Shot boundary detection based on Eigen coefficients and small Eigen value

Punitha P, Joemon M Jose

Department of Computing Science, University of Glasgow, Glasgow, United Kingdom

{punitha, jj}@dcs.gla.ac.uk

Abstract. Detection of shot boundaries in a video has been an active for quite a long time, till the TRECVID community almost declared it as a solved problem. A problem is assumed to be solved when there is no significant improvement being achieved from that of the state-of-the art methodologies. However, certain aspects can still be researched and improved. For instance, finding appropriate parameters instead of empirical thresholds to detect the shot boundaries is very challenging and is still being researched. In this paper, we present a fast, adaptive and non-parametric approach for detecting shot boundaries. Appearance based model is used to compute the difference between two subsequent frames. These frame distances, are then used to locate the shot boundaries. The proposed shot boundary detection algorithm uses an asymmetric region of support that automatically adapts to the shot boundaries. Experiments have been conducted to verify the effectiveness and applicability of the proposed method for adaptive shot segmentation.

Keywords: Video retrieval, eigen value, non-parametric, shot boundary detection.

1 Introduction

Proliferated production of videos has made the area of content based video retrieval and the underlying areas such as, visual content interpretation, analysis, and management, the most acclaimed and a focused area of research. In spite of the length of the video, most videos are handled in smaller chunks, either as a set of keyframes or as shorter video clips. This universal approach has made the very first step in video analysis, the shot boundary detection, an indispensable component of any video analysis and interpretation system [5, 15]. Due to the complexity and of shot boundaries, estimating the correct shot boundaries is more challenging. A shot in general, is a video segment, where the visual content of the segment remains consistent. The shot boundaries detection is difficult because of various video editing tools that result with a range of shot boundary types, e.g., abrupt, dissolve, fade in, fade out etc. Shot transitions can be of two types, abrupt and gradual. An abrupt shot transition is usually easier to detect than a gradual shot transition. Many different algorithms have been proposed in the literature [2, 3] to detect different types of transitions, and there have even been some attempts to handle all transitions at once [6]. It is very tough to detect all types of shot transitions using a single approach as the transitions are highly complex. One can still find such an attempt in [6].

Shot boundary detection algorithms are either based upon the pixel based difference between frames [14, 12, 10, 13, 8, 9, 6] or the motion difference on the temporal axis [3]. Despite the approach that is used, shot boundary detection algorithms require a prefixed threshold of difference value. The most challenging part of shot segmentation is fixing up such parameters or thresholds irrespective of the video genre and the type of shot transition, to determine a shot boundary. It is difficult to detect all types of shot transitions, at the same time with a fixed parameter. Fuzzy logic based algorithms [2] are not very sensitive to noise unlike direct threshold based approaches, but still requires threshold selection.

A very few attempts can be found in the literature to detect shot transitions adaptively. An attempt to adaptively select a global threshold to achieve a high shot cut detection rate was also made in [17]. An approach based on adaptive selection of the threshold using the average of the weighted variance of the change in previous detected shot to the current frame and the next frame was proposed in [11] and was reported to produce 94% accurate results.

In this paper, we propose an adaptive non-parametric method for detecting shot boundaries, without being specific to the type of transition. The method proposed to adaptively find the shot boundaries is inspired by a non-parametric corner point detection method [7, 1]. The distance between two subsequent frames is computed in terms of eigen distance between two frames. This vector of distances with its index can be perceived as the profile signature of an open contour. Therefore, each distance with its index forms a co-ordinate point on a two dimensional plane. The adapted corner detection method then determines the shot boundaries by computing the statistical and geometrical properties associated with the small eigen value for each co-ordinate point. For each point representing a frame, three features viz., region of support, confidence value and curvature value are computed and based on these features the true shot boundaries are located. The results indicate that the proposed algorithm is effective for various type video sequences. The various combination of the features help in detecting all types of shot boundaries.

This remaining part of this paper is organised as follows. In Section 2 we define the various video transitions. Section 3, presents an overview of the corner detection method presented in [1, 7]. Section 4 presents the proposed shot segmentation methodology adapted on corner detection mechanism. In Section 5, the performance metrics used to evaluate the proposed method is presented. Section 5, briefs about the experimental results and the paper concludes in Section 6.

2 Video transitions

A video V, in domain 2D+t, can be seen as a sequence of frames f_t, and can be described by $V = (f_t)_{t \in [0,T-1]}$, where T is the number of frames in the video. The way in which any two video shots are joined together is called the *transition* [9].
The most common transition is the *cut*, in which there is a sudden and abrupt change between two frames. *Fade* is characterised by a progressive darkening of a shot until the last frame becomes completely black. The next most common transition is the *cross fade* (*mix* or *dissolve*), where one shot gradually fades into the next. Fades have a slower, more relaxed feel than a cut. Flash is an increase of the luminosity in a few frames, which is common in television journal videos. Other advanced transitions include wipes and digital effects, which are complex changes, whilst leading into the next shot, such as, colour replacement, animated effects, pixelization, focus drops, lighting effects, a pan from one person to another, or a zoom from a mid-shot to a close-up etc. In the next section, we propose and present an algorithm that can adaptively detect all shot boundaries, irrespective of the types of shot transitions.

3 The Proposed Shot Segmentation Algorithm

This section describes the proposed feature extraction, frame difference computation and the algorithm to detect shot boundaries caused due to cut, flash, wipe and cross fade (mix, dissolve).
The problem of shot boundary detection can be related to that of corner point detection in objects. This is due to the following underlying similarities, (i) Corner points on a shape curve are effective primitives for shape representation and analysis. Similarly, shot boundaries in videos are effective primitives for video representation. (ii) Corner points on a digital boundary are found at locations where the nature of the boundary changes abruptly and significantly. On similar lines, shot boundaries in a video are also lying at locations, where the frames change abruptly and significantly. However, this definition only makes sense when the video or object is viewed from a global perspective and not at the local frame or point level.

3.1 Feature extraction and frame difference computation

As pixel based difference analysis remains as one of the most preferred approaches to find the dissimilarity between the frames, we also use pixel based frame difference to locate the shot boundary. As evident from, PCA is a linear method for data feature extraction. It is a mathematical technique used to analyse correlated random variables to reduce the dimensionality of a data set. This reduction is achieved by selecting the first few principal components. These components capture the most relevant features to use in classifying a group of objects to be recognised. Given the i^{th} frame as the current frame, its previous frame is considered as the reference frame. Both frames are normalised for their intensity values in the RGB plane and an average intensity frame is obtained. Two reflective frames lying on opposite sides of the average frame are found. The eigen vectors and the eigen values of the two reflective frames are then

computed and are used to calculate the representative eigen co-efficients of the two frames.
Let $F = [f_1, f_2]$ be the matrix representing the two frames under process and $\bar{F}$ be the average of normalised F. The average matrix $\bar{F}$ is obtained by replicating the columns with average vector to simplify the matrix subtraction.

Then, $X = F - \bar{F}$ (1)

A diagonal matrix D of eigen values and a full matrix V whose columns are the corresponding eigen vectors such that,
XV = VD are computed.
Using, X, V and D, the eigen coefficient matrix,

$$E_{coeff} = \frac{X'XV}{\sqrt{D}} \quad (2)$$

is computed.
E_{coeff} is a 2x2 matrix where the first row corresponds to the reference frame and the 2nd row corresponds to the current frame. The difference between these eigen coefficients of the two frames is recorded along with the current frame number as a boundary point. The process is repeated for all frames in sequence and the boundary points B, where $B = \{P_i \mid P_i = (f_i \in Z, d_i \in R),\ i = 1,2,3, \dots n-1\}$, n is the number of frames in the video, Z, R are the interger and real space. These boundary points are then used to locate the shot boundaries using the corner detection inspired non-parametric approach.

3.2 Shot Boundary Detection

In this section, we first present an overview of the corner detection method proposed in [7, 1] and then present how it is adopted for shot boundary detection.

An overview of non-parametric corner detection approach [7, 1]

In the domain of object detection, localising the true corner points using the local features. The paper presented asymmetric region of support to find true corner points. This subsection gives an excerpt from the original paper to keep the flow a smooth read.
Let $C = \{P_i \mid P_i = (x_i, y_i) \in R^2,\ i = 1,2,3, \dots n\}$ where, P_{i+1}, is a neighbour of P_i (mod n), and n be a close curve described in a clockwise direction.
Let P_i be the point of interest for which the region of support has to be determined to decide if P_i is a corner point or not. The region of support of P_i , consists of left arm L_i, right arm R_i, and the point P_i itself. That is,

$L_i = \{P_i = (x_j, y_j) \in R^2,\ j = i-1,\ i-2, \dots, i-l \mid (i-l) > 1\}$ and

$R_i = \{P_i = (x_j, y_j) \in R^2, \ j = i+1, i+2, \ldots, i+r \mid (i+r) < n\}$, where l and r denote, respectively, the sizes of left and right arms which are decided adaptively based on local properties of the contour and are not necessarily equal implying that the region of support are not necessarily symmetric.

It has been shown in [16], that for a straight line segment, the small eigen value in the continuous domain is zero, regardless of its length and orientation. Hence, the maximum sequence of points, on either side of the point P_i, for which the small eigen value associated with the covariance matrix of the sequence of points approximates zero, is selected as the points of the respective arms.

To compute the right arm, initially, $S_{i+1} = \{P_i, \ P_{i+1}\}$. Let (C_x, C_y) be the geometrical centroid of S_{i+1} and let C_{11}, C_{12}, C_{21} and C_{22} be the entries of the covariance matrix of S_{i+1}. The small eigenvalue is computed as given in equation (3),

$$\lambda_{S_{i+1}} = \frac{1}{2}\left[C_{11} + \ C_{22} - \sqrt{(C_{11} - C_{22})^2 + 4C_{12}^2}\right] \qquad (3)$$

Since, any two points forms a straight line, the small eigenvalue associated with two points is zero. The set S_{i+1} is updated by adding the next point P_{i+2}, in sequence and the small eigenvalue $\lambda_{S_{i+2}}$of $S_{i+2} = \{P_i, P_{i+1}, P_{i+2}\}$ is computed. In order to avoid recomputation of the small eigenvalue as and when the set is updated, the small eigenvalue is computed based on the previous set information as in [12]. This process of updating the set by adding the next point in sequence and computing the small eigen value associated with the updated set is repeated until $\lambda_{S_{i+r}}$ no more approximates zero. Once this condition is true, the value of r is taken as the size of the right arm and the points in S_{i+r} is the right arm R_i of point P_i. The left arm is computed in a similar manner, but with the points preceding the point P_i. Finally, the region of support of point P_i is the set $\{R_i \cup \ L_i\}$ with $l + r + 1$ points which are in sequence from $\{P_{i-l}, \ldots. P_i, \ldots P_{i+r}\}$. Since, the points P_{i-l} and P_{i+r} mark the end points of a segment, their confidence values are incremented by one.

It can be noticed that the size of the determined region of support varies from point to point depending on the local property of the curve within the vicinity of the point of interest and, thus, the proposed method determines adaptively the region of support which is not necessarily symmetric.

Once the region of support of a point is determined, the curvature at that point is then estimated as the reciprocal of the angle made at that point due to its left and right arms. Determination of region of support of all points helps in computing the size of the region of support of each point, curvature at each point and also to compute how many times (limit value) a point has been the endpoint of the region of support of other points. It is observed experimentally that the limit value, the size of the region of support and the curvature of an actual corner point are relatively larger than the respective values of its neighbors.

Non-parametric Shot Boundary Detection

Since after the computation of the frame difference, the frame number and the distance, can be visualised as a point on a open contour, we refer each point p(x, y) in the 2D plane, with x representing the frame number and y represents the eigen distance of frame x, to its next frame in the video. Hence, similar to the approach

referred and explained in previous section, for each point p, three features, (i) the region of support (ii) the curvature at p, formed by the two arms and (iii) the confidence value, as described earlier are extracted.
Now, if P_i is the point of interest for which the region of support has to be determined to decide if P_i marks a shot boundary, the region of support of P_i , left arm L_i, right arm R_i, are computed as explained above.
Once the region of support of a point is determined, the curvature at that point is estimated in three different ways and are labelled, θ_s, θ_c, and θ_G. θ_s is computed as the reciprocal of the angle made at that point due to the left and right arms, θ_c, and θ_G are computed as given by (4) and (5) respectively,

$$\theta_c = \begin{cases} 0 & if\ m_1 = m_2 \\ 90 & if\ m_1 m_2 = -1 \\ \frac{m_1 - m_2}{1-(m_1 - m_2)} {}^{\pi}/_{180} & Otherwise \end{cases} \quad (4)$$

Where, m_1 is the slope of the line joining the points $P_{i-1} and\ P_i$ and m_2 is the slope of the line joining the points $P_{i+1} and\ P_i$. θ_C, helps in detecting the cuts or abrupt change in the frames forming a clear cut shot boundary. But, is not very helpful to find shot boundaries caused due to gradual change in frames. This problem is generally caused when the subsequent frame difference is considered, as the subsequent frame difference remains small when there is only a slight change occurring between frames. At this point it seems plausible to assume that if instead of computing the difference between subsequent frames, if the difference between these frames is computed with respect to third frame, which could be a template frame/ image, it would provide more informative, helpful and discriminating differences. However, this is in fact not the case, as the selection of a template image/frame becomes a real challenge, as the resultant difference would depend completely on the template selected. Instead to overcome this problem, we use Von Mises probability density function θ_G, as used by [5] to find the distribution of angles at a point, on the contour and is given by (5),

$$\theta_G = -\log\left(\frac{e^{cos\left(\alpha - \frac{2\pi}{n}\right)}}{2\pi I_0(k)}\right) \quad (5)$$

$where,$

$$\alpha = cos^{-1}\left(\frac{\overrightarrow{P_{i-1}} \cdot \overrightarrow{P_{i+1}}}{|\overrightarrow{P_{i-1}}||\overrightarrow{P_{i+1}}|}\right) \quad (6)$$

To make the shot boundary detection more effective, instead of deciding a frame to be shot boundary frame, by just looking into the curvature corresponding to the frame, we also use the region of support and the confidence value at the point. For a frame marking the shot boundary, the three features, the confidence value, region of support and the curvature are expected to be larger than that of the neighbour frames as shown in Fig. 1-5 for first few frames of a video.

Following some experimentation it was observed that the shot cuts and flash are evident and easy to detect by curvature θ_C, although it can be ascertained with region of support and confidence value. However, when it is detection of shot boundaries due to gradual changes in the frames, the curvature alone is not sufficient. Therefore, selecting points on the boundary curve, corresponding to frames with local maximum curvature and local maximum confidence values, with a longer region of support becomes essential and effective.

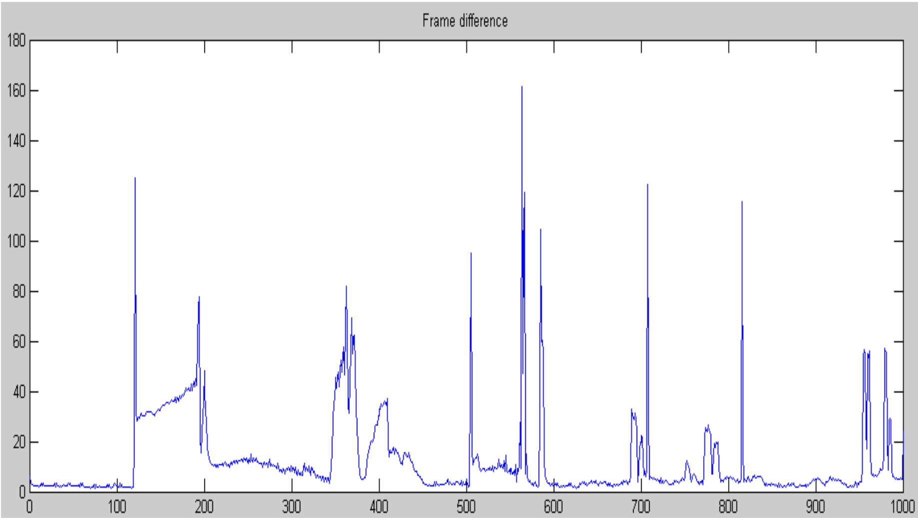

Figure 1 Frame Difference

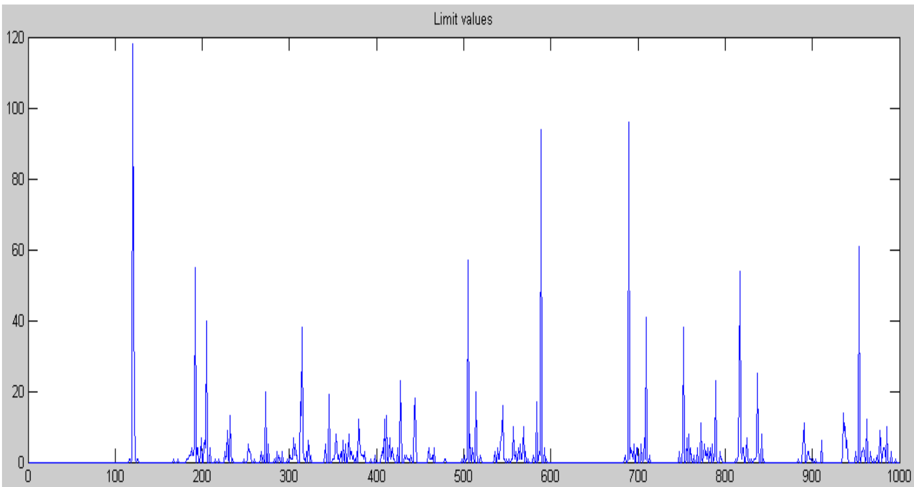

Figure 2 Confidence values for first 1000 frames of a video

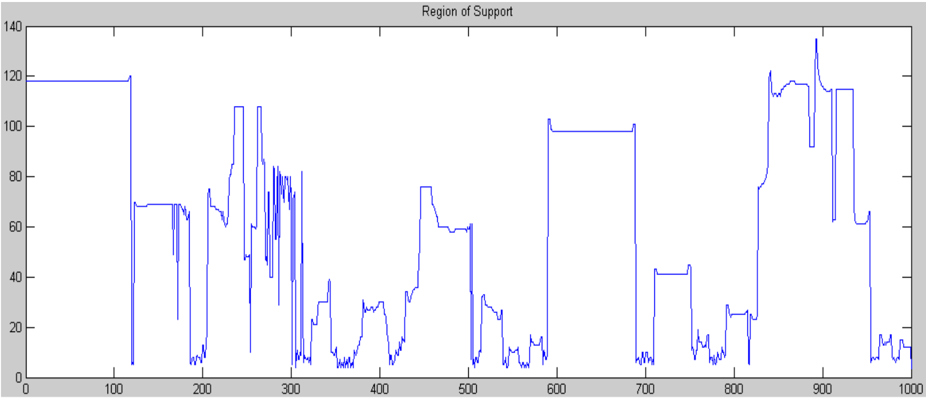

Figure 3 Region of support for first 1000 frames of a video

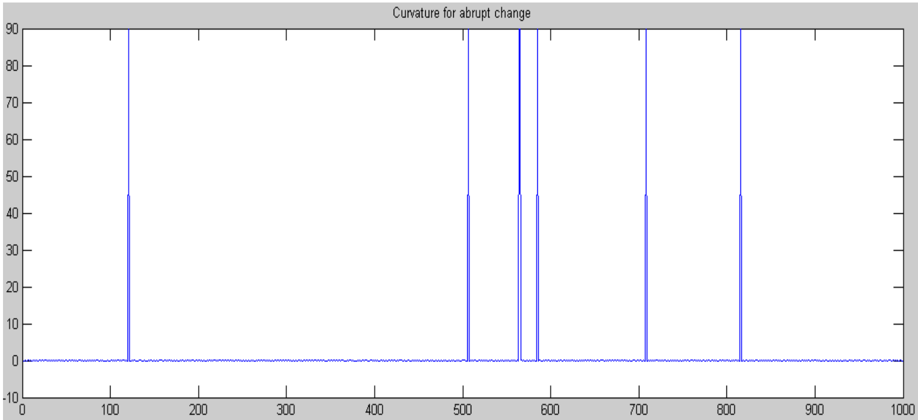

Figure 4 $\boldsymbol{\theta_C}$ - Curvature values helpful to detect shot cut

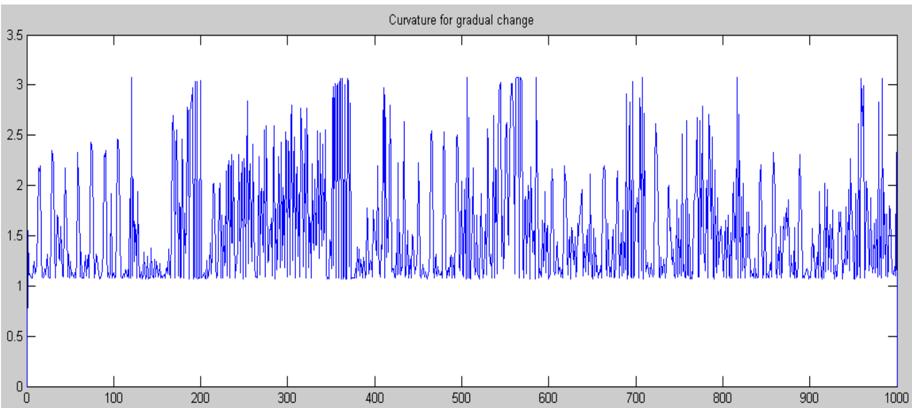

Figure 5 $\boldsymbol{\theta_G}$ - Curvature values for gradual change

4 Performance metric

To test the robustness and effectiveness of the proposed algorithm, we manually identified video shot boundaries in a number of videos, this will be used as reference for classifying the type of detection as correct, false or missed. Accordingly, #Total represents the number of shot boundaries, #Correct represents the number of events correctly detected, #False represents the number of detected shot boundaries, which do not actually represent the shot boundaries and #Missed represents the number of undetected shots. Two performance metrics, the hit rate (HR) and error rate (ER) are used to evaluate our proposed methodology. Hit rate and error rate are given in (7) and (8) below

$$HR = \frac{\#Correct}{\#Total} \quad \ldots..(7)$$

$$ER = \frac{\#Missed + \#False}{\#Total} \quad \ldots.. (8)$$

5 Experimental Results

The approach presented in this paper was applied on two genres of videos, news and commercial TV program sequences. The video sequences included cut, fade, dissolve (blur and mix), and wipes. The sequences together lasted for 3739 secs, with 112086 frames. The dataset comprised of 79 cuts, 11 flashes, 45 fades, 113 crossfades (mix, blur and dissolve), and 5 wipes.

Table 1: Shot cut detection rate

	Feature used	HR	FR
A	θ_C + confidence value	0.42	0.57
B	θ_C + confidence value	0.5	0.85
C	$\theta_G + \ confidence\, value$	0.57	0.64
D	Region of support + confidence value	0.42	0.57
E	Union of above results	1	0.42

The frame difference between the subsequent frames of a video was computed using eigen distance computation as explained in Section 3.1. The first frame index and its eigen distance to the next frame was regarded as a point on a open contour. Now the task reduces to finding the points on this contour which could be the possible shot cuts or corner points in object detection domain. For every point which represents a frame from a video, we computed the three features, (i) the region of support (ii) the curvature at p, formed by the two arms and (iii) the confidence value, as described earlier are extracted as explained in Section 3. Using these three features, for the video considered for experiments, we detected all 79 cuts with $\theta_c = 90$, with 100% accuracy, but only a very few flash, fade and dissolve shot boundaries were detected. When the common peaks were used in features, θ_c and the confidence value, we were able to find more boundaries caused due to fades. But, this also resulted with more fault detection. Using θ_G in combination to peak points of confidence, not only gave the shots undetected in previous setups, but had minimal fault rate. However, a small number of shot boundaries remained undetected. The combination of a longer region of support and the confidence values was able to detect most of the undetected shots in the above setup. But, when all three features were used in combination, many shots went undetected. Instead of using all three features together, we aggregated the results of earlier mention combination and were able to detect all shots, but at additional fault rate. The results are tabulated in Table 1.

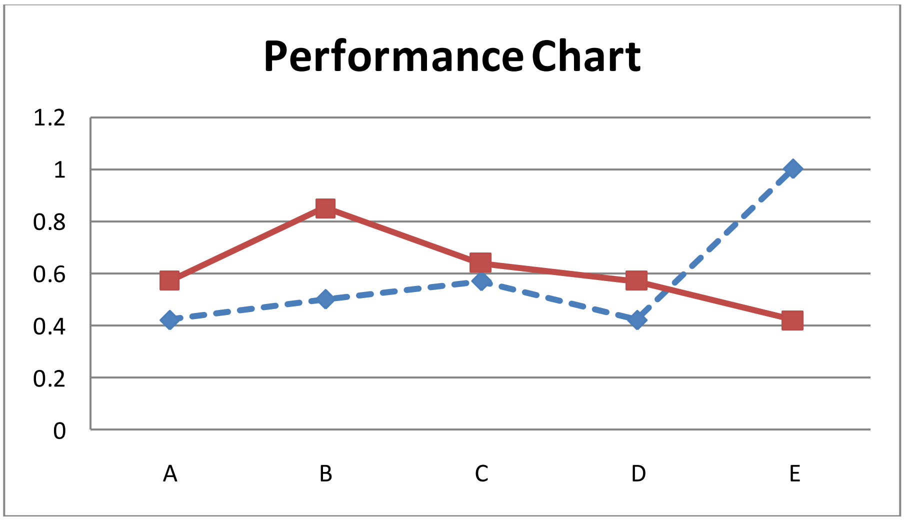

Figure 6 Hit Rate(dotted line) and Error Rate(solid line) in boundary detection for various combinations of features

6 Discussion and Conclusion

In this paper, we have made a successful attempt in exploring a model which overcomes the necessity of varying thresholds for shot boundary detection. The paper presents an adaptive and non-parametric approach for shot boundary detection, without limiting itself to a specific type of shot transition. The difference between two subsequent frames in the video is computed in terms of eigen distance between two subsequent frames of a video. The frame difference and the first frame number is regarded as the co-ordinates point on an open contour. The corner detection method is then used to determine the shot boundaries based on automatic computation of frame support, for each frame in the video using the statistical and geometrical properties associated with the small eigen value of the covariance matrix of a sequence of connected points on the open contour.
The results obtained for a synthetic video of length 4000 secs, shown in Figure 6 depicts that we achieve a high detection rate when the results of all feature listed in

the column 'Features' of Table 1 is used, however we also have a trade off with the false detection rate.

As a future work, we aim to reduce the false shot boundary detection rate by considering other heuristics. In addition, though the method was not proposed to distinguish various types of transition, the experimental results have opened up the avenues to explore possible detection of type of transition with the help of above mentioned features.

Acknowledgement

This research was supported by the EU commission, FP-027122-SALERO.

7 References

1 Dinesh R. and Guru D.S., Efficient Non-parametric corner detection: An approach based on small eigenvalue, IEEE Fourth Canadian conference on computer and robot vision, 2007.

2 Fang H, Jiang J., and Feng Y., A fuzzy logic approach for detection of video shot boundaries, Pattern Recognition 39, 2006, 2092-2100.

3 Fang H, Jiang J., Predictive based cross line for fast motion estimation in MPEG-4 videos, Proceedings of IS&T/SPIE's symposium on Electronic Imaging: Science & Technology, San Jose, USA, January 2004, 175 – 183.

4 Feldman J., Singh M.,, Information along contours and object boundaries, *Psychological Review* 112**,** 2005, 243 – 252.

5 Goyal A., Punitha P., Hopfgartner F., Jose J. M., 2009, Split and Merge Based Story Segmentation in News Videos, ECIR 766-770.

6 Guimaraes S. J. F, Couprie M., Araujo A. D. A., Leite N. J., Video segmentation based on 2D image analysis, Pattern Recognition Letters 24, 2003, 947-957.

7 Guru D. S. and Dinesh R., Non-parametric adaptive region of support useful for corner detection: a novel approach, Pattern Recognition 37, 2004, 165 – 168.

8 Hanjalic A, Shot boundary detection: unravelled and resolved? IEEE trans. Circuits Syst,. Video Technol 12(2), 2002.

9 Huang C, Liao B, A robust scene-change detection method for video segmentation, IEEE trans. Circuits Syst,. Video Technol 11(12) , 2001.

10 Jadon R.S., Chaudhury S., Biswas K.K., A fuzzy theoretic approach for video segmentation using syntactic features, Pattern Recognition Letters, 22, 2001, 1359-1369.

11 Kim W. H., Jeong Y. J., Moon K. S., and Kim J. N., 2008, Adaptive shot change detection technique for real time operation on PMP, In Proc. Of third Intnl Conf. on Convergence and Hybrid Technology, 295-298.

12 Liu T.Y., Lo K.T., Zhang X.D., Feng J., A new cut detection, algorithm with constant false alarm ratio for video segmentation, J. Vis. Commun. Image R. 15, 2004, 132-144.

13 Lo C., Wang S., Video segmentation using a histogram based fuzzy c-means clustering algorithm, Computing, Standards and Interface 23, 2001, 429-438.

14 Porter S, Mirmehdi M, Thosmas B, Temporal video segmentation and classification of edit effects, Image and Vision Computing, 21, 2003, 1098-1106.

15 Punitha P., Urruty T., Feng Y., Halvey M., Goyal A., Hannah D., Klampanos I., Stathopoulos V., Villa R., Jose J., 2008, Glasgow University at TRECVID 2008, TRECVID workshop at NIST Gaithersburg, MD, USA.

16 The C.H., Chin R.T.,, On detection of dominant points on digital curves, IEEE trans. Pattern Analysis and Machine Intelligence, 11, 1989, 859-856.

17 Whitehead, Bose P., and Laganiere R., Feature based cut detection with automatic threshold selection, LNCS Image and Video Retrieval, Volume 3115/2004, 410-418.