# Speedup Video Segmentation via Dual Shot Boundary Detection (S-DSBD)

**Prof. Adel A. Sewisy[1], Dr. Khaled F. Hussain[2], Amjad D. Suleiman[3]**

[1] *Dean, Faculty of Computer & Information Science, Assiut University, Egypt, sewisy@aun.edu.eg*

[2] *Computer Science Dep., Assiut University, Egypt, Khussain@aun.edu.eg*

[3] *Basic Sciences Department, KSAU-HS, Riyadh, KSA, suleimana@ksau-hs.edu.sa*

---***---

**Abstract -** *The high pace emergence in technologies and associated applications have triggered scientific society to explore and enable more effective and advanced mechanism so as to enable optimal decision process and computations. On the other hand, this era can be referred as the "era of data" where data and associated processing enable significant decision process. In recent years, a number of applications have come into existence that employs multimedia data processing to make certain decision process. In fact dealing with multimedia data is highly intricate and even computational complexity gets exponentially increased with video data or multi modal data types. There are a number of applications where video data plays vital role in entertainment, education, and training purposes. Even it has been employed in major surveillance, security, monitoring and control purposes. The term "video" is used extensively in the industry to represent all audiovisual recording and playback technologies. Video has been the primary concern of the movie and television industry. Over the years, that industry has developed detailed and complete procedures and techniques to index, store, edit effects and content retrieve. The techniques, however, are mostly manual in nature and are designed mainly to support human experts in creative moviemaking. They are not set up to deal with the large quantity of video materials available. To manage these video materials effectively, it is necessary to develop automated techniques to model and manage large quantities of videos.*

***Key Words*****: Digital Video, Video Segmentation, Cosine Distance, Correlation Measurement, Threshold.**

## 1. INTRODUCTION

Indexing and retrieval of digital video is an active research area in computer science. The increasing availability and use of on-line video has led to a demand for efficient and accurate automated video analysis techniques. As a basic operation on digital video, much research has focused on segmenting video by detecting the boundaries between camera shots. In videos, a shot is an unbroken sequence of frames from one camera; meanwhile a scene is defined as a collection of one or more adjoining shots that focus on an object or objects of interest. A large number of different types of boundaries can exist between shots, such as a cut, which is an abrupt transition between two shots and a fade that is a gradual change in brightness, either starting or ending with a black frame. A dissolve is similar to a fade except that it occurs between two shots (frames overlapping). Other types of shot transitions include wipes and computer generated effects such as morphing [1].The segmentation of video into scene is far more desirable than simple shot boundary detection [2]. This is because people generally visualize video as a sequence of scenes not of shots, so shots are really a phenomenon peculiar to only video. Scene boundary detection requires a high level semantic understanding of the video sequence and such an understanding must take cues from, amongst other things, the associated audio track and the encoded data stream itself. Shot boundary detection still plays a vital role in any video segmentation system. This paper focuses on a color histogram comparison, with dual methods for adjacent frames similarity testing.

## 2. RELATED WORK

A great deal of researches has been done on content analysis and segmentation of video using cut detection and gradual transitions.

- Zhang, Kankanhalli, and Smoliar used a pixel-based difference method, which is one of the easiest ways to detect if two adjacent frames are significantly different [3].
- Kasturi and Jain used a statistical methods expand on the idea of pixel differences by breaking the images into regions and comparing statistical measures of the pixels in those regions [4].
- Ueda, Miyatake, and Yoshizawa used the color histogram change rate to find shot boundaries [5].

- Nagasaka and Tanaka compared several simple statistics based on gray level and color histograms [6].
- Zhang, Kankanhalli, and Smoliar used a running histograms method to detect gradual as well as abrupt shot boundaries [2].
- Cabedo and Bhattacharjee use d the cosine measure for detecting histogram changes in successive frames and found it more accurate than other similar method [7].
- Swanberg, Shu, and Jain used grey level histogram differences in regions; weighted by how likely the region was to change in the video sequence [8].
- Little, et al used differences in the size of JPEG compressed frames to detect shot boundaries as a supplement to a manual indexing system [9].
- Zabih etal, compared the number and position of edges in successive video frames, allowing for global camera motion by aligning edges between frames [10].
- Canny suggested the replacement of Sobel filter ing with more robust methods, with the aim of defining edges more clearly, particularly in very bright or dark scenes [11].

## 3. OUR TECHNIQUE

The proposed scheme implements a technique that detect a shot boundaries by creating a 64-bin color histogram for each, the vector produced as output of this operation will be compares with its adjacent frame [1,2] as in figure 1 , and figure 2.

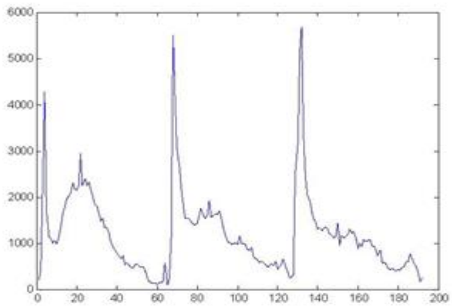


**Fig -1 : Show color histogram as example of one frame.**

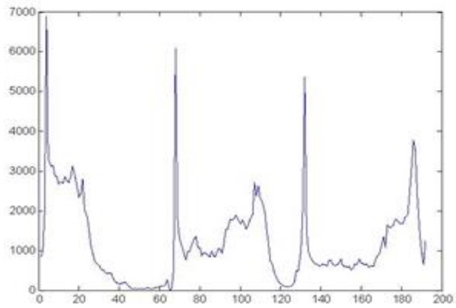


**Fig -2 : Shows color histogram for a different frame .**

We use a dual techniques to comparing the similarity of adjacent frame histograms, these techniques described as follows:

- **Cosine Distance Measurement:**

The distance Dcos (A,B) between vectors A and B is given by:

$$D_{cos}(A,B)=1-\frac{\sum\left(a_i^2\bullet b_i^2\right)}{\sqrt{\sum a_i^2}\bullet\sqrt{\sum b_i^2}} \quad \ldots\ldots (1)$$

Where ai is one bin in A and bi is the corresponding bin in B. As can been seen the cosine measure is basically the dot product of two unit vector. The result is the cosine of the angle between the two vectors subtracted from one. Therefore a small value for Dcos indicates that the frames being considered are similar, while a large Dcos value indicates dissimilarity.

The high cosine value could be indicated as a new shot or a noise in the video sequence, which may be caused by fast camera motion, flashing, computer- generated effects, or any other effects.

- **Correlation Measurement:**

The correlation R (A, B) between vectors A and B is given by:

$$R(A,B)=\frac{\sum_{i=1}^{n}((ai-\bar{A}).(bi-\bar{B}))}{\sqrt{(\sum_{i=1}^{n}(ai-\bar{A})^2\ .\ \sum_{i=1}^{n}(bi-\bar{B})^2)}}$$

Where ai is one bin in A and bi is the corresponding bin in B. also I is the mean of vector A, and F is the mean of vector B. The result is the correlation value between the two vectors. Therefore a small value for R indicates that the frames being considered are dissimilar, while a large R-value indicates similarity.

## 4. FIXED THRESHOLD

To decide whether a shot boundary has occurred, it's necessary to set a threshold, or thresholds for the similarity testing between any adjacent frames. We used fixed values for both measurements. These two values are:
Correlation value = α, since the values for correlation in -1 ≤ R ≤ 1.
Cosine Distance value = β, where β is a minimal correlation value.

If the correlation similarity value were greater than this threshold value , and cosine similarity value were less than this threshold , then a new shot boundary is detected otherwise the tested frames still in the same scene.
The companion of two fixed measurements allows the system to manipulate with different types of video data although these types need different thresholds.
Figure 3 and figure 4 show the results obtained from one tested video data, which consists of 2880 frames (1 min and 30 sec), taken from a commercial video. The peaks indicate high difference values and therefore denote shot boundaries. It can be seen that the noise are very high in both figures since commercials videos are typically have a huge number of cuts in a short time and also effected by a modern techniques generated by computers.

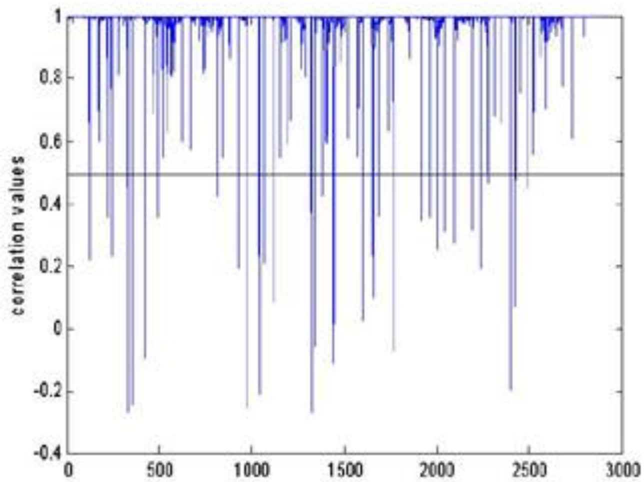


**Fig - 3 : Correlation similarity results**

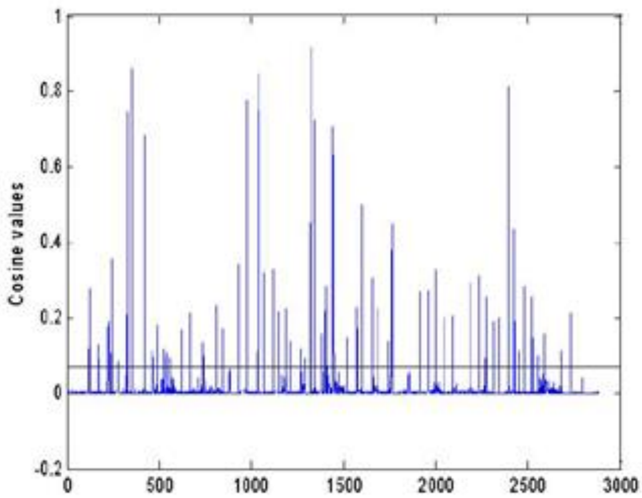


**Fig - 4 : Cosine similarity results**

## 5. REAL-TIME TESTING

To evaluate the results of our research, a datub0se of various video types was used. The video clips were digitized in MPEG format rate of 32fps (total of 345600 frames) and resolution of 320*240 pixels. Table 1 describes the test analyzed using different video types.

| Video Type | # of Frames | # of Cuts | Duration /M |
|---|---|---|---|
| News | 57600 | 371 | 30 |
| Movies | 57600 | 419 | 30 |
| Cookery Program | 48000 | 222 | 25 |
| Sports | 53760 | 210 | 28 |
| Documentary | 23040 | 66 | 12 |
| Comedy / Drama | 69120 | 210 | 36 |
| commercials | 36480 | 783 | 19 |
| Total | 345600 | 2280 | 180 |

**Table 1.** Video test set analyzed by video content type.

## 6. EXPERIMENTAL RESULT

In reporting the experimental results, I use recall and precision to evaluate system performance. Recall and precision are commonly used in the field of information retrieval. Recall is defined as the proportion of shot boundaries correctly identified by the system to the total number of shot boundaries present (correct and missed).
Precision is the proportion of correct shot boundaries identified by the system to the total number of shot boundaries identified by the system. I expressed recall and precisions as:

$$\text{Recall} = \frac{\textit{Correct Shot Changes}}{\textit{Correct+Missed Shot Change}}$$

$$\text{Precision} = \frac{\textit{Correct Shot Changes}}{\textit{Correct+False Shot Change}}$$

Ideally, both recall and precision should equal (1). This would indicate that we have identified all existing shot boundaries correctly, without identifying any false boundaries.
Now we will describe the results obtained from this experiment, the recall and precision for each video type as in table 2 and chart 1 below:

| Video Type | Recall | Precision |
|---|---|---|
| News | 93.55 | 91.8 |
| Movies | 88.71 | 91.73 |
| Cookery Program | 98.23 | 91.83 |
| Sports | 88.45 | 90.52 |
| Documentary | 95 | 100 |
| Comedy / Drama | 95.01 | 100 |
| commercials | 86.19 | 90.16 |

**Table 2.** Experiment results Recall and Precision.

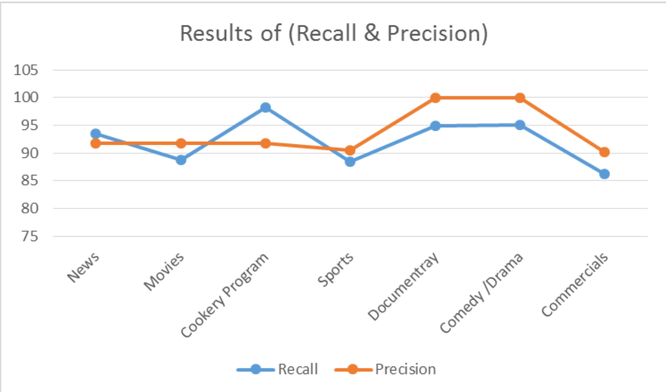


**Chart -1**: Results obtained from Experiment (Recall & Precision).

## 7. CONCLUSIONS

The high pace emergence in technologies, especially low cost video acquisition techniques and associated video processing approaches have given rise to a new era of multimedia data based computation and decision process. Video based applications have been rising with a vast pace, where key feature extraction, goal oriented event detection and further decision process are the key applications for upper end applications. In this thesis, a novel histogram analysis based video shot segmentation scheme has been developed. The proposed approach employs different enhanced video and image processing techniques for shot identification in a video data. Unlike traditional color histogram based shot detection, this thesis proposed a novel color histogram based shot segmentation scheme has been developed. At first, the video data was split into definite number of frames, then converted every frame to a histogram and deal with it as a vector. And compared them using two techniques to check the similarity of these frame histogram. This thesis has proposed a dynamic threshold based shot segmentation that enables precise shot detection in different frames even having diverse spatial nature and motion features. Here, the applied adaptive shot boundary detection algorithm changes the threshold value as per the information change per frame.

Since, the proposed work has been focused only on the shot segmentation, in future classification can also be explored and targeted object oriented shot identification and classification can be done with more enhanced classification algorithms.

## REFERENCES

[1] J. Boreczl:y and L.A. Rowe, Comparison of video shot boundary detection techniques, in IS&T/SPIE proceedings: Storage and Retrieval for Images and Video Databases IV, volume 2670, pages I 70- 179, February 1996.

[2] C O'TooJe, A. Sn4eaton, N. Murphy, and 8. Harlow, Evaluation of Automatic Shot boundary Detection on a Large Video Test Suited Challenge of Image Retrieval, Newcastle, 1959.

[3] H.J. Zhang, A. Kankarihalli and S W. Smoliar, Atltomabc partitioning of full-motion video , in Multimedia Systems, volume I , pages 10-28, 1993.

[4] R Kasturi, and R. Iain, Dynariiic Vision, in Computer Vision: Principles, R. Kasturi, and R. Jain, Editors, IEEE, Computer Society Press, Washington, 1991.

[5] H. Ueda, T. Miyatake, and S. Yosluzawa, "IMPACT: An Interachve Natural-motion-picture dedicated Multimedia Authoring System, 'M proceedings of CHI, i 991(New Orleans, Louisiana, Apr-May, 1991) ACM, New York, 1991, pp. 343-350.

[6] A. Nagasaka and Y. Taka, Automatic video indexing and full —video search for object appearances, in Visual Database SZ system II, Elsevier Science Publishers , pages 113-117, 1992.

[7] X.U Cahedo and S.K. Bhattacharjee: Shot detection tools in digital video, in Proceedings Non-linear Model Based Image Arial sis 1998. Springer Verlag, pp 12 I - 1 26, Glasgow, July 1998.

[8] Swanberg, D., Slim C.F., and Jain, R., "Knowledge Guided Parsing and Retrieval in Video Databases", in Storage and Retrieval for Image and Video Databases, Wayne Niblacl, Editor, Proc. SPIE 1908, February, 1 553, pp. 173-187.

[9] Little, T.D.C, Alianger, G , Folz, R. J., Gibbon, I F. , Reeve, F.W., ScheJ leng, D.H., and Ventkatesh, D. , "A Digital Demand Video Service Supporting Content-Based Queries" Proc. ACM Multimedia 93, Anaheim, CQ August, 1993, pp.422-436

[10] R. lebih, I Miller, and K. Mai, A feature —based algorithm for detecting and classifying scene breaks, In Proceedings ACM Multimedia 95, pages 189-200, November 1993.

[11] J. by , A computational approach to edge detection ,in IEEE Transaction on Pattern Analysis and Machine Intelligence 8(6),pages 679-698,1986.