Frames
Compute HDM and PDM for each frame pair
Eliminating False Positives
K means clustering

Figure 1: The key steps in the proposed Shot Boundary Detection Algorithm

A shot boundary signifies either a significant change in the overall color composition or a significant change in the object location or both. To detect this content change automatically, we propose to use both the histogram difference metric hereby termed as HDM and the spatial difference metric hereby termed as SDM. The block diagram of the proposed algorithm is as shown in Figure 1. Let f_k denote the k th frame and f_{k+1} denote the $k+1$ th frame. Let $H_k(\bullet)$ denote the 3-channel linearized histogram for the k th frame. We consider 256 uniform quantization levels for each channel. Let $D_h(f_k, f_{k+1})$ denote the histogram difference metric and $D_s(f_k, f_{k+1})$ denote the spatial frame difference for the k th frame pair. Consider the frame size to be $M \times N$. The histogram difference based metric HDM is then computed for every frame pair as

$$D_h(f_i, f_{i+1}) = \frac{1}{M \times N} \sum_{j=1}^{768} |H_i(j) - H_{i+1}(j)| \qquad (1)$$

The second metric,the spatial difference metric is defined as follows: Let $I_{i,j}(f_k)$ and $I_{i,j}(f_{k+1})$ denote the intensity of a pixel at location (i, j) in the frames f_k and f_{k+1}. Then the difference operator is defined as follows

$$d_i, j(f_k, f_{k+1}) = \begin{cases} 1 & \text{if } |I_{i,j}(f_k) - I_{i,j}(f_{k+1})| > 0 \\ 0 & \text{otherwise} \end{cases}$$

The reason to choose this variation amongst the various available spatial difference metrics, is that this metric can be insensitive to small displacements. Also, to avoid getting a noisy metric, this subtraction can be performed in a shrunken space where each channel is quantized into 32 bins. The spatial difference metric SDM is then defined as

$$D_s(f_k, f_{k+1}) = \frac{1}{M \times N} \sum_{k=1}^{M} \sum_{l=1}^{N} d_j, k(f_k, f_{k+1}) \qquad (2)$$

The approach represents every frame pair in a 2-feature input space using the features $D_h(f_i, f_{i+1})$ and $D_s(f_k, f_{k+1})$. Both features are normalized. Ideally we would expect that all the frame pairs corresponding to shot boundaries would cause collectively the two features to come out with high values. In case of real videos, this may manifest itself in large HDM values or fairly large HDM values coupled with very large SDM values. This classification can be done by avoiding the hard thresholds, using an unsupervised K means clustering algorithm. In this case the K means algorithm would actually classify all the feature inputs into 2 clusters. It is our claim that, this 2-feature, 2-means clustering algorithm clusters all the shot boundaries in the video into a single cluster. However, there are also other frame pairs in this cluster which do not correspond to shot boundaries and they are fairly large in number due to the relatively high activity in the SDM.

The elimination step is therefore introduced to process further, the frame pairs clustered as the possible frame pairs. The basis of this elimination process is a heuristic observation, that every shot boundary causes a local maximum in the HDM. This can be seen from the plot in Figure 2 which shows the HDM against the frame pair number for the video clip com01_na.mpg. This commercial has 3600 frames with each frame of size 352 × 240. It is observed, that every shot boundary corresponds to a local maximum. However, not every local maximum in the HDM plot corresponds to a shot boundary. To be detected as a shot boundary it is a necessary, but not sufficient, condition to have a local maximum in the HDM plot. The second observation is that not all local maxima may be equally pronounced. Quantitatively we can see, that if the frame pair corresponds to a shot boundary, then

$$D_h(f_i, f_{i+1}) > D_h(f_{i-1}, f_i) \qquad (3)$$

$$D_h(f_i, f_{i+1}) > D_h(f_{i+1}, f_{i+2}) \qquad (4)$$

This leads to a simple rule for determining whether the member of the possible shot boundary cluster is a

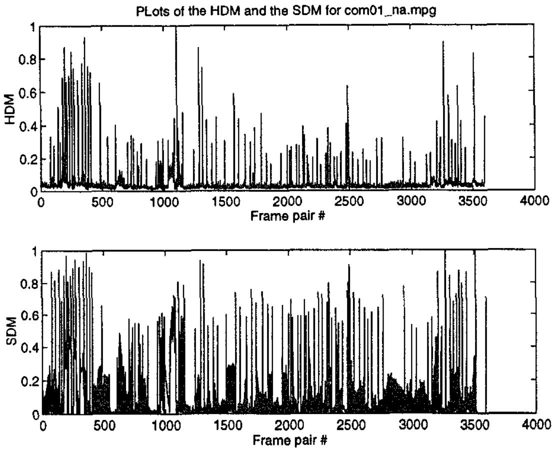

Figure 2: A plot of the 2 metrics for com01_na.mpg

false positive or not. If the possible boundary cluster member satisfies equations (3) and (4), then it is a shot boundary. This video clip has 97 shot boundaries. The algorithm detects all the 97 shots boundaries and gives 2 false positives. The size of the possible shot boundary cluster given by the clustering algorithm is 279. This is indicative of the two assumptions. One, every shot boundary is clustered in the possible shot boundary cluster. Two, the elimination based on the heuristics is accurate to a very high degree, as it eliminates 180 false positives. Figure 3 shows a plot of the 2-feature input space for the com01_na.mpg commercial.

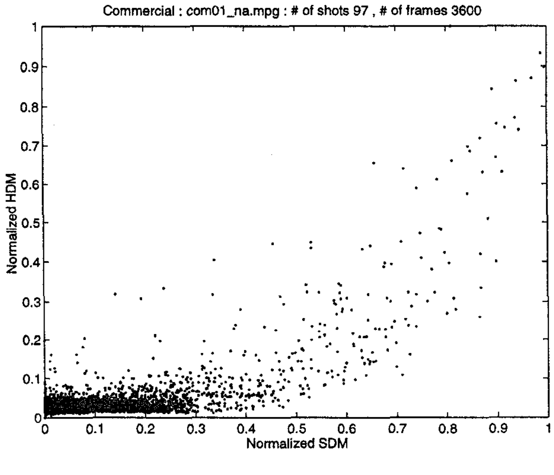

Figure 3: A plot of the 2-feature input space

4. EXPERIMENTAL RESULTS

The algorithm developed was used to carry out the segmentation on the training set videos. Four videos with 190 shot boundaries within 11241 frames. All frames are 352 × 240 in size. These video clips represent most of the difficult aspects that challenge the shot boundary detection algorithms, including rapid changes in object locations, large motion, special editing effects like zooming panning, flashing, mpeg artifacts etc. The performance analysis was done by recording the performance of our algorithm on this set and then comparing the performance of our algorithm with two other algorithms using the same set of videos. The table below shows the results of shot boundary detection of our algorithm on these videos.

Table 1: Performance of the proposed algorithm on the video sequences used for comparison against other algorithms.

Sequence	# of scenes	#of scenes detected	# of False positives
Seq_1	66	66	3
Seq_2	32	32	0
Seq_3	39	38	1
Seq_4	53	52	7

As can be seen from the above table, the algorithm detected 188 out of 190 shot boundaries and gave only 11 false positives. The above algorithm was tested against two representative algorithms, one performing unsupervised clustering on a single feature input space (without any elimination stage) [7], and the other performing an adaptive thresholding in compressed domain [9]. Individually the two algorithms have been reported to perform with good accuracy. The comparison in Table 1 indicates the superior performance of the proposed algorithm both in terms of accuracy in detecting the right shot boundaries and in terms of the false positives. False positive occurs when the algorithm declares a shot boundary when there is none. The performance comparison of the algorithm on the training set sequences is as shown in the table below.

Table 2: Comparision of performance of proposed algorithm with existing algorithms.

Algorithm	%Accuracy	%False Alarms
Proposed Algorithm	98.5	5.79
Histogram difference & Clustering	88	10.53
Compressed domain,adaptive threshold	59	71.05

From the above comparison it is evident that our algorithm outperforms most of the existing algorithms by a wide margin. The superior performance in the case of difficult videos is even more highlighted when we test our algorithm on video shots from commercials, sitcoms and movies. The performance is near perfection all the videos. The table below gives the performance figures on a set of seven such video clips. The first clip in the table is a movie, the next three are from soaps and television serials and the last three are from commercials. The commercials have rapid shot boundaries, that are closely spaced as they try to deliver a lot of content in a very limited time unlike the sitcoms. Table 3: Performance of algorithm on common videos.

Sequence	# of scenes	#of scenes detected	# of False positives
Days_01	20	29	0
TV_01	81	81	7
TV_02	23	23	0
TV_03	75	75	0
B_1	17	17	2
Gremlin	64	64	3
Pieplate	58	58	2

As can be seen the algorithm detected all the 347 shot boundaries and gave only 14 false positives which amounts to 100% detection accuracy and 4% false positives. The algorithm thus performs near perfect shot boundary detection. It is also observed that there is no deterioration in the algorithm performance when the images in the video were subsample by a factor of two and four in both the directions.

There are three important observations from these experiments. The first one indicates that the compressed domain algorithm with adaptive thresholding [9] performs poorly when presented with video with a lot of drastic content change. Its percentage of false positives is more than its accuracy. The second observation is that histogram alone cannot reach the desirable high-performance. This is evident from the comparison with the algorithm in [7]. The third observation is that the fusion of multiple cues for shot boundary detection followed by the elimination results in a very robust high performance algorithm. Video segmentation paves way for efficient browsing and further analysis. It also paves way to developing algorithms for semantic video indexing search and retrieval [10]. Compact representation of video data can be done using key frames belonging to every shot. A simple way would be to represent every shot by its first frame. A more sophisticated approach is presented in [11].

5. CONCLUSIONS

This paper proposes a new algorithm for shot boundary detection with emphasis on high performance. We demonstrate the superior performance of the algorithm and believe, that its strength lies in the choice of the features, unsupervised clustering and most importantly in the method of elimination. Near perfect performance is achieved by this algorithm for all the video clips presented to it.This paves way for further analysis of video for content based retrieval and efficient management. We intend to use the shot boundary to analyze in greater depth and isolation, video data in each shot. We also intend to define shot similarity for efficient browsing and for query by example. Also a new approach is reported for semantic video indexing and retrieval is reported in [10], which makes use of this algorithm for shot boundary detection. The high performance shot boundary detection algorithm thus plays a central role in further video analysis.

6. REFERENCES

[1] G. Davenport, et al., "Cinematic primitives for multimedia," *Computers and Graphics,* Vol. 15, 1991, pp. 67-74.

[2] S.W. Smoliar and H.J. Zhang., " Content-based video indexing and retrieval," *IEEE Multimedia,* Vol. 1 No. 2, Apr. 1994, pp. 66-72.

[3] H. Zhang., et al., " Automatic partitioning of full-motion video," *Multimedia Systems,* Vol. 1, 1993, pp. 10.

[4] A. Nagasaka and Y. Tanaka, " Automatic video indexing and full-video search for object appearances," *Proc. 2nd Working Conf. Visual Database Systems,* Budapest, Oct. 1991, pp. 119-133.

[5] B-L Yeo and B. Liu, "Rapid scene change detection on compressed video," *IEEE Transactions on Circuits and Systems for Video Technology,* Vol. 5, No. 6, Dec. 1995, pp. 533-544.

[6] N.V. Patel and I.K. Sethi, " Video segmentation for video data management," *The Handbook of Multimedia Information Management,* Eds. W.I. Grosky, R, Jain, and R. Mehrotra, Prentice Hall, PTR, 1997, pp.139 - 165.

[7] B. Gunsel, et al., "Video indexing through integration of syntactic and semantic features," *Proc. Workshop on Applications of Computer Vision,* Sarasota, Fl, Dec. 1996, pp. 90-95.

[8] H.J. Zhang, et al., "Video parsing using compressed data," *Proc. IS&T/SPIE Conf. on Image and Video Processing II,* San Jose, CA, 1994, pp. 142-149.

[9] J. Meng, et al., "Scene change detection in a MPEG compressed video sequence," *Proc. IS&T/SPIE Symposium,* Vol. 2419, Feb. 1995, San Jose, CA, pp. 1-11.

[10] M. Naphade et al., "Probabilistic Multimedia Objects (Multijects): A Novel approach to Indexing and Retrieval in Multimedia Systems," *to be presented at the International Conference on Image Processing, Oct. 98.*

[11] Y. Zhuang et al., "Key Frame Extraction using Unsupervised Clustering," *to be presented at the International Conference on Image Processing, Oct. 98.*