

A Comparison of Corner Feature Detectors for Video Abrupt Shot Detection

Matheel E. Abdulmunem¹ and Eman Hato^{2*}

¹Department of Computer Science, University of Technology, Bagdad-Iraq.

²Department of Computer Science, University of Mustansiriyah, Bagdad-Iraq.

*Corresponding Author: emanhato@yahoo.com

Abstract

Comparison of feature detectors and evaluation of their performance is very important in computer vision. A new algorithm is proposed in this paper to compare the performance of four corner feature detectors based on abrupt shot boundary detection. The proposed algorithm consists of two stages: feature vectors generation where corner detector for all video frames is computed to obtain the descriptor feature vectors, and features matching where the number of matching features between two successive frames is calculated. The corner feature detectors used in this paper are BRISK, Harries, MinEigen, and FAST. Experimental results indicate that the proposed algorithm using MinEigen features detector provides better performance than other features detectors where the average value of recall, precision, and F measure is 0.99083, 0.98808, and 0.98875 for selected testing videos respectively. The results also show that the FAST is superior to others feature detectors when considering execution time. [DOI: [10.22401/JNUS.21.3.20](https://doi.org/10.22401/JNUS.21.3.20)]

Keywords: Abrupt dedication, BRISK, FAST, Harries, MinEigen.

Introduction

The video shot detection techniques has become one of the most important research areas in content based video analysis and retrieval. The aim of shot detection techniques is to divide the video into essential and meaningful parts called shots which is often used as basic meaningful unit for analyzing, summarization and indexing video. Each video shot represents a continuous series of actions, which corresponds to a series of frames captured by single camera without significant changes in visual content [1]. The basic principle of shot detection algorithms is to find the boundary between the shots using video content features, where the boundary is the discontinuity frame(s) that describes the transition from one shot to the subsequent shot [2]. Shot boundary fall into two classes [3]: Abrupt Transition (AT) and Gradual Transition (GT). The shots are separated by a single frame called AT, while the shots are separated by several frames having highly interrelated visual information called GT. A GT is obtained using some effects, for example fade in where a new shot appear gradually with an increase in the brightness from a black color frame, fade out is reverse of a fade in, and dissolve where two consecutive shots overlap [4].

However, it is quite difficult to detect a video shot transition due to many reasons such that object motions, camera movements and noise addition which often may change the video content dramatically and cause a false detection on shot transition [5]. Researchers have attempted to detect the abrupt shot boundaries by focusing on visual interruptions between frames that can be detected by searching for two successive frames that are totally uncorrelated. Various useful features extracted from the video frames are used to distinguish between transition and non-transition frames by applying similarity measures that are then compared to threshold [6].

The image features (frame features) can be categorized into two classes: the features represent the content of the whole image by only single multidimensional feature vector called global features while the features detect and describe interest regions in an image by set of several feature vector called local features [7] as shown in Fig.(1). The extraction of local features consists of two independent stages: feature detection and feature description. The aim of a feature detector is to detect a set of interest regions (also called keypoint), while the aim of features descriptor is to extract stable features for information around the detected keypoint or the determined

regions mathematically [8]. Although global features are fast, simple in computation and require small amounts of memory, but they are not invariant for significant motion, mix information from both parts (not discriminate the background from foreground of the image), and sensitive to clutter and occlusion. Unlike global features, local features are more robust for variations in scale, rotation, illumination and noise, they are more useful for matching images and recognizing objects, but they require a large amount of memory [7, 8].

Various local and global features extracted from video frames are employed in abrupt shot boundary detection techniques. For example in [9] abrupt transition is detected by computing the number of the matched keypoints of Scale Invariant Feature Transformation (SIFT) value between adjacent frames that must be lower than predefine threshold. Abrupt shot boundary detection approach based on the Weber local Descriptor (WLD) and adaptive thresholds is proposed in [10]. In [11] Hue Saturation Value (HSV) color histogram and Histogram of Gradient (HOG) features are exploited to detect AT. HSV color histogram is used for introductory detection of video shot while HOG feature is adopted for secondary detection to improve the algorithm performance. Chi-square is used as a similarity measure which is then compared to the global adaptive threshold. In [12] authors employed absolute sum gradient orientation feature difference which is compared to threshold generated from local and global threshold for AT detection. The authors in [13] used the

features extracted by Gray Level Co-occurrences Matrix (GLCM) and correlation measure to calculate the difference between two consecutive GLCM video frames. A shot detection method presented in [14] combines local and global features by using distance of SIFT Point Distribution Histogram (SIFT-PDH) of consecutive frames and an adaptive threshold for detection the cut and gradual transition. Speeded Up Robust Features (SURF) features and entropy are employed in [15] for scene detection. Entropy analysis is used to eliminate the frames which contain gradual effects and blank screens. Candidate abrupt boundaries are detect by analyzing the abrupt changes in consecutive frames after that SURF features matching is used to refine the candidate abrupt boundaries. The AT is detected in [16] by Support Vector Machine (SVM) with scores generated from consecutive frames similarity measurements. The scoring of frame boundaries is calculated with the combined matching distances of the HSV histograms as global features and SURF as local features. After the scores of frame boundaries are performed, the SVM works on classifying the cut transitions and non-cut transitions. The objective of this paper is to make a comparison between four of the corner feature detectors based on their performance to detect abrupt video shot boundaries.



Fig.(1): Image feature representation.

The remainder of this paper is structured as follows: The following section discusses the description of features extraction techniques. The proposed shot boundary detection algorithm is presented in section 3. Section 4 illustrates the obtained experimental results. Lastly, the conclusions are contained in section 5.

Local Feature Extraction

Local features refer to a distinct structure or pattern found in an image, such as blobs, corners, and edge pixels. The following section briefly describes the types of the corner feature detectors techniques used in this paper.

1.Binary Robust Invariant Scalable Keypoints (BRISK)

BRISK is a binary detectors and descriptors computed directly on image patches. First the keypoints of interest regions are determined across both the image and scale dimensions. Next the sampling pattern consisting of points lying on scaled concentric circles is applied at the neighborhood of each keypoint to processing local intensity gradients. Finally, the oriented BRISK sampling pattern is used to obtain pairwise brightness comparison results that are assembled into the binary BRISK descriptor. BRISK used for hard real time constraints such as mobile wireless devices that have limited bandwidth channel and low power requirements [17].

2. Harries

Harris combines corner and edge detector using local self-correlation function that measures the local changes of the image with patches shifted by a small amount in different directions. The Harris algorithm first calculates the difference value of each point in the grayscale image. The self-correlation matrix is then obtained by applying convolution to the difference values using the Gaussian Function. Self-correlation matrix is a real symmetrical matrix, so it has two eigenvalues that fall into the three cases: (1) when the two eigenvalues are both larger positive values, then the pixel point is considered as a corner, (2) when an eigenvalue

is large and the other is small, then the pixel point is located in the edge line, and (3) when both eigenvalues are small, then the pixel point is located within a smooth region. The Harris detector is easy to compute and insensitive to image rotation, grayscale gradation and noise effect but sensitive to changes in scale [18].

3.Minimum Eigenvalue Algorithm (Min Eigen)

In minimum eigenvalue algorithm (known as Kanade-Lucas-Tomasi (KLT)), corner detection is based on the calculation of the cornerness response of each pixel by measuring the change in intensity due to local integration window shifts in all directions giving peaks in cornerness response to the corner pixels. MinEigen is similar to Harris in detecting corner points; differ only in the way the cornerness functions are evaluated. The Harris corner detector evaluates the cornerness of each pixel without clear decomposition of eigenvalue, while the MinEigen explicitly calculates the eigenvalues and chooses those points whose minimum eigenvalue is greater than a given threshold. The MinEigen detector is simple in computation and insensitive to image variation [19].

4. Features from Accelerated Segment Test (FAST)

FAST corner detector detects the candidate points by performing a segment test to each pixel in the image using a circle of 16 pixels (bresenham circle) around the corner candidate pixel as introductory calculation. If a set of neighboring pixels (n) in the bresenham circle with a radius (r) are all brighter than the candidate pixel intensity plus a threshold value, or all darker than the candidate pixel intensity minus the threshold value then candidate pixel is considered as a corner. To make the algorithm fast the value of (n) value is determined to be twelve to exclude a very large number of non-corners. Fast detector has a high speed computation but suffers from several limitations like the algorithm does not work well if (n) value less than twelve and the selecting and arranging a fast test pixels have implicit assumptions therefore a machine learning approach has been used to improve and address these constraints [20].

Proposed Video Shot Detection

In abrupt shot the last frame of a shot is very different from the first frame of the next shot, so the similarity measure between two consecutive shots is always very low as shown in Fig.(2). The proposed algorithm is composed of two main stages: feature vectors generation, features matching.

1. Feature Vectors Generation

First, the frame images are extracted from the input video, and then all frames are resized to 256x256 and converted into grayscale images. After that, corner detector and descriptor is computed for all gray scale frames to obtain the descriptor feature vectors. Four type of corner detectors are used in this paper which are BRISK, Harries, MinEigen, FAST. Corner detector and descriptor algorithm works in two steps. First step is to detect corner points and the second step is to describe the detected corner points in a vector format (feature vectors). Here 64 is the length of the feature vector of each corner point detected. Fig.(3) shows an example of applying the features detectors for a selected image from video test file (View2).

2. Features Matching

Basically, after the features and their descriptors extracted from all frames, the next step is to perform feature matches between two consecutive frames. The purpose of features matching is to determine the best correspondence features in adjacent frame from the set of descriptors (features vectors) using distance function. The features matching specify how nearest neighbors between features vectors in two successive frames are found. Two feature vectors match when the distance between them is less than the predefined threshold parameter and rejects otherwise. The number of matching features can be corresponding as the similarity value between two successive frames. A predetermined global threshold (Th) is defined for each type of the corner detectors. If similarity value of frame satisfies the condition in equation (1), the abrupt boundary is recorded.

$$FS_i = \begin{cases} true, & (FS_i < Th) \& (FS_{i+1} > Th) \& (FS_{i-1} > Th) \\ false, & otherwise \end{cases} \dots\dots\dots (1)$$

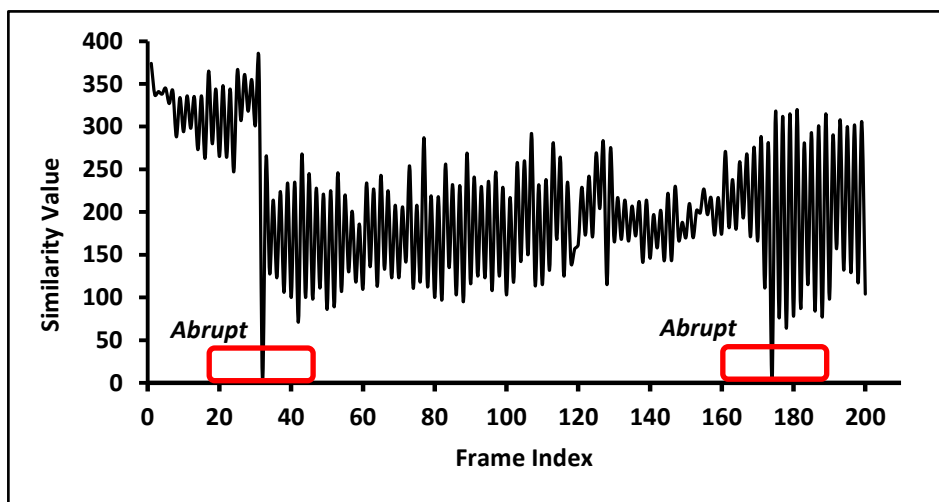


Fig.(2): Multiple abrupt transitions for track of video test file (View1).

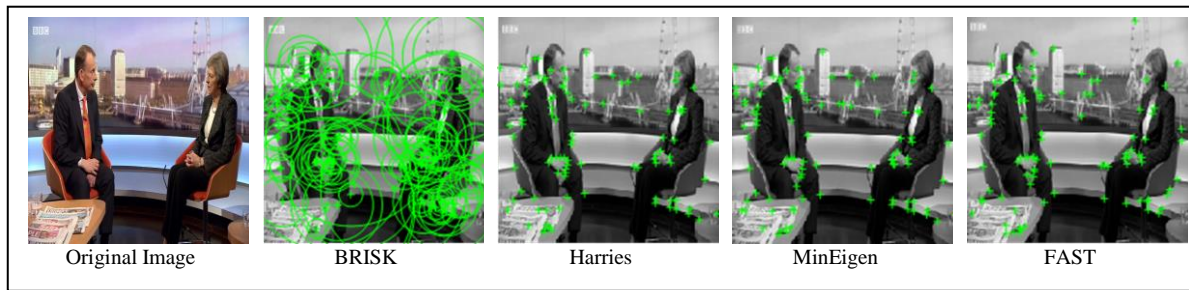


Fig.(3): Corner detection for a selected image from video test file (View2).

Where the FS_i is similarity value of current frame, FS_{i+1} is similarity value preceding current similarity value FS_i , while FS_{i-1} is similarity value previous current similarity value FS_i . The procedure of the abrupt shot detection process is illustrated in algorithm (1), and an example of abrupt transition detection is illustrated in the Fig.(4).

Experimental Results

Evaluations tests are performed on the proposed algorithm using twelve video files that have been selected from the BBC archive as test material; Table (1) shows the details of all selected video files. The measures used to compare the corner detector performance employed in proposed algorithm are Recall (R), Precision (P) and F measure (F) as defined in the following equation [12,15]:

$$Recall(R) = \frac{True}{True+Miss} \dots\dots\dots (2)$$

$$Precision(P) = \frac{True}{True+False} \dots\dots\dots (3)$$

$$F\ measure(F) = \frac{2*Precision*Recall}{Precision+Recall} \dots\dots (4)$$

Where True, Miss, False is the number of correct, missed and false transitions detected respectively. High precision and high recall indicate perfect shot transition detection.

In order to obtain optimal thresholds for each corner detector used in the proposed algorithm, an experiment is performed by changing thresholds values until the best overall performance is achieved.

Algorithm (1): Abrupt transition detection
Input: Video file, Threshold value (Th).
Output: Abrupt transition frames index.
Begin Vid←upload video and extract all frames NumFram ← get number of frames (Vid) j←1 //counter for abrupt transition frames. For i = 1 : NumFram Im←resize and convert to grayscale (Vid (i)) corners ← Detect Corner Features(Im) features(i)←Extract Features(Im,corners) End For For i=1: NumFram -1 KeyMach ← Match Features (features (i), features (i+1)) NoKey←size (KeyMach) // NoKey: number of matching features If (Output of equation (1) == true) Abrupt(j) ←i j←j+1 End If End For End

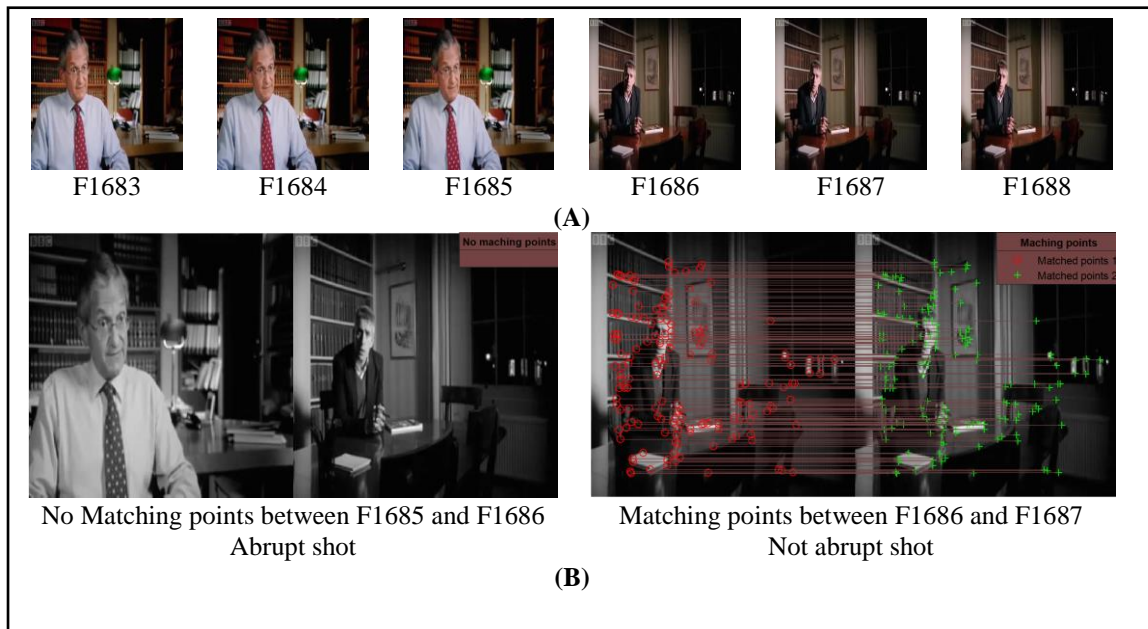


Fig.(4): Abrupt transition for the track of video test file (Doc3)
(A) Successive frames with abrupt transition (B) Abrupt transition detection process.

Table (1)
Specifications of selected video files.

Video	Number of Frames	Transition Abrupt	Source (BBC archive)
Doc1	1023	26	Documentary video
Doc2	6189	55	
Doc3	13517	96	
Doc4	20100	151	
Eng1	14578	45	English learning news review video
Eng2	12453	68	
Eng3	12026	72	
Eng4	14505	76	
View1	32264	73	Interview video
View2	23299	61	
View3	8775	43	
View4	7670	31	

Thresholds are set to $Th=1$ when the proposed algorithm use BRISK, Harries, FAST, and $Th=5$ when the proposed algorithm use MinEigen. For simplicity, we will refer to proposed algorithm when using BRISK, Harries, MinEigen and FAST, as PBRISK, PHarries, PMinEigen and PFAST respectively. The performance of PBRISK, PHarries, PMinEigen and PFAST according to Recall and Precision measure is presented in Table (2), while Table (3) presents their performance according to F measure.

The results in reference to Table (2) and (3) obviously demonstrate that the PMinEigen has high performance compared to the PBRISK, PHarries and PFAST for all video categories. Object/camera motions and noise addition in variety classes of video may dramatically change video content and cause a false detection on shot transition. One can conclude that the PMinEigen handles the object and camera motion very well with a very less false detection. Thus the PMinEigen has some robustness for object/camera motion and illumination changes where the false or miss detections are mainly due to the fast object movement as shown in Fig.(5(A)).

On the other hand the performance of BRISK and Harries is relatively low compared to the PMinEigen where the false or miss detections of abrupt transition is due to the minor object motion or the color change is not clear as shown in Fig.(5(B) and (C)). The

substantial sensitivity of PFAST can be observed from the low values of recall and precision to any change in the video content. Fig.(5(D)) shows an example of false abrupt detection due to the slight object movement. A graphical representation summarizes the overall performance of the proposed algorithm is shown in Fig.(6), It is evident that the PMinEigen achieved the highest average recall, precision and F values whereas the PFAST gave the lowest one. Both PHarries and PPBRISK have a close performance, where PHarries is better than PPBRISK in in terms of recall, but worse than in term of precision, PHarries is superior when considering F measure. The running speed is an important aspect for an efficient shot boundary detection algorithm. The implement average time of PBRISK, PHarries, PMinEigen, PFAST is 48.06625, 14.75491, 15.50994, 12.74355 second respectively to detect abrupt shot from track of test video file (View1) when.

The code executed by MATLAB R2017a on an Intel Core i7 hp laptop 5500U clocked at 4.40 GHz. It is noticeable from time calculating that PFAST has lowest time comparative to other algorithms as express in Fig.(7).

Table (2)
Proposed algorithm performance according to Recall and Precision.

Video	PBRISK		PHarries		PMinEigen		PFAST	
	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision
Doc1	0.885	1	0.923	1	1	1	0.846	0.917
Doc2	0.855	0.959	0.909	0.981	1	0.965	0.818	0.918
Doc3	0.948	0.929	0.927	0.918	0.989	0.913	0.854	0.812
Doc4	0.755	0.826	0.795	0.889	0.947	0.979	0.642	0.808
Eng1	0.971	1	1	1	1	1	0.985	1
Eng2	0.556	1	0.512	1	0.977	1	0.4	1
Eng3	0.75	1	0.931	1	1	1	0.917	1
Eng4	0.974	1	0.987	1	1	1	0.947	1
View1	0.877	0.901	0.836	0.782	1	1	0.822	0.208
View2	0.819	0.781	0.885	0.701	1	1	0.852	0.722
View3	0.931	1	0.791	1	0.977	1	0.931	0.976
View4	1	1	1	1	1	1	0.968	0.909
Average	0.86008	0.94967	0.87467	0.93925	0.99083	0.98808	0.83183	0.85583

Table (3)
Proposed algorithm performance according to F measure.

Video	PBRISK	PHarries	PMinEigen	PFAST
Doc1	0.939	0.959	1	0.880
Doc2	0.904	0.944	0.982	0.865
Doc3	0.938	0.922	0.944	0.832
Doc4	0.789	0.839	0.963	0.715
Eng1	0.985	1	1	0.992
Eng2	0.715	0.677	0.988	0.572
Eng3	0.857	0.964	1	0.957
Eng4	0.987	0.993	1	0.973
View1	0.888	0.808	1	0.332
View2	0.799	0.782	1	0.782
View3	0.964	0.883	0.988	0.953
View4	1	1	1	0.937
Average	0.89708	0.89758	0.98875	0.81583

Conclusions

The performance of four types of corner feature detector and descriptor is assessed in this paper based on the detection of abrupt shot boundary for different categories of video.

In order to make a comparison, the proposed algorithm first, compute descriptors features vectors for all frames of video using four type of corner detectors that are BRISK, Harries, MinEigen, FAST. A matching feature is then performed to determine the best correspondence features between two consecutive frames using the distance

function. A number of matching features are corresponded as a similarity value that compares with a predefined global threshold used to detect the abrupt shot.



Fig.(5): Example of wrong abrupt detection.
 (A) False detection in PMinEigen. (B) Missing detection in PHarries.
 (C) False detection in PBRISK. (D) False detection in PFAST.

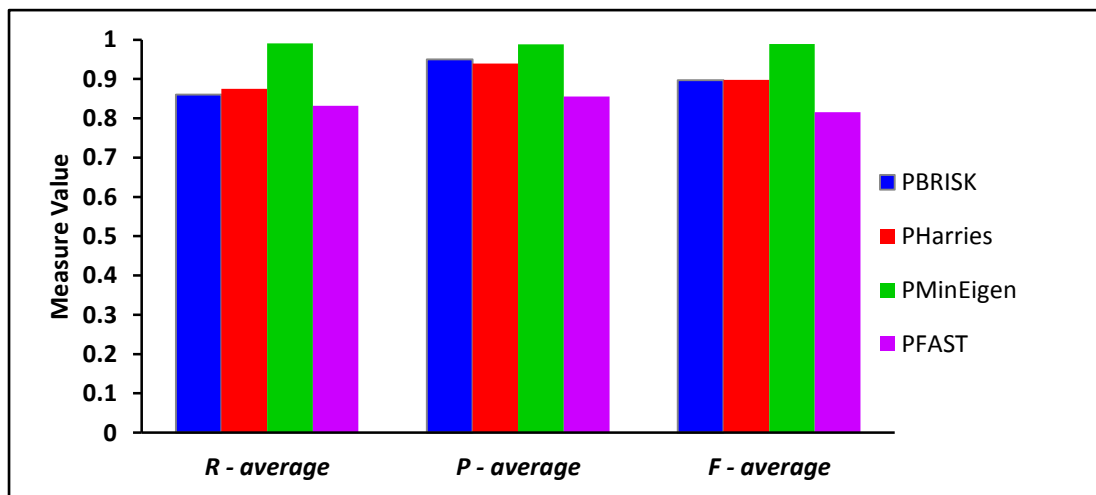


Fig.(6): Performance comparison.

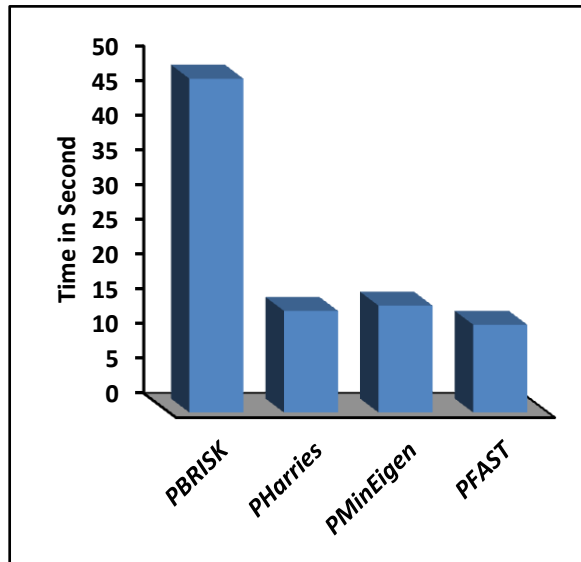


Fig.(7): Execution Time.

Upon examining the results obtained; it is observed that the best performance is achieved using MinEigen while the worst performance is achieved using FAST in this algorithm when considering F measure. On the other hand the FAST is superior to others when considering execution time. Moreover, differences in performance of the four corner detector vary with the different video category. The comparison in this paper show the proposed method that based on MinEigen features achieve good accuracy for detecting abrupt shot and reduce the effects of the object motions and camera movement for different kinds of videos with reasonable execution time.

References

- [1] Singh, R. D. and Aggarwal, N. "Novel research in the field of shot boundary detection—a survey", in *Advances in Intelligent Informatics*, eds. E.S. El-Alfy, S. Thampi, H. Takagi, S. Piramuthu, T. Hanne, Springer, Cham, 457-469, 2015.
- [2] Ko, K.C., Cheon, Y. M., Kim, G., Choi, H., Shin, S. and Rhee, Y. W. "Video shot boundary detection algorithm", in *Computer Vision, Graphics and Image Processing*, eds. S.P. Kalra and S. Peleg, Springer, Berlin, Heidelberg, 388-396, 2006.
- [3] Mondal, J., Kundu, M. K., Das, S. and Chowdhury, M. "Video shot boundary detection using multiscale geometric analysis of nsct and least squares support vector machine", *Multimedia Tools and Applications*, 1-23, DOI 10.1007/s11042-017-4707-9, 2017.
- [4] Qian, X., Liu, G. and Su, R. "Effective fades and flashlight detection based on accumulating histogram difference", *IEEE Transactions on Circuits and Systems for Video Technology*, 16(10), 1245-1258, 2006.
- [5] Janwe, N. J. and Bhojar, K. K. "Video shot boundary detection based on JND color histogram", In *Proceedings of the IEEE Second International Conference on Image Information Processing*, 476-480, 2013.
- [6] Xiao, Y.; Xia, L.; Zhu, S.; Huang, D. and Xie, J. "Video shot boundary recognition based on adaptive locality preserving projections", *Mathematical Problems in Engineering*, 2013(353261), 1-10, 2013.
- [7] Hassaballah, M., Abdelmgeid, A. A. and Alshazly, H.A. "Image Features Detection, Description and Matching", In *Image Feature Detectors and Descriptors*, Studies in Computational Intelligence, eds. A.I. Awad and M. Hassaballah, Springer, Cham, 11-45, 2016.
- [8] Tuytelaars, T. and Mikolajczyk, K. "Local invariant feature detectors: a survey", *Foundations and trends in computer graphics and vision*, 3(3), 177-280, 2008.
- [9] Li, J.; Ding, Y., Shi, Y. and Li, W. "Efficient shot boundary detection based on scale invariant features", In *Proceedings of IEEE Fifth International Conference on Image and Graphics*, 952-957, 2009.
- [10] Santos, A. and Pedrini, H. "Shot Boundary Detection for Video Temporal Segmentation based on the Weber Local Descriptor", In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 1310-1315, 2017.
- [11] Shao, H., Qu, Y. and Cui, W. "Shot boundary detection algorithm based on HSV histogram and HOG feature", In *Proceedings of 5th International Conference on Advanced Engineering Materials and Technology*, 951-957, 2015.
- [12] Kar, T. and Kanungo, P. "A motion and illumination resilient framework for automatic shot boundary detection",

- Signal, Image and Video Processing, 1-8, DOI 10.1007/s11760-017-1080-0, 2017.
- [13] Thounaojam, D. M., Roy, S. and Singh, K. M. "Video Shot Boundary Detection using Gray Level Cooccurrence Matrix", *Indian Journal of Science and Technology*, 9(7), 1-5, 2016.
- [14] Hannane, R., Elboushaki, A., Afdel, K., Naghabhushan, P., and Javed, M. "An efficient method for video shot boundary detection and keyframe extraction using SIFT-point distribution histogram", *International Journal of Multimedia Information Retrieval*, 89-104, DOI 10.1007/s13735-016-0095-6, 2016.
- [15] Baber, J., Afzulpurkar, N. and Satoh, S. "A framework for video segmentation using global and local features". *International Journal of Pattern Recognition and Artificial Intelligence*, 27(05), 1355007-1 - 1355007-29, 2013.
- [16] Zheng, Y. and Zhang, Y. "Abrupt shot boundary detection with combined features and SVM", In *Proceedings of 2nd IEEE International Conference on Computer and Communications*, 409-413, 2016.
- [17] Leutenegger, S., Chli, M. and Siegwart, R.Y. "BRISK: Binary robust invariant scalable keypoints", In *Proceedings of IEEE International Conference on Computer Vision*, 2548-2555, 2011.
- [18] Wan, L.; Yu, Z. and Yang, Q. "Corner Detection Algorithm with Improved Harris", In *Chinese Conference on Image and Graphics Technologies*, eds. T. Tan et al., Springer-Verlag, Berlin Heidelberg , 260-271, 2015.
- [19] Mainali, P.; Yang, Q., Lafruit, G., Gool, L.V. and Lauwereins, R. "Robust low complexity corner detector", *IEEE Transactions on Circuits and Systems for Video Technology*, 21(4), 435-445, 2011.
- [20] Rosten, E. and Drummond, T. "Machine learning for high-speed corner detection", in *Computer Vision–ECCV*, eds. A. Leonardis, H. Bischof, and A. Pinz, Springer-Verlag, Berlin Heidelberg, 430-443, 2006.