

# Performance Analysis of Shot Change Detection Methods

M. Banwaskar and A. Rajurkar

MGM's College of Engineering, Nanded, India  
{banwaskar\_mr@mgmcen.ac.in rajurkar\_am@mgmcen.ac.in}

**Abstract:** Shot change detection is a significant step in content based video indexing and retrieval. There are different types of transitions between the shots. Most of the shot change detection algorithms deal with these transitions separately. In this paper, we have carried out the analysis of shot change detection methods like pixel difference, histogram difference and Chi-square test and our proposed method. The proposed shot change detection method is integration of pre-processing and KLT (Kanade-Lucas-Tomasi) corner detection technique. In the pre-processing stage, adaptive local thresholding is used to eliminate non-boundary segments and only candidate segments are retained. The candidate segments are refined using bisection-based comparisons to eliminate non-boundary frames. Only refined candidate segments are preserved for further detections; hence, the speed of shot change detection is improved. KLT corner detection approach is used for obtaining key points in the frames from candidate segments. Shot change is detected if the key points between successive frames are not matching. Experimental results indicate that the proposed method is effective in terms accuracy and also helps in accelerating the shot change detection process, which can lead to better and fast retrieval of video.

**Keywords:** *Shot change detection, Corner detection, Adaptive threshold.*

## 1 Introduction

With the rapid advances in multimedia and internet technologies, video data is being uploaded at an explosive rate. It is amazing to note that thousands of new videos are being uploaded to YouTube everyday. With such huge video data resources, sophisticated video database systems are needed to enable efficient browsing, searching and retrieval. Shot change detection is a technique of detecting changing frames in video and is one of the important techniques required for efficient management of video data [1-3]. The ever growing amount of digital videos poses new challenges in Content Based Video Retrieval (CBVR) as vast repositories are being built at an increasing pace. A key step for managing a large video database is to detect the shot change. A shot is a sequence of frames generated during a continuous camera operation and represents a continuous action in time and space. Video editing procedures produce abrupt and gradual shot transitions. A hard cut is an abrupt shot change that occurs in a single frame. A gradual transition occurs over multiple frames and is the product of fade-ins, fade-outs, or dissolves.

A hard cut is instantaneous transition from one shot to the subsequent shot. Gradual Transition (GT) occurs over multiple frames, which is generated via the application of more elaborated editing effects involving several frames. Gradual transition can be further classified into fade out/in(FOI) transition, dissolve transition, wipe transition, and other transitions, according to the characteristics of the different editing effects [1-3]. Different researchers have proposed separate methods for abrupt and gradual transition detection. In our research we have developed an algorithm which can detect both abrupt and gradual transitions. The shot change detection process is the identification of considerable discontinuities in the visual-flow of the frame sequence. In this paper, we have analysed the performance of shot change detection methods such as pixel difference, histogram difference and Chi-square test and proposed an effective method for shot change detection using pre-processing and KLT corner detection.

In the first stage, pre-processing of the video is done to remove the redundant frames. The pre-processed video consisting of candidate segments is used further for KLT corner detection. Shot change is detected if the key points between successive frames are not matching. Pre-processing stage is effective in reducing the detection time.

Experiments on different types of video data show the proposed method can provide high speed and accuracy for detecting both abrupt and gradual transitions. Traditionally, video shot segmentation approaches rely on threshold method, which do not generalize for different types of video.

## 2 Related Work

In the last decade, significant work has been done in the area of video processing to partition a given video into shots. The basic idea of most of the techniques is to measure and compare the similarities between consecutive frames. Early work on shot change detection mainly considered abrupt shot transitions. A comprehensive survey of comparison and performance evaluation of existing shot change detection algorithms is given in [5-8]. In [4], shot detection techniques are reviewed and a statistical detection technique based on motion feature is proposed. Color histogram is a commonly used feature for detecting gradual transitions [9-12]. Luminance [13,14], chromaticity [15], motion [4] and edge [5] information have also been used for shot boundary detection. Boreczky et al. [16] segmented the video by using audio-visual features and Hidden Markov Models (HMM) to hypothesize the various shot transitions. The problem of shot change detection is approached by Hanjalic [17] using a probabilistic approach. For detecting abrupt transitions, adjacent frames are compared, while for gradual transitions, frames separated by the minimum shot length are compared. The a priori likelihood functions of the discontinuity metric are obtained using manually labelled data. Thus, different likelihood functions are estimated for all types of shot transition. Gradual transitions are generally more difficult to detect due to camera and object motion.

Detection of gradual transitions, such as fades and dissolves is examined in [9-11]. The approach proposed by Lienhart [18] detects dissolves with a trained classifier, operating on either YUV color histograms, magnitude of directional gradients, or edge-based contrast. The classifier detects possible dissolves at multiple temporal scales and merges the results. The classifier is trained using a dissolve synthesizer, which creates artificial dissolves from any available set of video sequences. The performance is shown to be superior when compared to simple edge-based dissolve detection methods. Cernekova et al. [19] performed Singular Value Decomposition (SVD) on the RGB color histograms of each frame to reduce the dimensionality of feature vector to ten. Initially, video segmentation is performed by comparing the angle between the feature vector of each frame and that of the average of feature vectors of the current segment. If their difference is higher than a static threshold, a new segment is started. Segments whose feature vectors exhibit large dispersion are considered to depict a gradual transition between two shots, whereas segments with small dispersion are characterized as shots. The main problem with this approach is the static threshold applied on the angle between vectors to detect a shot change, especially in the case of large intrashot content variation and small intershot content variation. A survey of core concepts underlying the different schemes of shot boundary detection is presented in [6], while a comprehensive comparison of different shot change detection algorithms is discussed in [5].

Lo and Wang [35] used clustering algorithms to monitor frame similarity. When certain frames are identified as belonging to a scene change, adjacent frames are marked as gradual transitions while the remaining frames are detected as cuts. In [36], methods for detecting shot boundaries in video sequences and for extracting key frames using metrics based on information theory are proposed. In [37], a moving query window of frames is maintained and the average frame similarities of the frames of the left side and right side of the center frame are calculated, respectively. Then the ratio of these two similarities is monitored and used to detect gradual transitions.

In all the above mentioned algorithms, each frame has to be scanned at least once for feature extraction and comparison. In the algorithms proposed in [32] and [33], each frame has to be scanned for several times in GT detection, with different frame step or transition centre in each time. Despite the excellent detection accuracies, the computation complexity is considerably high, which is the most prominent limitation in real-time applications. The problem of computation complexity in shot detection is rarely addressed. An elaborate review of shot detection algorithms on the computation complexity criterion is given in [34]. It is pointed out that this criterion is of great significance, especially in uncompressed videos that contain a huge quantity of data. In order to make a trade-off between detection accuracy and computation complexity, a fast shot detection algorithm is required. Our motivation is to propose an integrated approach for hard cut or GT detection. Thus, the fast shot detection framework is our main contribution, as it is beneficial to real-time video applications with high speed and reliable accuracy.

The rest of the paper is organized as follows: Section 3 explains the traditional shot change detection methods. Proposed shot change detection method is explained in Section 4. Experimental results and analysis is presented in Section 5. Finally in Section 6 we give the conclusion.

### 3 Shot Change Detection Methods

#### 3.1 Pixel Difference

A simple method to check whether two frames are significantly different is the direct comparison of the pixels in the consecutive frames [20-22]. If the number of different pixels is large enough, there must be shot change at these frames. The pixel-based method is easy and fast. But it is extremely sensitive to local motion, camera motion and minor changes in illumination [23,24]. The effect of motion is reduced by using 3x3 averaging filter before pixel-wise comparison. The pixel difference PD is defined as

$$PD(i) = \frac{1}{PQ} \sum_{x=1}^P \sum_{y=1}^Q |(f[x, y, i] - f[x, y, i+1])| \quad (1)$$

Pair wise pixel comparison [25,26] is a straightforward and simplest way, in which the number of pixels changed from a frame to the next is counted. When the total percentage of the pixels has changed, a shot change is detected. In this method pixel intensity of the two adjacent frames is calculated. Many color space systems, such as RGB, HSV, YIQ, HVC, or L\*u\*v, can be used for this comparison. When the frames are black and white, gray values are calculated. In the next step, difference of pixel intensities of two frames is determined and it is compared against the threshold. Shot change is detected if the difference is greater than the threshold. Rather than its simplicity, a disadvantage of this method is that it is quite sensitive to fast object movements and the camera motion - fast camera panning or zooming and to noise.

#### 3.2 Histogram Difference

Histogram difference based methods are generally used for detecting abrupt transitions. These methods use the statistics of the luminance and color. Xue L. et. al. [27] proposed a shot change detection measure in which features are obtained from the color histogram of the hue and saturation image of the video frame. Vasileios Chasanis et. al.[28] chose normalized RGB histogram as the feature vector in his shot change detection system. The advantage of the histogram-based shot change detection is that it is quite discriminant, easy to compute, and mostly insensitive to translational, rotational, and zooming camera motions. Hence it is widely used. The weakness of the histogram-based shot boundary detection is that it does not incorporate the spatial distribution information of various color, hence it fails in the case when two frames have similar histograms but different content. The histogram difference is defined by

$$HD(i) = \sum_{j=1}^G |H_i[j] - H_{i+1}[j]| \quad (2)$$

Where,  $H_i[j]$  and  $H_{i+1}[j]$  denote the histogram value for the  $i^{\text{th}}$  frame and  $(i+1)^{\text{th}}$  frame, respectively, and  $j$  is one of the  $G$  possible gray levels. The histogram difference method is less sensitive to object motion than pixel difference. The main limitation of this method is that two different frames may have similar histograms irrespective of the contents of frames. Histograms are also sensitive to light intensity changes and quantization errors.

Zhang et.al. [38] proposed a histogram method and stated that this method is a good settlement between accuracy and speed. A histogram of 64 bin gray level is computed over each image. If the difference of histograms between consecutive frames exceeds a threshold then a cut is detected. Some researchers have used DC values of Y, Cb, Cr, along with other features, to form a feature vector for each video frame for shot boundary detection purpose.

### 3.5 Chi-Square Test

Nagasaka and Tanka [29] found that when image is broken into 16 regions using Chi-square test on color histogram of those regions then better results for shot change detection are observed. Chi-square test is defined as

$$CS(i) = \sum_{j=1}^G \frac{|(H_i[j] - H_{i+1}[j])|^2}{H_{i+1}[j]} \quad (3)$$

Where,  $H_i[j]$  denotes the histogram value for the  $i^{\text{th}}$  frame and  $j$  is one of the  $G$  possible gray levels.

## 4 Proposed Shot Change Detection Method

There are many methods of shot change detection. Very few of these methods focus on detecting both abrupt and gradual transitions. In this paper we have proposed an algorithm which can detect both these transitions. The proposed shot change detection method is implemented in two steps i) Pre-processing and ii) KLT Corner Detection

### 4.1 Pre-processing

Pre-processing of video is done in two stages a) Adaptive local thresholding and b) Bisection based comparisons

*Adaptive local thresholding:*

To increase the efficiency of shot change detection algorithms, redundant frames are removed. This leads to reduction in computation burden. Hence instead of scanning the sequence frame by frame, the redundant frames are skipped without affecting the accuracy. Consecutive frames that locate in a short temporal segment within a shot usually are highly co-related. Hence, if the first and last frames of a segment show great similarities, a conclusion is drawn that it contains no boundary. As a result, this non-boundary segment is skipped. On the contrary, for the segment that spans two shots, the distance between the first and last frames is quite large. In order to filter out nonboundary segments, adaptive local thresholding technique is used in the proposed algorithm. The preserved segments are labelled as candidate segments that may contain shot changes, large motions or other effects, and further verifications are required. For non-boundary segments, the computation intensive feature extraction and comparison processes can be eliminated. In thresholding process the video sequence is first divided into segments consisting of 20 frames [30]. The reasons that we select 20 as the length for distance calculation are as follows. Firstly, it can achieve a satisfactory trade-off between detection accuracy and speed. If the length is reduced to 10, the number of distance calculations is doubled, which results in the decrease of detection speed.

Moreover, the distance calculated in GT using a small length is quite small, and therefore some GT segments may be discarded in thresholding. On the contrary, using larger length leads to higher distances in some non-boundary segments. The distance of active non-boundary segment (e.g. with motion) can be as high as those of boundary segments. In this way, boundary and non-boundary segments become non-separable. Secondly, it is observed in a large number of sequences that 20 frames is the shortest duration of shot. Therefore, a segment of 20 frames contains at most one shot boundary, which simplifies the bisection-based comparisons.

For each segment, the distance between the first and last frames is calculated, and the pixel-wise distance of the luminance component is used. The distance calculation for the  $n^{\text{th}}$  segment is as follows:

$$d(20n, 20(n+1)) = \sum_x \sum_y |F_{lum}(x, y; 20n) - F_{lum}(x, y; 20(n+1))| \quad (4)$$

Where  $F_{lum}(x, y; t)$  denotes the luminance component of the pixel in the position of  $(x, y)$  in frame  $t$ , the distance of the  $n^{\text{th}}$  segment in eq.(4) can be denoted as  $d20(n)$ . Every ten segments are then grouped together to form a basic thresholding unit, and all the segments within this unit share one threshold. The segments whose distance values are smaller than the corresponding local thresholds are classified as non-boundary segments. The local

threshold for each unit is defined adaptively according to the local and global statistics of distance values. The local thresholds ( $T_L$ ) for each unit can be computed as follows:

$$T_L = \mu_L + 0.6 \mu_G \sigma_L / \mu_L \quad (5)$$

Where  $\mu_G$  denotes the global mean value, namely the mean of all the distance values.  $\mu_L$  denotes the local mean value, namely the mean of the distance values in the thresholding unit.  $\sigma_L$  denotes the local standard deviation. The segments whose distance values satisfy the following criterion are also classified as candidate segments

$$(d_{20}(n) > 3d_{20}(n-1) \text{ or } d_{20}(n) > 3d_{20}(n+1)) \text{ and } d_{20}(n) > \mu_G \quad (6)$$

False positive is much better than false negative in thresholding. The reason is that all candidate segments have to go through further detections where the false positive ones can still be eliminated, while false negative segments that do contain shot boundaries are discarded directly as non-boundary segments. Hence, post-verification is implemented as shown in eq. (6).

*Bisection-based comparisons:*

Some non-boundary frames may still exist in candidate segments. For example if a hard cut occurs at the fifth frame of a 20-frame segment, hence last ten frames should be removed. Hence, bisection-based comparisons are performed in each candidate segment to make further eliminations of non-boundary frames. For each candidate segment, the distance values between the middle frame (i.e. the tenth frame) and the first as well as the last frames are calculated. After the pre-processing process, a large percentage of non-boundary frames are removed to reduce the scopes of hard cut and GT detections.

## 4.2 KLT Corner Detection

Corners are local image features characterised by locations where variations of intensity function  $f(x,y)$  in both X and Y directions are high. Both partial derivatives  $f_x$  and  $f_y$  are large. KLT [31] searches for points where variations in two orthogonal directions are large based on local structure matrix.

$$C_{str} = \omega_G \sigma^* \begin{bmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{bmatrix} \quad (7)$$

Derivatives of intensity function are calculated in each point. Then, the entries of the matrix  $C_{str}$  are obtained for  $D \times D$  neighbourhood of R. Each of the entries is smoothed by Gaussian filter. The diagonal entries will be the two Eigen values.

$$C_{str} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (8)$$

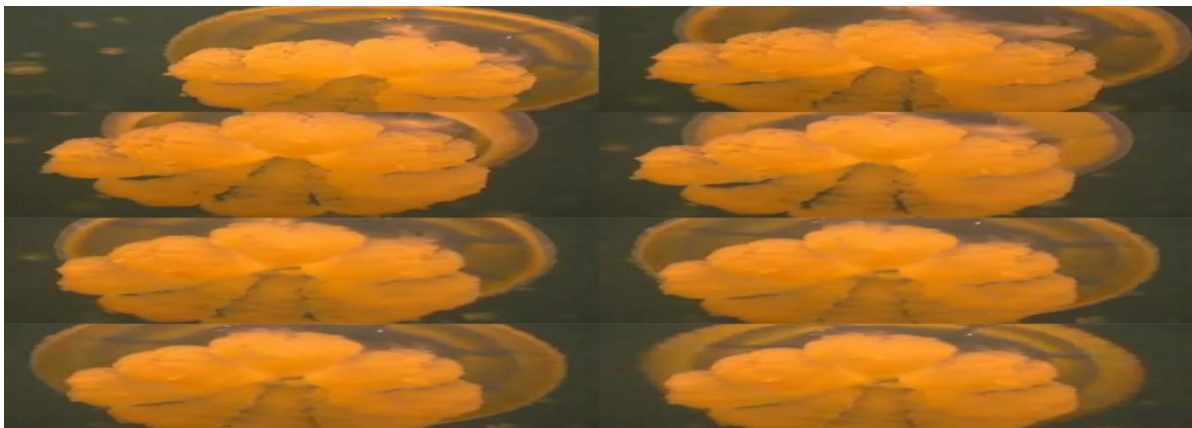
As C is symmetric and positive, both the eigen values  $\lambda_1$  and  $\lambda_2$  are non-negative, and at the location of a corner,  $\lambda_1 \gg \lambda_2 > 0$  where both eigen values are large. The KLT algorithm compares the smaller eigen value  $\lambda_2$  to a threshold value  $\lambda_{min}$  and if greater saves (x,y) in a potential corner list L. Then it sorts L in decreasing order of  $\lambda_2$  and scan the sorted list from top to bottom, selecting points in the list in sequence and removing points that fall inside the neighbourhood R of any selected points, until having the required count of features.

In our work, we have compared the adjacent frames by extracting key points in them instead of using low level features. Shot transition is detected if the key points do not match. With integration of pre-processing and this approach, we can detect both abrupt and gradual transitions. Key points are detected using KLT corner detector.

## 5 Experimental Results and Analysis

The performance analysis of shot change detection methods is done using a variety of test video clips downloaded from Youtube. In order to make a comprehensive evaluation, totally 20 test videos from six different categories, including documentary, movie, news, sports, underwater, and wild life are tested. The test set contains hard cuts and gradual transition effects that consist of dissolve, fade in and out, wipe and computer-generated effects. These 320x240 videos have 30 frames per second. The test videos were viewed frame by frame and the numbers of shots were determined manually. The number of shots obtained by pixel difference, histogram difference and Chi-square test are compared with number of shots obtained by the proposed algorithm. As given in Table 1, the number shots manually detected closely matches the number of shots obtained by the proposed algorithm. In the other methods more shots are detected hence more key frames will be extracted. Feature vector size of these key frames needed for retrieval purpose would be larger which requires more computational power and time. Hence, the proposed method is effective in CBVR.

Figure 1 shows the shot change frames detected by histogram difference method for video clip 493. Figure 2 shows the shot change frames for the same video detected by proposed method.



**Fig. 1.** Shot change frames detected by Histogram Difference method for video clip 493

**Table 1.** Number of Shots for various Shot Detection Methods

Video Name	No. Of Shots				
	Manually Detected	Pixel Difference	Histogram difference	Chi square test	Proposed method corner detection
Clip 493	13	41	12	3	16
CHI	21	38	58	8	21
Horses	14	44	17	5	12
Clip 78	9	38	3	3	9
W6	11	10	16	9	11
W3	40	37	13	9	33
W1	24	53	37	95	24
Amazing	37	146	35	50	34

It is observed that the shots are correctly identified by proposed algorithm. In pixel difference method, more false positives are observed due to fast object and camera motion. We have also analysed the performance of pixel difference and Chi-square test in YUV color space using only Y component and observed that Chi-square test gives better results than pixel difference. However, the performance of the existing shot change detection methods is poor due to disturbances caused by fast camera and object motion.



Fig. 2. Shot change frames detected by proposed method for video clip 493.

## 6 Conclusions

Detection of shot boundaries is a major challenge in CBVR systems. Videos contain abrupt as well as gradual transitions. Therefore, it is a challenging task to develop a method that can detect both these transitions. To address this challenge, we have proposed an algorithm for shot change detection using pre-processing to eliminate redundant frames followed by feature points extraction using KLT corner detector. This is important for accurate and robust detection of shot boundaries and in turn critical for high-level content based retrieval of video. We have also compared performances of our algorithm and traditional algorithms like pixel difference, histogram difference and Chi square test. It is observed that pixel difference method gives large number of shots as it takes all the frames in the video. Even if we modify the threshold, more shots are detected. Histogram difference gives comparatively less shots but not close to actual number of shots. Very less number of shots are observed with Chi square test, which loses many key frames reducing retrieval accuracy. The numbers of shots detected by our proposed method are almost same as manually detected shots for most of the videos. The variation in the number of shots is observed in videos with fast object motion. In our future work, we will improve the algorithm for the same. Also, we will carry out the analysis using more detection methods, using larger datasets and more video content types.

## References

- [1] C. Cotsaces, N. Nikolaidis, and I. Pitas, "Video shot detection and condensed representation," *IEEE Signal Processing Magazine*, Vol. 23, pp. 28-37, 2006.
- [2] S. W. Smoliar and H. J. Zhang, "Content-based video indexing and retrieval," *IEEE Multimedia*, Vol. 1, No. 2, pp. 62-72, 2006.
- [3] J. Yu and M. D. Srinath, "An efficient method for scene cut detection," *Pattern Recognition Letters*, Vol. 22, pp. 1379-1391, 2001.
- [4] H. Zhang and S. Kankanhalli, "Automatic partitioning of full-motion video," *ACM Journal of Multimedia Systems*, vol. 1, pp. 10-28, Jan. 1993.
- [5] R. Lienhart, "Comparison of automatic shot boundary detection algorithms," in *Proc. SPIE Storage and Retrieval for Image and Media Databases VII*, (San Jose, CA, USA), pp. 290-301, Jan. 26-29, 1999.
- [6] R. Lienhart, "Reliable transition detection in videos: A survey and practitioner's guide," *Int. Journal of Image and Graphics*, vol. 1, pp. 469-486, Sept. 2001.
- [7] I. Koprinska and S. Carrato, "Temporal video segmentation: A survey," *Signal Processing: Image Communication*, vol. 16, pp. 477-500, Jan. 2001.
- [8] U. Gargi, R. Kasturi, and S. H. Strayer, "Performance characterization of video-shotchange detection methods," *IEEE Trans. Circuits, Systems, Video Technology*, vol. 10, pp. 1-13, Feb. 2000.
- [9] M. Drew, Z.-N. Li, and X. Zhong, "Video dissolve and wipe detection via spatiotemporal images of chromatic histogram differences," in *Proc. IEEE Int. Conf. Image Processing*, pp. 929-932, 2000.



- [10] Y. Wang, Z. Liu, and J.-C. Huang, "Multimedia content analysis using both audio and visual clues," *IEEE Signal Processing Magazine*, vol. 17, pp. 12–36, Nov. 2000.
- [11] B. T. Truong, C. Dorai, and S. Venkatesh, "New enhancements to cut, fade and dissolve detection processes in video segmentation," in *ACM Int. Conf. on Multimedia*, pp. 219227, Nov. 2000.
- [12] Tsekeridou and I. Pitas, "Content-based video parsing and indexing based on audiovisual interaction," *IEEE Trans. Circuits, Systems, Video Technology*, vol. 11, no. 4, pp. 522–535, 2001.
- [13] B. Truong, S. Venkatesh, and C. Dorai, "Automatic genre identification for contentbased video categorization," *Proc. of Int. Conf. Pattern Recognition*, vol. 4, pp. 230–233, Sep. 2000.
- [14] L. Perez-Freire and C. Garcia-Mateo, "A multimedia approach for audio segmentation in TV broadcast news," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, (Canada), pp. 369-372, May 17-21, 2004.
- [15] J. Zhou and X.-P. Zhang, "Video shot boundary detection using independent component analysis," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, (Philadelphia, USA), pp. 541-544, Mar. 18-23, 2005.
- [16] J. S. Boreczky and L. D. Wilcox, "A hidden Markov model framework for video segmentation using audio and image features," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, (Seattle, WA), pp. 1228-1231, May 7-10, 1998.
- [17] A. Hanjalic, "Shot-boundary detection: Unraveled and resolved?," *IEEE Trans. Circuits, Systems, Video Technology*, vol. 12, pp. 90–104, Feb. 2002.
- [18] R. Lienhart, "Reliable dissolve detection," *SPIE*, vol. 4315, pp. 219–230, Jan. 2001.
- [19] Z. Cernekova, C. Kotropoulos, and I. Pitas, "Video shot segmentation using singular value decomposition," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, (Hong Kong), pp. 181-184, Apr. 6-10, 2003
- [20] A. Nagasnlm, Y.Tanaka. "Automatic video indexing and full-motion search for object appearance". In *Proc. IFIP 2nd Working Conference on Visual Databases Systems*, Budapest, Hungary, 30(3), pp. 113-127, September 1991
- [21] H. Zhang, A. Kankanhalli, and S.W. Smoliar. "Automatic partitioning of full-motion video". *Journal of Multimedia Systems on ACM*, 1(1), pp. 10--28, 1993
- [22] S. Choubey and V.~V. Raghavan. "Generic and fully automatic content-based image retrieval using color". *Journal of Pattern Recog*, 18(11-13), pp. 1233--1240, 1997
- [23] C. Cotsaces, Nikolaidis, and I. Pitas. "Video Shot Detection and Condensed Representation". *Journal of IEEE Signal Processing Magazine*, March, pp. 28--37, 2006.
- [24] J. H. Yuan, H. Y. Wang, and B. Zhang. "A formal study of shot boundary detection". *Journal of Transactions on Circuits and Systems for Video Technology*, 17(2), pp. 168--186, February 2007.
- [25] C. Cotsaces, Nikolaidis, and I. Pitas. "Video Shot Detection and Condensed Representation". *Journal of IEEE Signal Processing Magazine*, March, pp. 28--37, 2006.
- [26] J. H. Yuan, H. Y. Wang, and B. Zhang. "A formal study of shot boundary detection". *Journal of Transactions on Circuits and Systems for Video Technology*, 17(2), pp. 168--186, February 2007.
- [27] L. Xue, C. Li, H. Li, and Z. Xiong. "A general method for shot boundary detection". In *Proceedings of the 2008 International Conference on Multimedia and Ubiquitous Engineering*, PP.394-397, 2008.
- [28] V. Chasanis, A. Likas, and N. Galatsanos. "Simultaneous detection of abrupt cuts and dissolves in videos using support vector machines". *Journal of LSEVIER Pattern Recognition Letters*, 30, pp.55-65, 2009
- [29] A. Nagaska, and Y. Tanka, "Automatic video indexing and full video search for object appearance", *Visual Database systems II*, Elsevier Science Publishers, pp 113-27,1992
- [30] Y.N. Li, Z.-M. Lu, X.-M Niu, "Fast video shot bpundary detection framework employing pre-processing techniques", *IET Image Process*, 2009, vol 3, Iss 3, pp 121-134.
- [31] J. Majumdar, G. M. Venkatesh, Amith Kamath, " Video Shot detection using Corner detectors and Optical flow", Elsevier Publications 2013.
- [32] Bescos J. 'A unified model for techniques on video-shot transition detection', *IEEE Trans. Multimed.*, 2005, 7, (2), pp. 293–307
- [33] Grana C., Cucchiara R.: 'Linear transition detection as a unified shot detection approach', *IEEE Trans. Circuits Syst. Video Technol.*, 2007, 17, (4), pp. 483–489
- [34] Bastien L., Je 'Ro.: 'A review of real-time segmentation of uncompressed video sequences for content-based search and retrieval', *Real-Time Imaging*, 2003, 9, (1), pp. 73–98
- [35] C. Lo and S. Wang, Video segmentation using a histogram-based fuzzy C-Means clustering algorithm, in *Journal of Computer Standards and Interfaces*, 23:429-438, 2001.



- [36] Z. Černeková, I. Pitas, S. Member and C. Member, Information theory based shot cut/fade detection and video summarization, in IEEE Transactions on Circuits and Systems for Video Technology, volume 16, pages 82-91, 2006.
- [37] T. Volkmer, S. Tahaghoghi and H. Williams, Gradual Transition Detection Using Average Frame Similarity, in Proceedings of Multimedia Data and Document Engineering 2004, Washington, DC, USA, July 2, 2004.
- [38] H. J. Zhang, "Content-based video browsing and retrieval" in handbook of Internet and multimedia systems and applications, B. Furht, Ed.: CRC press LLC, 1999.