A.A. Karim
Computer Sciences Dept., University of Technology Baghdad, Iraq.

E. F. Nasser
Computer Sciences Dept., University of Technology Baghdad, Iraq.
ekhlas_uot1975@yahoo.com

Received on: 04/08/2016
Accepted on: 08/06/2017

Improvement of Corner Detection Algorithms (Harris, FAST and SUSAN) Based on Reduction of Features Space and Complexity Time

Abstract- *The active detection for gratifying features can be a definitive pace for computer vision in different tasks. Corners become more preferable models because of their two dimensional constrain; two dimensional limitations and algorithms can be rapid to detect them. Corners in images form significant information. Elicitation corners precisely are significant for processing image data to minimize a lot of computations. This paper can be used three vastly algorithms for detection the corner in images improvement Harris, improvement FAST, and improvement SUSAN which are based on two criteria for comparison to minimize the space of interest features and runtime reduction. From that, it can conclude that the algorithm of improvement FAST was outstanding to improvement Harris and improvement SUSAN algorithms on these criteria. FAST, SUSAN and Harris algorithms for corner detected were improved by applying Haar transform and choosing an adaptive gray difference threshold. Improvement FAST, has been offered which can be exceeded the previous two algorithms, improvement Harris and improvement SUSAN in both less run time and small features space. For example, the time taken by car image is 0.0005 second to extract the features using improvement FAST algorithm, which is much less than that used by the SUSAN and Harris algorithms. Improvement Harris takes 0.0074second and SUSAN takes 0.0096 second.*

Keywords- *Corner detection, Harris, SUSAN, FAST, Haar Transform, adaptive gray difference threshold.*

How to cite this article: A.A. Karim and E. F. Nasser, "Improvement of Corner Detection Algorithms (Harris, FAST and SUSAN) Based on Reduction of Features Space and Complexity Time," *Engineering and Technology Journal*, Vol. 35, Part B, No. 2, pp. 112-118, 2017.

1. Introduction

Corners are important local attributes in images. These points have high drooping and lie in cross of brightness of image's areas. In a diversity of attributes in image, corners cannot be affected by lighting and can be have rotational constancy. Corners form 0.05% in the all image's pixels [1]. Without missing data of image, elicitation corners can reduce image's data processing. Therefore, the detection of corner has factual value and it plays an important place for motion tracking [2], image matching [3], augmented reality [4], representation of image [5] and other different fields. Tremendous techniques for detection the corners was suggested from multiple searchers. These techniques are divided into two groups: group of techniques focus on contour and the other group focus on intensity. Techniques focus on contour work at first to extract all contours from the image and then seek for points that have maximum diversity over those contours. Masood discovers corners on flat curves by slipping along the curve three rectangles and computing the occurrences of points of the contour that lie in every rectangle [6]. The technique that offers by Peng works on boundary corner detection which transforms the image using wavelet for its mastery to detect sharp divergence [7]. Scale space corner detectors diversity deals also with the class of technique that based on contour [8].

Techniques focus on intensity assessment a measure that can be indicated the existence of a corner from the gray image values directly. These techniques are differentiating by fast speed and independence of local features [9].

Three algorithms of this type of techniques were used in this paper SUSAN, FAST and Harris with improvement to detect the corner. These three algorithms were compared based on reducing both features space and time complexity.

2. Corner Detection Methods

Three vastly algorithms for detection the corner with improvement can be used in this paper, such as

I. SUSAN Corner Detection algorithm

The first algorithm to detect the corner is called SUSAN (Smallest Univalue Segment Assimilating Nucleus) .The algorithm was recognized by circular mask. If compared to the brightness of pixel mask with the nucleus of the mask brightness, then a mask's region can be introduced which own the same (or similar) brightness as the nucleus [10]. This mask's region was called "USAN", stand for "Univalue Segment

https://doi.org/10.30684/etj.2017.138622
2412-0758/University of Technology-Iraq, Baghdad, Iraq

Assimilating Nucleus". Suppose Figure1 that shows on a white background a shade rectangle, they are five rotational masks discern at various image's locations. The detection of corners can be based on the region of USAN. When the region of USAN is up to the smallest then nucleus is on the corner, such as location "a".
For corners detection, the function between every pixel of a mask and nucleus of that mask can be used for comparison, this function can be given by Eq. (1):

$$\mathrm{comp}(\mathrm{pix},\mathrm{pix}_0) = \begin{cases} 1, & |\mathrm{img}(\mathrm{pix}) - \mathrm{img}(pix_0)| < \mathrm{thr}; \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Where pix_0 is pixel coordinates of nucleus ; *pix* is the pixel coordinates of another points in the mask; *comp*(*pix*, pix_0) represents outcome of comparison function; *Img*(*pix*) represent pixel point at gray value; *thr* represent gray difference's threshold which can be used to determine the feature space ability size and lesser contrast that discovered by SUSAN revealer algorithm [11].

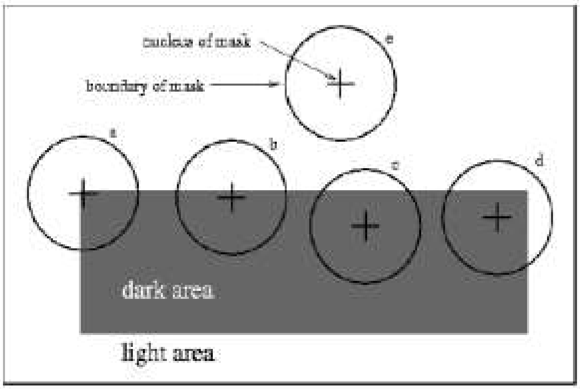


Figure 1: Four circular masks at different places on a simple image

II. Harris *Corner Detection algorith*m
The second algorithm can be used in this paper was Harris algorithm for corner detection. This algorithm can be recognized by calculating the gradient of each pixel's.
The pixel can be taken as a corner if the absolute gradient values in two directions are both mighty. Harris algorithm for detection the corner can be calculated by Eq. (2):

$$\mathrm{Reult} = \mathrm{detm(G)} - \mathrm{ktr}^2\,(\mathrm{G}) \quad (2)$$

$$\mathrm{G(a,b)} = \begin{bmatrix} \mathrm{Img}_m^2\,(a,b) & \mathrm{Img}_{mn}(a,b) \\ \mathrm{Img}_{mn}(a,b) & \mathrm{Img}_m^2\,(a,b) \end{bmatrix} \quad (3)$$

$$\mathrm{Img}_m^2\,(a,b) = A^2 \otimes \mathrm{GAUS}(a,b),$$
$$\mathrm{Img}_n^2\,(a,b) = B^2 \otimes \mathrm{GAUS}(a,b),$$
$$\mathrm{Img}_{mn}(a,b) = AB \otimes \mathrm{GAUS}(a,b),$$

$$\mathrm{GAUS}(a,b) = \frac{1}{2\pi} e^{\frac{-a^2+b^2}{2}} \quad (4)$$

Where $Img_m(a,b)$ and $Img_n(a,b)$ are gray values partial derivatives in coordinate *m* and *n* at pixel (*a*,*b*), and $I_{mn}(a,b)$ represent the partial derivative jumbled of second-order; *k* represent practical value; *GAUS*(*a*,*b*) refer to a Gaussian function; *A* and *B* are the directional differentials at first-order, which can compute by convolving gray values and operators of difference in coordinate *m* and *n*.
A function of Gaussian that used to minimize the noise affect due to first-order of directional differentials which are critical to noise. Pick the pixel's point as a corner if *R* exceeds the values of threshold [12].

III. FAST Corner Detection algorithm
Feature from an accelerated segment test (FAST) uses a Bresenhams algorithm for circle drawing with diameter of 3.4 pixels for trial mask. Trial 16 pixels compared to the nucleus's value for a complete accelerated segment using threshold. The criterion of corner should be more relaxed to block this broad trial. A pixel's criteria must be a corner based on an accelerated segment test (AST) which there must exist at least S pixels that have more brilliant circle connection or darker than a threshold . Other values of 16 pixels are disregarded. So the value of S can be used to determine the detected corner at maximum angle [13].

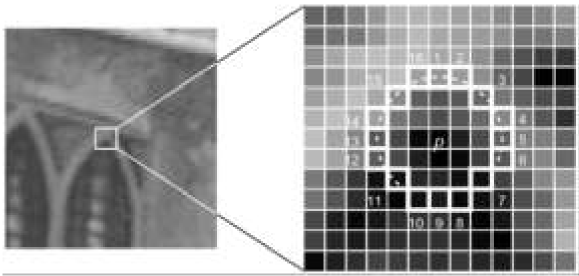


Figure 2: Image display the point of interest under a test and the circle of 16 pixels

Steps of FAST algorithm
1. From an image, chose a pixel p". IP represent pixel's intensity. This pixel can be specified as a point of interest or not. (Returning to figure (2)).
2. Get **thr** that **... 3** the value of threshold intensity.
3. Assume periphery a pixel p represents the center of circle which has16 pixels. (Brenham circle of radius 3.)

4. Need "N" exposure contiguous pixels out of the 16 pixels, either below or above IP by **thr** value, if the pixel wants to discover as a point of interest.
5. First match 1, 5, 9 and 13 of the circle pixels' intensity with IP to make an algorithm fast. From figure (2), at least three of these four pixels should accept the norm of the threshold for this it subsist an interest point. P is not an interest point (corner) if at least three values of - I1, I5, I9 and I13 are not below or above IP + **thr** . For this, a pixel p can be rejected as a potential point of interest. Else if three pixels at least are up or down Ip + **thr**, for whole 16 pixels seek and check if 12 neighboring pixels drop in the norm.
6. A same procedure can iterate for whole image's pixels.

3. Proposed methodology

The suggested method in this paper for feature extraction consists from four main steps. Firstly, images database can be taken and these images were transformed to frequency domain by applying Haar filter. Secondly, the features of interest are extracted using adaptive threshold with SUSAN, FAST and Harris algorithms. Thirdly, compute the numbers of extracted features and the time that could be taken by these methods. Fourthly, determine the best method which has less run times and little numbers of features.

Figure (3) displays block diagram of the suggested system to extract features.

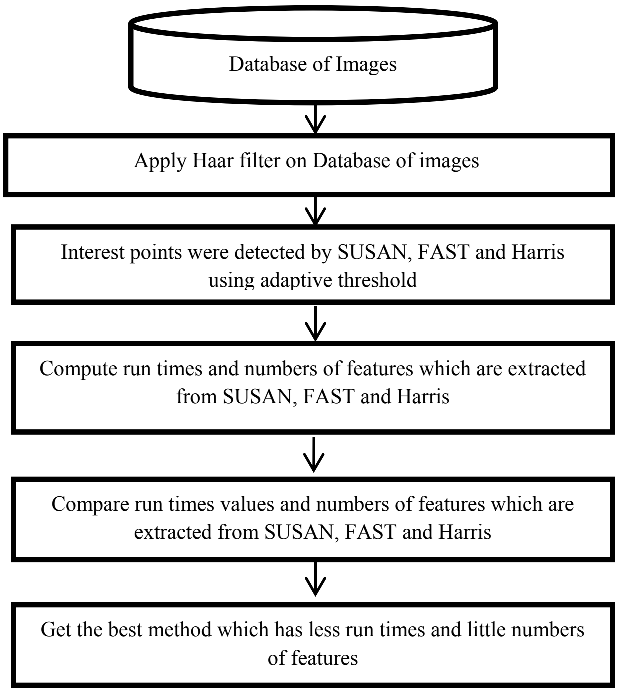


Figure 3: Block diagram of proposed system for feature extraction

I. Wavelet Transform using Haar Filter

Wavelet transforms based on sub-sampling low pass and high pass filters (Quadrature Mirror Filters (QMF)). By splitting the data into low pass band and high pass band with or without losing any information, matching the filters is done. Wavelet filters can be organized for applications of a broad range and numerous different sets of filters can be proposed for various applications. Wavelets are functions identified over a finite interval. The purpose from wavelet transform is to transform the data from **Time-space domain** to **Time-frequency domain** which can perform best compression results. There are a wide variety of popular wavelet algorithms, including Daubechies wavelets, Mexican Hat wavelets and Morlet wavelets. These algorithms have the disadvantage of being more expensive to calculate than the Haar wavelets. Haar wavelet is a simplest form of wavelets; the function is defined in Eq.5. The four bands are indicates to Low-Low (LL), Low-High (LH), High-Low (HL) and High-High (HH). It can potential to implement group of wavelet filters on LL band with self-path as implemented to the main image because it contains image-like information. An image dividing operation into sub-bands can be permanent as far as wished (based on an image resolution), probably for image compression it is commonly continued only to 4 or 5 levels [14] .Figure (4) shows a wavelet transform on gray scale image.

$$\varphi(x) = \begin{cases} 1 & 0 \le x < 1/2 \\ -1 & 1/2 \le x < 1 \\ 0 & \text{otherwise} \end{cases} \qquad (5)$$

Figure (4) shows an example of applying Haar transform on Lena image

Figures 4: Lena images a) Original image b) Two Dimensional Haar Transform

II. Improvement of SUSAN, FAST and Harris algorithms

To reduce feature space complexity of an image and increase the implementation speed of the suggested methodology, the proposed method was used an adaptive threshold **thr** instead of static threshold. An adaptive threshold can be computed using Eq. (6).

$$thr=(Img_{max} - Img_{min})/2 \quad (6)$$

where Img_{max} and Img_{min} are the largest and smallest gray value of whole image.
To improve SUSAN method, the threshold value in Eq. (1) can be computed from Eq. (6).
To improve FAST method, the threshold value in step (2) of an algorithm can be taken from Eq. (6).To improve Harris method, the value of k in Eq. (2) can be computed from Eq. (6).

4. Evaluation of an algorithms after improvement

The detection of the corner has been executed using improvement SUSAN, improvement FAST, and improvement Harris methods. The proposed system can compare them and can be evaluated using different criterions. There are two criterions can be used such as number of features which are resulted from each method and complexity time that can be taken each method for features extraction. The criterions can be illustrated bellow:-

I. Feature Space reduction criterion

Large corpuses of representative images can be used to build the space of features. White ellipses denote the regions of local feature. The size of feature space becomes large based on collecting a large sample of features from corpuses of images representation. For real time processing, large feature space can be taken more computational cost, therefore, it must be reduced. From practice, it can be seen that improvement SUSAN algorithm for detection of corner can be generated more features compared with improvement FAST and improvement Harris algorithms.
To decrease the numbers of features which are resulted from a corner detection algorithm, an improvement of FAST corner detection can be used.

II. Runtime and Complexity criterion

High speed and less algorithm's complexity must be available for real time tasks. An algorithm must be speedily enough to use in the ultimate system for image processing. The algorithm's real time can be describing the complexity of an algorithm.
To execute the algorithm with the least complexity and increase speed, Haar filter can be used with adaptive threshold.

5. Proposed algorithm

The algorithm of the proposed method is illustrated as:

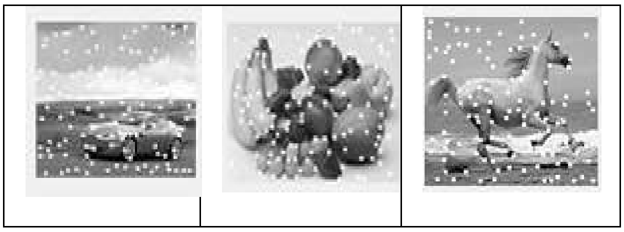
Input : Images from Database **Output :** Number of extracted features and the consumer time to extract these features
Step1: Select one image from a database **(Img)** and covert an image to gray scale **(Gray_Img)** **Step 2:** Compute Haar transform on image **(Gray_Img)** and put the result in **(H_img)** using Eq.**5**. **Step 3:** Detect the interest points for **(H_img)** using SUSAN, FAST , and Harris corner detection with adaptive threshold based on Eq.6 and put the results in **(D_S_H_Img)** ,**(D_F_H_Img)**, and, **(D_H_H_Img)** respectively. **Step 4:** Compute number of extracted features and time that can spend for features extraction from **(D_S_H_Img)** ,**(D_F_H_Img)**, and, **(D_H_H_Img)** respectively **.** **Step 5:** Compare run times values and numbers of features which are extracted from **(D_S_H_Img)** ,**(D_F_H_Img)**, and **(D_H_H_Img)** respectively**.** **Step 6:** Get the best method which has less run times and little numbers of features

6. Experimental results

The outcomes of suggested methodology are offered and discussed at this part. The suggested method is executed in C#. Three types of databases like (Car, Fruit, and Horse) are employed for evaluation the suggested method. Database images are colored, and with size 320 × 240 pixels. Figures (6), (7) and (8) represent the results of applying classical methods using static threshold.
The value of static threshold can be chosen randomly such as 40 which can be used in these figures. Figures (9), (10) and (11) represent the results of applying classical methods using adaptive threshold. The value of adaptive can be computed from Eq. (6).

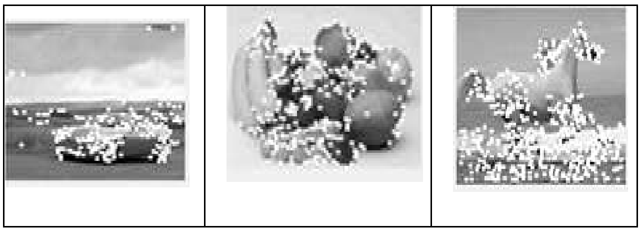
a) b) c)

Figure 5: Original images of a) Car database, b) Fruit database and c) Horse database



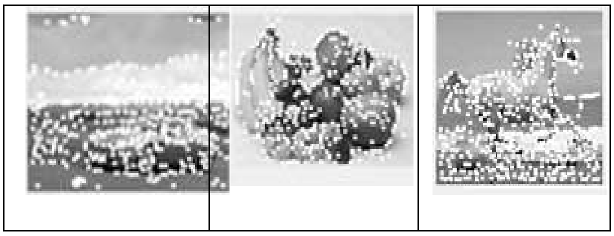
a) b) c)

Figure 6: Harris corner detection using static threshold a) Car, b) Fruit and c) Horse



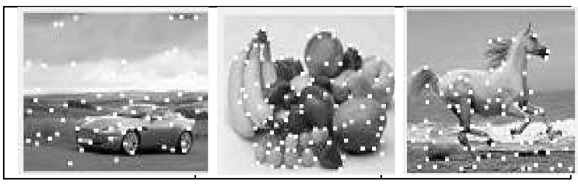
a) b) c)

Figure 7: FAST corner detection using static threshold a) Car, b) Fruit and c) Horse



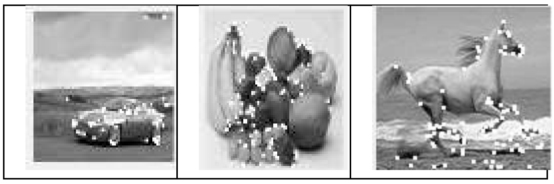
a) b) c)

Figure 8: SUSAN corner detection using static threshold a) Car, b) Fruit and c) Horse



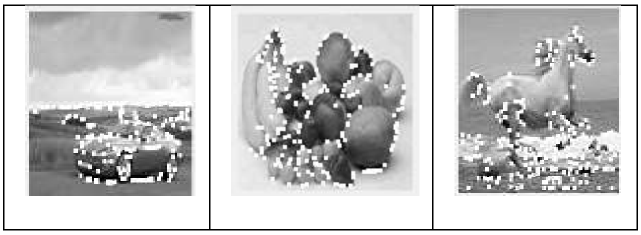
a) b) c)

Figure 9: Harris on images using adaptive threshold only a) Car, b) Fruit c) Horse



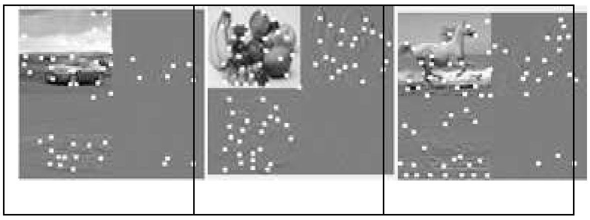
a) b) c)

Figure 10: FAST on images using adaptive threshold only a) Car, b) Fruit c) Horse



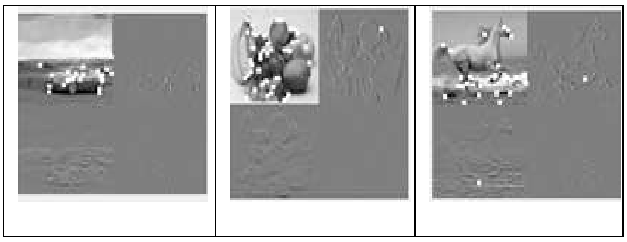
a) b) c)

Figure 11: SUSAN on images using adaptive threshold only a) Car, b) Fruit c) Horse



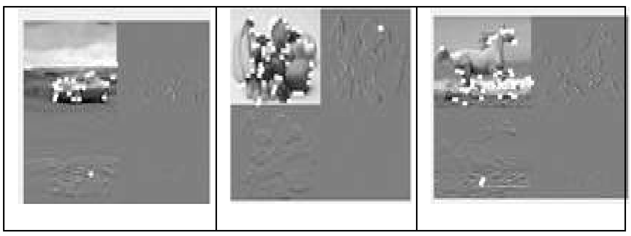
a) b) c)

Figure 12: Apply Harris to the images in the database generated by the Haar transform filter with adaptive threshold a) Car, b) Fruit and c) Horse



a) b) c)

Figure 13: Apply Improvement FAST to the images in the database generated by the Haar transform filter with adaptive threshold a) Car, b) Fruit c) Horse



a) b) c)

Figure 14: Apply SUSAN to the images in the database generated by the Haar transform filter with adaptive threshold a) Car, b) Fruit c) Horse

Tables below illustrate the comparison of the algorithms (improvement Harris, improvement FAST, and improvement SUSAN) depending on the time consuming for features extraction and number of extracted features.

Table1: Computational time in seconds' when using Harris algorithm on three databases of images (Car, Fruit, and Horse) can be used.

Evaluation Criteria	Time consuming for Databases name			Number of extracted features from Databases name		
	Car	Fruit	Horse	Car	Fruit	Horse
Classical Harris method	0.3870	0.1882	0.1088	90	98	90
Harris with adaptive threshold only	0.0212	0.0285	0.0295	67	96	87
Improvement of Harris (generated by Haar filter with adaptive threshold)	0.0074	0.0061	0.0054	٤٨	75	69

Table2: Computational time in seconds' when using FAST and improvement of FAST algorithms on three databases of images (Car, Fruit, and Horse) can be used.

Evaluation Criteria	Time consuming for Databases name			Number of extracted features from Databases name		
	Car	Fruit	Horse	Car	Fruit	Horse
Classical Harris method	0.8521	0.3319	0.4393	173	295	417
Harris with adaptive threshold only	0.1040	0.1356	0.1451	57	90	85
Improvement of Harris (generated by Haar filter with adaptive threshold)	0.0005	0.0008	0.0006	٢٠	34	22

Table3: Computational time in seconds' when using SUSAN algorithm on three databases of images (Car, Fruit, and Horse) can be used..

Evaluation Criteria	Time consuming for Databases name			Number of extracted features from Databases name		
	Car	Fruit	Horse	Car	Fruit	Horse
Classical Harris method	0.0248	0.0791	0.0686	354	381	450
Harris with adaptive threshold only	0.0167	0.0117	0.0137	193	212	222
Improvement of Harris (generated by Haar filter with adaptive threshold)	0.0096	0.0074	0.0153	٥٢	67	83

7. Conclusions

Haar filter is memory efficient, due to it could be computed in place without using a temporary array. Employing improvement FAST algorithm for detection of corner with Haar filter, make extraction of feature fast and more efficient for storage because it can be extracted less features but very important.

From the comparison results of Tables 1,2, and 3 of these three algorithms' for detection of corner, it can conclude that the algorithm of FAST after improvements was outstanding compared with the Harris and SUSAN algorithms in both feature space reduction and speed increases so it is suitable for processing in real time.

Drawbacks of Harris and SUSAN algorithms after improvement:-

1) They require more time for extraction the features.

2) They require much space to store the extracted features because they produce more features than improvement FAST.

It can conclude that Harris and SUSAN algorithms after improvement are not suitable for processing in real time.

References

[1] Y. Wenjia., D. Lihua, Z. Juan and L. Jinghua, “Automatic Moving Object Detection and Tracking in Video Sequences,” SPIE Fifth International Symposium on Multispectral Image Processing and Pattern Recognition, pp.676-712, 2007.

[2] D. Nguyen, "SURF Trac: Efficient Tracking and Continuous Object Recognition using Local Descriptors". Certified by IEEE PDF eXpress, March, 2009.

[3] C. Aswini and D. Chitra, "Enhanced Logo Matching and Recognition using SURF Descriptor", Department of Computer Science and Engineering, P. A. College of Engineering and Technology, Pollachi, Tamil Nadu, India, 2014.

[4] T. Kim and J. Im, “Automatic satellite image registration by combination of matching and random sample consensus,” *IEEE Trans on Geoscience and Remote Sensing*, Vol. 41, No. 5, pp.1111-1117, 2003.

[5] T. Tuytelaars and K. Mikolajczyk: "Local invariant feature detectors: a survey. Foundations and Trends" , in Computer Graphics and Vision3, pp. 177–280, 2008.

[6] A. Masood and M. Sarfraz. “Corner detection by sliding rectangles along planar curves” ,Computers & Graphics, Vol. 31, pp.440-448, 2007.

[7] L. Amsaleg , P. Gros , and S. Berrani., “Robust Object Recognition,” in Images and the Related Database Prob-lems, Multimedia Tools and Applications, v. 23 n. 3, p. 221–235, August 2004

[8] K. Sander and P. Ioannis , “The fast-3D spatio-temporal interest region detector”, in 10th workshop on Image Analysis for Multimedia Interactive Services WIAMIS09, pp. 242 - 245, 2009.

[9] R. Elias and R. Laganiere , “Cones: a new approach towards corner detection,” Proc Canadian Conference on Electrical and Computer Engineering, Vol. 2, pp.912-916, 2002.

[10] E. Rosten, R. Porter and T. Drummond, “FASTER and better: A machine learning approach to corner detection” in IEEE Trans. Pattern Analysis and Machine Intelligence, VOL 32, pp. 105-119, 2010.

[11] J. Chen, “The Comparison and Application of Corner Detection Algorithms”, Education Ministry Key Laboratory of Complex System Intelligent Control and Decision, Beijing Institute of Technology, Beijing, China, Sponsored by a provincial key project: No. 62301020403, Journal of Multimedia, VOL. 4, NO. 6 , DECEMBER February 20, 2009.

[12] S. Harpreet, “Robust Video Watermarking Algorithm Using K_Harries Feature Point Detection”, International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Volume-5 Issue-4, September 2015.

[13] M. Elmogy and H. El-Bakry , "Real Time Image Mosaicing System Based on Feature Extraction Techniques", Mansoura University, Conference Paper IEEE, · December 2014.

[14] D. Salomon, "Data Compression", the Complete Reference. Fourth Edition, Professor David Salomon (emeritus) Computer Science Department, California State University, Northridge, CA 91330 8281, USA, Springer-Verlag London Limited, 2007.