DOI: http://dx.doi.org/10.26483/ijarcs.v8i7.4238

ISSN No. 0976-5697

Volume 8, No. 7, July – August 2017

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

SFAST: A NEW ROBUST REGRESSION BASED COMBINED EDGES AND CORNER DETECTION

Muthukrishnan.R
Associate Professor, Department of Statistics, Bharathiar University, Coimbatore, TamilNadu, India.

Ravi.J
Assistant Professor, Department of Statistics, Vivekanandha College for Women, Unjanai, Tiruchengode, Namakkal, TamilNadu, India.

Abstract: Edge and Corner detection is a fundamental task in image processing and computer vision. Many procedures have been established during the past few decades. Yet there is a need of procedures which are less time consuming in nature and more accurate when considering blurred images. Recently, Feature Acclerated Segment Test (FAST), a corner detection model, had been presented which outperforms other procedures in both computational performance and repeatability. The FAST mainly uses simple regression and is used by machine learning approaches. This paper proposes a robust regression scale of residual (SSAC)-estimator based FAST algorithm namely, SFAST which can significantly improve its performance. SFAST is combined edge based corner detection method. The main feature of this proposed method is to use the edge points and their accumulated information for corner detection, for fast and more accurate results. The experimental results show that the proposed SFAST algorithm is fast, reliable and can be used in environments with noise.

Keywords: Robust statistics, Edge detection, Corner detection, Computer Vision.

1. INTRODUCTION

The field of computer vision is undergoing tremendous development in recent years. Computer vision is concerned with developing systems that can interpret the content of natural scenes. Computer vision systems begin with the process of detecting and locating some features in the input image. The degree to which a computer can extract meaningful information from the image is the most powerful key to the advance of intelligent image understanding systems [1]. Feature extraction and image segmentation plays a vital role to fill the gap between what we can get and what we want to have because corners are proven to be stable across sequences of images. One of the biggest advantages of feature extraction lies that it significantly reduces the information to represent an image for understanding the content of the image. Many computer vision algorithms use feature detection as the initial step, so as a result, a very large number of feature detectors have been developed based on edges and regions [2].

The Moravec operator is one of the well-known point extractors. This operator extracts points which have higher intensity variations; however the variations are only measured in four directions. The Harris algorithm is another method of corner detection, which defines the corner, measure function to detect corners. The feature of SUSAN method is a small disk-shaped mask is moved over the image pixels [3].

A large variety of methods have also been used for the task of feature matching. Among these methods, the similarity measure is one of the most powerful tools for feature matching. In order to find the corresponding point of a feature point using the similarity measure, a template window is considered around the feature point and this window is shifted pixel by pixel across a larger search window around an estimated corresponding point, and in each position, the similarity between the two regions is measured. The maximum or minimum value of the resultant measurements defines the position of the best match. Normalized cross correlation is a well-known method for measuring similarity between two regions. In addition to a normalized similarity value, normalized cross correlation has the advantage of being invariant to the linear change between the data sets, which makes the algorithm robust against low varying illumination which change the scene [4-5].

Another strategy to find matching points is the use of corners attributes. In these methods it is necessary for the corners to be detected in both images used for feature matching. The method has a low computation overhead, however the matching algorithm is sensitive to noise and illumination changes [6]. To increase the robustness of matching algorithms, other application dependent constraints such as target motion information in tracking applications or epipolar constraints in stereo vision may be used along with matching algorithms. One of the methods to increase the robustness of the matching process, especially in the tracking algorithms, is the use of statistical data association. In statistical data association the match point and search area are first estimated using motion information. Then the real match point is found using a method like normalized cross correlation. However, in non-uniform motions this may make the matching algorithm more erroneous [8, 9].

In this paper a robust algorithm based edge and corner detection method is proposed, namely SFAST method and its performance is studied using real images with error tolerance. The most widely used Harris, SUSAN, SIFT and FAST techniques, are briefly discussed in the section 2. The proposed SFAST technique and its computational algorithm are presented in the section 3. The performance of the proposed algorithm is carried out using different types of images/blurred images by using MATLAB software and the results thus obtained are summarized in the section 4. The last section discusses the conclusion of the study.

2. EDGE AND CORNER DETECTION

2.1 Harris Corner detection

The Harris corner detection was introduced by Harris and Stephen in 1988. The Harris corner detector gives a noisy response due to a binary window function. These methods apply the Gaussian noisy filter [12, 13]. The Harris corner detector is based on the local auto-correlation function of a signal which measures the local changes of the signal with patches shifted by a small amount in different directions. Given a shift (x, y) and a point the auto-correlation function is defined as

$$C(x,y) = \sum w[\, I(x_i,y_i) - I(x_i+\Delta x, y_i+\Delta y)]^2$$

Not only do we need corner and edge classification regions, but also a measure of corner and edge quality or response. The size of the response will be used to select isolated corner pixels and to thin the edge pixels. The measures of corner response, R, which we require to be function:

$$T(M)=\alpha+\beta= A+B \text{ and } D(M)= \alpha\beta= AB-C^2$$

and the inspired formulation for the corner response, $R=D-k*(T)^2$.

2.2 Smallest Univalves Segment Assimilating Nucleus (SUSAN) corner detection

SUSAN corner detection was introduced by Smith and Brady 1997 [14]. The SUSAN's method doesn't use the derivatives of the image or edge pixels for corner detection. The feature of this method is a small disk-shaped mask is moved over the image pixels. The central point of the mask is called the Nucleus. The intensity value of the Nucleus is compared with other pixels in the mask. If the difference is less than a threshold, the pixel is categorized in a group called USAN. The area of the mask shall be known as the "USAN", as an acronym standing for "Univalves Segment Assimilating Nucleus". According to USAN values for different pixels, the locations of the edge pixels or corners are detected. Because this method doesn't use image derivatives, it has less sensitivity to image noise, especially snow nose. The SUSAN principle is formulated in the following equation, where $n(x_0)$ is the USAN size at x_0, on the simplification,

$$\frac{dI}{dx}(x_0 + a(x_0) - \frac{dI}{dx}(x_0 + b(x_0)) = 0$$

2.3 SIFT corner Detection

SIFT (Scale Invariant Feature Transform) corner detection is realized by extracting distinctive invariant features from image algorithm was proposed by Lowe in 2004 [7,11]. For any object in an image, interesting points on the object can be extracted to provide a "feature description" of the object. This description, extracted from a training image, can then be used to identify the object when attempting to locate the object in a test image containing many other objects. To perform reliable recognition, it is important that the features extracted from the training image be detectable even under changes in image scale, noise and illumination. Such points usually lie in high-contrast regions of the image, such as object edges. Another important characteristic of these features is that the relative positions between them in the original scene shouldn't change from one image to another. They are rotation-invariant, which means, even if the image is rotated, we can find the same corners. It is obvious because corners remain corners in rotated image also.

2.4 FAST Corner Detection Algorithm

Several feature detectors have been established and many of them are really good. But when looking for a real-time application point of view, they are not fast enough. FAST (Features from Accelerated Segment Test) algorithm was proposed and extended by Edward and Tom (2006, 2010). The FAST method is based on the SUSAN corner detection. The center of a circular area is used to determine brighter and darker neighboring pixels. However, in the case of FAST, the whole area of the circle is not evaluated, only pixels in the discretized circle describing the segment is evaluated. Like SUSAN, FAST also uses a Bresenham's circle of diameter 3.4 pixels as test mask. Thus, for a full accelerated segment test 16 pixels have to be compared to the value of the nucleus. To prevent this extensive testing, the corner criterion uses a more relaxed approach. A small rotations of the camera may yield pixel configurations which have not been measured in the test images. And even if all the pixel configurations are present, a small rotation about the optical axis would cause the probability distribution of the measured pixel configurations to change drastically. This may result in an incorrect and slow corner response. To learn the probabilistic distribution of a certain scene is therefore not applicable unless only the same viewpoints and the same scene are expected.

In the FAST algorithm, the state of each pixel can be one of the possibilities. The FAST approach which uses machine learning to address the first two points. The process operates in two stages. First, to build a corner detector for a given n, all of the 16 pixel rings are extracted a set of images (preferably from the target application domain). These are labelled using a straightforward implementation of the segment test criterion for n and a convenient threshold.

For each location of the circle $x \in (1,2...16)$, the pixel at that position relative to p, can have one of three states,

$$S_{p\to x} = \begin{cases} d, & I_{p\to x} \le I_p - t(darker) \\ s, & I_p - t < I_{p\to x} < I_p + t(similar) \\ b, & I_p + t \le P_{p\to x}(brighter) \end{cases}$$

Let P be the set of all pixels in all training images. Choosing an x partitions P into three subsets, P_d, P_s and P_b, where

$$p_b = \{p \in P : S_{p\to x} = b\}$$

The FAST assumes that the closest edge to the expected edge position is the correct match. This can lead to a large number of correspondence errors if the motion is large. Edward and Tom (2006,2010)), the feature detection using FAST and machine learning approach procedures are summarized given below:

Algorithm: Feature detection using FAST

- Select a pixel 'p' in the image and the intensity of this pixel is denoted by I_p.
- Set a threshold intensity value T.
- Consider the circle of the pixel as 'p' and radius 3.

- 'N' neighboring pixels out of the 16 need to I_P by the value T, if the pixel needs to be detected as an interest point.
- To make the algorithm fast, first compare any four intensities of pixels of the circle. If at least three of the four pixel values I_p+T, then P is not a corner point. In this case reject the pixel p as a possible interest (corner) point. Otherwise, if at least three of the pixels are above or below I_p+T, then check for all 16 pixels and check if 12 contiguous pixels fall in the criterion.
- Repeat the procedure for all the pixels in the image.

Algorithm: Machine Learning Approach

- Select a set of images for training. In every image run the FAST algorithm to detect the interest points by taking one pixel at a time and evaluating all the 16 pixels in the circle.
- For every pixel 'p', store the 16 pixels surrounding it, as a vector Repeat this for all the pixels in all the images.
- Each value (one of the 16 pixels, say x) in the vector, can take three states. Darker than p, lighter than p or similar to p. Robust estimator (Simple regression), Depending on the states the entire vector P will be subdivided into three subsets, P_d, P_s, P_b.
- This order of querying which is learned from the decision tree can be used for faster detection in other images also.

These algorithms exhibits high performance, but there are some limitations:

- This high-speed test does not reject as many candidates for n < 12, since the point can be a corner if only two out of the four pixels are both significantly brighter or both significantly darker than p (assuming the pixels are adjacent). Additional tests are also required to find if the complete test needs to be performed for a bright ring or a dark ring.
- The efficiency of the detector will depend on the ordering of the questions and the distribution of corner appearances. It is unlikely that this choice of pixels is optimal.
- Multiple features are detected adjacent to one another.

3. S estimator Feature Accelerated Segment Test (SFAST)

The theory of the proposed SFAST method is described in this section. First the concept of SSAC- it is a robust estimator was briefly discussed in the regression context and then the implementation of S in the FAST algorithm is discussed.

3.1 Robust Sample Consensus Estimator

Rousseeuw and Yohai (1984) proposed the concept of the S-estimator, is based on a residual scale of M estimation [15]. The weakness of M estimation is the lack of consideration on the data distribution and not a function of the overall data because only using the median as the weighted value. This method uses the residual standard deviation to overcome the weaknesses of median. According to Salibian and Yohai (2006), the S-estimator is defined by $\hat{\beta}$ = minβ $\hat{\sigma}_s$ ($e_1,e_2,\dots,e_n$) with determining minimum robust scale estimator $\hat{\sigma}_s$.

Let X_1, X_2,……X_N be the set of all data points (inliers and outliers). First, assume that the initial data point X_0. Given a model that requires a minimum of n data points to instantiate its free parameters, and a set of data P such that the number of points in P is greater than n (P ≥ n) randomly selected subset S_1 of n data points form P and instantiate the model. Use the instantiated model M_1 to determine the subset S_1^* of points in P that are within some error tolerance of M_1. The set S_1^* is called the consensus set of S_1.

If S_1> t, use S_1^* to estimate a new model M_1^* and if S_1< t, randomly select a new subset S_2 and repeat this process. If after some predetermined number of trials, no consensus set with t or more members has been found, either solve the model with the largest consensus set found.

Without loss of function, that the noise in the images is Gaussian on each image coordinate with zero mean and uniform standard deviation σ. Thus, the joint probability density function of the inliers is

$$P(x,y,z) = \begin{cases} \frac{1}{4t^2\sigma\sqrt{2\pi}} exp\left(\frac{-z^2}{2\sigma^2}\right) + (1-\gamma)\frac{1}{v}, & -t < x < t, -t < y < t, -t < z < t \\ 0 , & otherwise \end{cases} \tag{3.1}$$

One of the robust regression estimation methods is the M estimation and is defined by $\hat{\beta} = \beta_n(x_1, x_2 \dots\dots x_n)$ then

$$E[\beta_n(x_1, x_2 \dots\dots x_n)] = \beta \tag{3.2}$$

Equation (3.2) shows that the estimator $\beta^\wedge = \beta_n(x_1, x_2 \dots\dots x_n)$ is an unbiased and has minimum variance is

$$var(\hat{\beta}) \geq \frac{\beta^2}{nE\left(\frac{d}{d\beta} Inf(x_i, \beta)\right)^2} \tag{3.3}$$

where $\hat{\beta}$ is other linear and unbiased estimator for β. The principle is to minimize the residual function ρ,

$\hat{\beta}_s = \frac{min}{\beta} \rho(y_i - \sum_{j=0}^{k} x_{ij}\beta_j)$. We have to solve

$$\frac{min}{\beta} \sum_{i=1}^{n} \rho(u_i) = \frac{min}{\beta} \sum_{i=1}^{n} \rho\left(\frac{e_i}{\hat{\sigma}_s}\right) = \frac{min}{\beta} \sum_{i=1}^{n} \rho\left(\frac{y_i - \sum_{j=0}^{k} x_{ij}\beta_j}{\hat{\sigma}_s}\right)$$

where $\hat{\sigma}_s = \sqrt{\frac{1}{nk}\sum_{i=1}^{n} \omega_i e_i^{\,2}}$ and

$$\omega_i = \omega_\sigma(u_i) = \frac{\rho(u_i)}{u^2{}_i} \tag{3.3}$$

The initial estimate is , $\hat{\sigma}_s = \frac{median|e_i - median(e_i)|}{0.6745}$ and

$$u_i = \frac{e_i}{\sigma_s} \tag{3.4}$$

For ρ function we use the Tukey's bi-square objective function

$$\rho(u_i) = \begin{cases} \frac{u_i^2}{2} - \frac{u_i^4}{2c^2} + \frac{u_i^6}{6c^4}, & |u_i| \leq c \\ \frac{c^2}{6}, & |u_i| > c \end{cases} \tag{3.5}$$

Furthermore we look for first partial derivative the solution is obtained by differentiating to β so that $\widehat{\beta_s}$ to β so that

$$\sum_{i=1}^{n} x_{ij} \emptyset \left(\frac{y_i - \sum_{j=0}^{k} x_{ij}\beta_j}{\widehat{\sigma_s}} \right) = 0 \qquad , j=0,1,2,\ldots\ldots k$$

where $\emptyset = \rho^{|}$, x_{ij} is i-th observation on the j-th independent variable and $x_{i0}=1$. The function $\emptyset$ is a function as derivative of ρ is,

$$\emptyset(u_i) = \rho^{|}(u_i) = \begin{cases} u_i\left[1-\left(\frac{u_i}{c}\right)^2\right]^2 , & |u_i| \le c \\ 0 , & |u_i| > c \end{cases} \tag{3.6}$$

The usual choice is c= 1.547 and k= 0.199 for 50% breakdown and about 28% asymptotic efficiency. So the equation (5.8) becomes

$$\sum_{i=1}^{n} x_{ij} w_i \left(y_i - \sum_{j=0}^{k} x_{ij}\beta\right) = 0 , j=0,1,\ldots\ldots k \tag{3.7}$$

where is an iteratively reweighted least square (IRLS) weighted function in the bi-square family with score function is

$$\omega_i (u_i) = \begin{cases} \left[1-\left(\frac{u_i}{c}\right)^2\right]^2 , & |u_i| \le c \\ 0 , & |u_i| > c \end{cases}$$

Solve equation (3.7) by using IRLS method, assuming that there is an initial estimate β_0 , $\emptyset_0$ and $\widehat{\sigma_i}$ is a scale estimate. If j is the number of parameters then,

$$\sum_{i=1}^{n} x_{ij}\, \emptyset_0 \left(\frac{y_i - \sum_{j=0}^{k} x_{ij}\, \beta^0}{\widehat{\sigma_s}} \right) = 0 , \ j=0,1,2,\ldots\ldots k \tag{3.8}$$

The matrix notation, equation (3.8) can be written as

$$X^{|}\emptyset_i X\beta = X^{|}\emptyset_i Y \tag{3.9}$$

where $\emptyset_i$ is a n x n matrix with its diagonal elements are the iteratively reweighted. Equation (3.9) is known as IRLS equation. The solution of this equation gives an estimator for $\hat{\beta}$, is given by

$$\hat{\beta} = (X^{|}\emptyset_i X)^{-1}(X^{|}\emptyset_i Y)$$

The computational S-estimator based sample consensus (SSAC) algorithm is proposed by Muthukrishnan.R and Ravi. J, [10] used in the place of sample selection which is as follows:

Algorithm: SSAC

Step1: Select initial data point x_0 from X at random.

Step2: Calculate robust regression coefficients on the data β_0

Step3: Calculate residual value ε_i.

Step4: Calculate value of $\widehat{\sigma_i}$ and the value of u_i.

Step5: Calculate $\widehat{\beta_s}$ with WLS method with weighted w_i.

Step6: Repeat the above steps to obtain $\widehat{\beta_s}$ for converges.

Since it is infeasible to store an infinite number of models, the model with lowest estimated probability of detection is replaced with the newest model generated when processing an outlier. To avoid multiple estimates of the same models, after processing each observation scan similar model estimates are combined using the threshold. Good models are determined when the estimated probability of detection is greater than some threshold.

3.2 SFAST (5.8)

Edges are the key points in the corners and also curve extraction is the one of the most powerful tools in the edges. The edges are generated by the curves based on the regression lines. Most of the methods to be applied, generally use least square regression line and fit the curves. Outliers can affect the least square fit and thus gives imperfect edges. FAST algorithm uses least squares procedure. Least square procedure is not robust, hence it gives imperfect edges and leads to improper curve extractions. The proposed method uses the robust estimator namely S-estimator in the place of least square in the FAST algorithm, namely, SFAST. Generally, most lines fit the regression lines randomly, but does not verifies whether the line fit is good or not. But our SFAST method verifies whether the regression line fit is good or not. The SFAST corner detector is a suitable corner detector, which extracts the corners of the image from the contours of the edge detected image. The SFAST corner detector works as follows:

Algorithm: SFAST

In the proposed SFAST combined edge and corner detection, detailed configuration space is considered in order to provide a more efficient solution, instead of only considering a restricted configuration space, as in FAST algorithm. In classical robust estimator fashion, the most likely model parameters are computed by improving the probability of the observed data given the parameters. The proposed method SFAST works in the same manner as (3.8) FAST, but only difference is, it applies robust regression technique (S-estimator) instead of simple regression.

Although the CSS (Curvature Scale Space) corner detector considers the edge junctions and edge curvature, which are good features for edge matching, the curvature of only one contour is considered. In other words, each contour is handled separately for the purpose of corner detection. This enables the CSS algorithm to have good corner localization properties; however, some of the features which are proper for matching are not detected. To detect more appropriate edge features, we have developed an edge feature detector algorithm which considers the accumulated curvature of edge pixels in the match window. The algorithm also considers the number of edge pixels in the match window, which is another useful factor for correct edge matching.

The SFAST algorithm consists the following steps:

- Extract the edge contours from the input image using any good edge detector such as Canny.
- Fill small gaps in edge contours. When the gap forms a T, mark it as a T-corner. To fill small gaps in the edge contour, we check the small windows (typically 6*6) centered on the end points of the contours. In the case of at least two other edge contours in the neighborhood of an end point, it is considered as a T. If only one contour is found, two contours are merged by considering the pixels in the short distance between their end points as edge pixels.
- Calculate the curvature of Gaussian smoothed edge pixels. Because of the averaging property of

accumulated curvature, in our algorithm, it is not necessary to calculate the curvature at different scales to reduce the noise effect.

4. EXPERIMENTAL RESULTS

The SFAST algorithm was implemented in MATLAB software. Performance of the proposed algorithm is tested with different image types including both actual and noisy (salt and pepper) images. The Canny edge detector is used for the extraction of edge points, and gaps for 1 pixel wide are filled for the detection of edge features. The processing time for detection of edges and curve extracting in images with and without noise is summarized in Table 4.1. Table 4.2 displays the number of true/false corners detected along with the processing time. The Edges and curve extracted, and the number of corners detected in images are shown in figures 4.1 and 4.2 respectively given in appendix.

Table 4.1: Time taken for detection edges and curve extracting (with and without noise)

Image	Harris			SUSAN			SIFT			FAST			SFAST		
	JPG	BMP	GIF	JPG	BMP	GIF	JPG	BMP	GIF	JPG	BMP	GIF	JPG	BMP	GIF
Processing Time (in Seconds) - (Image without noise)															
Cameraman	1.281 (0.71)	0.412 (0.38)	0.424 (0.42)	0.415 (0.41)	0.393 (0.37)	0.376 (0.38)	0.642 (0.39)	0.301 (0.41)	0.299 (0.39)	0.215 (0.13)	0.314 (0.41)	0.371 (0.40)	0.105 (0.07)	0.281 (0.32)	0.270 (0.34)
House	0.918 (0.13)	0.735 (0.13)	0.576 (0.51)	0.510 (0.42)	0.399 (0.39)	0.381 (0.31)	0.511 (0.38)	0.312 (0.39)	0.316 (0.41)	0.211 (0.11)	0.332 (0.38)	0.296 (0.38)	0.076 (0.07)	0.073 (0.09)	0.269 (0.31)
Angle box	2.131 (1.89)	1.913 (0.13)	1.399 (0.68)	1.992 (2.11)	1.110 (0.11)	1.634 (0.63)	1.876 (1.51)	0.976 (0.12)	1.698 (0.74)	1.631 (1.41)	0.471 (0.10)	1.213 (0.58)	1.368 (1.38)	0.117 (0.09)	1.146 (0.45)
Processing Time (in Seconds) - (Image with noise – added salt and pepper noise)															
Cameraman	1..671 (0.87)	0.611 (0.76)	0.712 (0.59)	0.515 (0.61)	0.593 (0.47)	0.426 (0.41)	0.744 (0.41)	0.391 (0.46)	0.400 (0.39)	0.325 (0.13)	0.412 (0.46)	0.442 (0.41)	0.395 (0.50)	0.384 (0.36)	0.415 (0.37)
House	1.011 (0.24)	0.860 (0.23)	0.756 (0.49)	0.587 (0.51)	0.423 (0.45)	0.512 (0.39)	(0.641 (0.41)	0.342 (0.40)	0.442 (0.47)	0.321 (0.24)	0.382 (0.39)	0.394 (0.41)	0.149 (0.10)	1.224 (1.48)	0.189 (0.13)
Angle box	2.530 (1.91)	1.987 (0.43)	1.971 (0.69)	2.091 (2.23)	1.469 (0.14)	1.732 (0.73)	1.972 (1.66)	0.998 (0.24)	1.796 (0.81)	1.771 (1.50)	0.491 (0.16)	1.289 (0.61)	0.156 (0.09)	0.135 (0.09)	0.149 (0.10)

(.) Indicate curve (Fitting) extracting time

It is observed from the table 4.1 that the performance of the proposed SFAST is better than the other edge detection methods. It is noted that edge detection and curve extraction timings are considerably reduced in SFAST procedure, since the proposed method works with 36 pixels, whereas FAST method works with 16 pixels.The cameraman image has 189 actual corners and also 81 and 57 corners for House and Angle box images respectively.

Table 4.2: Number of true/false corners detected with time taken (with and without noise)

Image	Harris			SUSAN			SIFT			FAST			SFAST		
	JPG	BMP	GIF	JPG	BMP	GIF	JPG	BMP	GIF	JPG	BMP	GIF	JPG	BMP	GIF
Number of Corners (without noise)															
Cameraman	**160** [25] (0.03)	**159** [23] (0.06)	**161** [20] (0.07)	**157** [24] (0.07)	**163** [21] (0.08)	**161** [19] (0.12)	**165** [18] (0.02)	**163** [20] (0.06)	**166** [17] (0.07)	**166** [16] (0.02)	**164** [19] (0.06)	**163** [21] (0.07)	**170** [12] (0.01)	**174** [14] (0.05)	**172** [12] (0.06)
House	**60** [15] (0.06)	**57** [21] (0.07)	**61** [17] (0.10)	**58** [19] (0.07)	**55** [21] (0.06)	**60** [20] (0.09)	**64** [13] (0.06)	**61** [14] (0.07)	**60** [16] (0.07)	**67** [11] (0.05)	**69** [13] (0.01)	**66** [16] (0.06)	**72** [8] (0.03)	**74** [7] (0.03)	**77** [10] (0.06)
Angle box	**46** [14] (0.29)	**40** [16] (0.30)	**41** [17] (0.42)	**45** [16] (0.28)	**38** [18] (0.29)	**40** [17] (0.34)	**50** [13] (0.28)	**47** [15] (0.29)	**45** [16] (0.29)	**50** [12] (0.27)	**49** [15] (0.28)	**47** [14] (0.29)	**55** [7] (0.22)	**54** [9] (0.22)	**52** [8] (0.28)
Added Salt and Pepper noisy															
Cameraman	**157** [27] (0.11)	**158** [24] (0.09)	**162** [22] (0.10)	**155** [28] (0.09)	**160** [23] (0.09)	**157** [22] (0.09)	**161** [22] (0.08)	**160** [22] (0.08)	**161** [24] (0.08)	**165** [19] (0.07)	**164** [21] (0.06)	**161** [19] (0.08)	**168** [13] (0.06)	**171** [16] (0.05)	**171** [13] (0.07)
House	**58** [19] (0.08)	**55** [25] (0.12)	**58** [20] (0.10)	**55** [18] (0.07)	**51** [20] (0.11)	**53** [20] (0.11)	**62** [16] (0.06)	**58** [17] (0.08)	**61** [19] (0.65)	**67** [16] (0.04)	**68** [16] (0.07)	**67** [19] (0.05)	**73** [11] (0.02)	**73** [10] (0.06)	**75** [12] (0.03)
Angle box	**48** [17] (0.08)	**55** [24] (0.08)	**40** [19] (0.07)	**46** [18] (0.08)	**40** [19] (0.07)	**41** [21] (0.87)	**52** [16] (0.06)	**49** [19] (0.05)	**47** [18] (0.06)	**51** [16] (0.04)	**52** [19] (0.03)	**49** [18] (0.03)	**55** [11] (0.02)	**53** [13] (0.02)	**54** [11] (0.02)

Bold font-True corners, [.]- False corners, (.)Processing time (in seconds).

It is noted that, the proposed SFAST method almost detected all the true corners of the image. Also, it is observed that the number of true corners detected is more than the number of true corners detected by the other methods which includes FAST method.

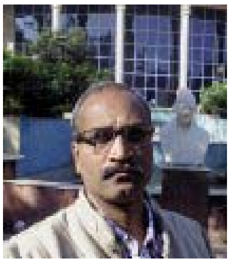
4. CONCLUSIONS

A new robust edge and corner detection algorithm is proposed. The superiority of the proposed algorithm can be tested with various features such as processing time, the number of true/false corners detected in images with different types. Also, the proposed method gives reliable results in case of noisy images or images with illumination change. It is suggested that the proposed algorithm can be used in various machine vision applications such as target tracking and image registration.

REFERENCES

1. Bandera, A., Urdiales, C., Arrebola, F. & Sandoval, F. (2000). Corner detection by means of adaptively estimated curvature function. Electronics Letters, 36(2), 124-126.
2. Canny, J. (1986). A computational approach to edge detection. IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6), 679-698.
3. Candocia, F. & Adjouadi, M. (1997). A similarity measure for stereo feature matching. IEEE Transaction On Image Processing, 6(10), 1460-1464.
4. Edward Rosten and Tom Drummond (2005). Fusing Points and Lines for High Performance Tracking, IEEE International Conference on Computer Vision ,1-8.
5. Elmar Mair, Gregory D. Hager, Darius Burschka, Michael Suppa, and Gerhard Hirzinger. Adaptive and generic corner detection based on the accelerated segment test. In Proceedings of the European Conference on Computer Vision *(ECCV'10)*, September 2010.
6. Harris, C. G. & Stephens, M. (1988). A combined corner and edge detector. In 4th Alvey Vision Conference, Manchester, England, 147-151.
7. Lowe,G. (2004). SIFT: The Scale Invariant feature Transfiorm, International journal of Computer vision, 60 (2), 91-110.
8. Moravec, H. P. (1977). Towards automatic visual obstacle avoidance. In Proc. of the International Joint Conference on Artificial Intelligence, Cambridge, MA USA, 584.
9. Mokhtarian, F. & Suomela, R. (1998). Robust image corner detection through curvature scale space.' IEEE Trans. Pattern Analysis and Machine Intelligence, 20(12), 1376-1381.
10. Muthukrishnan.R & Ravi.J .(2016). A robust regression scale of residual estimator: SSAC, International Journal of Applied Engineering Research, 11(7),5086-5090.
11. Muthukrishnan.R & Ravi.J .(2015). Image type- based Assessment of SIFT and FAST Algorithms, International Journal of Signal Processing, Image Processing and Pattern Recognition, 8(3), 211-216.
12. Muthukrishnan.R & Ravi.J .(2015). Time Based Appraisal of Corner Detection Technique, International Journal of Statistics and Systems, 10(2), 177-183.
13. Mokhtarian, F. & Suomela, R. (1998). Robust image corner detection through curvature scale space.' IEEE Trans. Pattern Analysis and Machine Intelligence, 20(12), 1376-1381.
14. Smith, S. M. & Brady, J. M. (1997). SUSAN - a new approach to low level image processing. Int. Journal of Computer Vision, 23(1), 45-78.
15. Salibian, M and V.J. Yohai (2006). A Fast Algoritm for S-Regression Estimates, Journal of Computational and Graphical Statistics, 15(2), 414-427.

About the Authors



R.Muthukrishnan

He has more than 17 years teaching and research experience in academic institutions. He holds a Degree in Computer Science, PG and Ph.D. in Statistics. Presently, he is working as Associate Professor in the Department of Statistics, Bharathiar University, Coimbatore, Tamil Nadu, India. His research area of interests includes Statistical Inference (specifically Robust Inference), Sampling Techniques and Multivariate Analysis. He has efficiency in handling computer languages such as programming in C++, VB and statistical software such as SPSS, SAS, MATLAB and R. He has participated in more than 50 National / International Conferences and presented his research articles. He has published about 40 research articles in various peer reviewed journals to his credit. He has delivered more than 30 special invited lectures. 17 scholars received M.Phil and Ph.D degree under his guidance. He is updating domain knowledge through participation in Training Programmes such as Workshop /Winter /Summer Schools /Orientation courses /Refresher courses.

J.Ravi

He has more than 11 years teaching experience in various Colleges. He is pursuing Ph.D. in Statistics in Bharathiar University, Coimbatore, Tamil Nadu. Presently, he is working as Assistant Professor in the Department of Statistics, Vivekanandha College for Women, Tiruchengode, Namakkal, Tamil Nadu, India. His research area of interests includes Statistical Inference, Image Processing, Computer Vision and Bio-Statistics. He has published about 6 research articles in various peer reviewed journals to his credit. He has delivered more than 5 special invited lectures. 15 scholars received M.Sc and M.Phil degree under his guidance.

APPENDIX

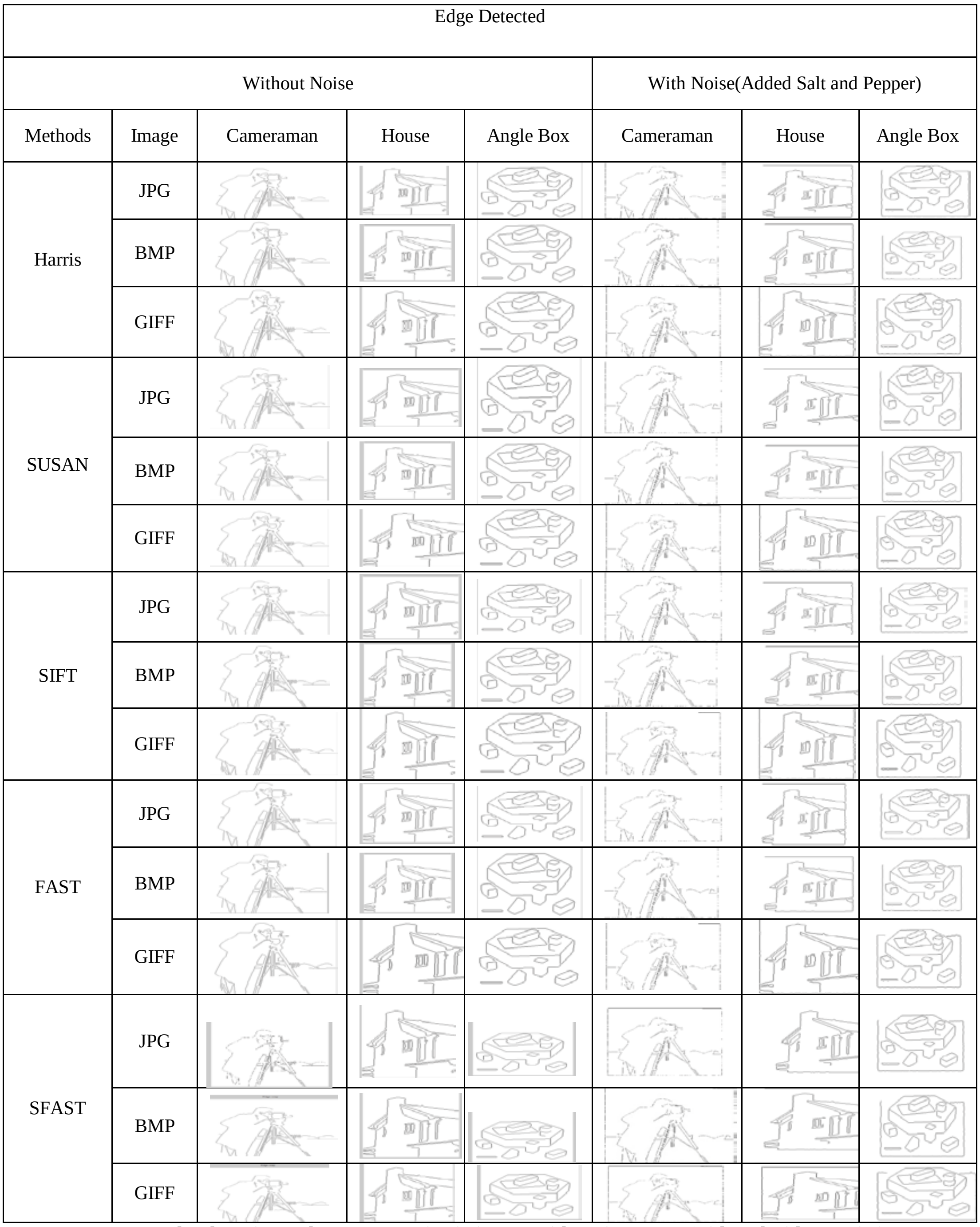


Fig 4.1: **Edge detection and curve extracting in images with various types (with and without noise)**

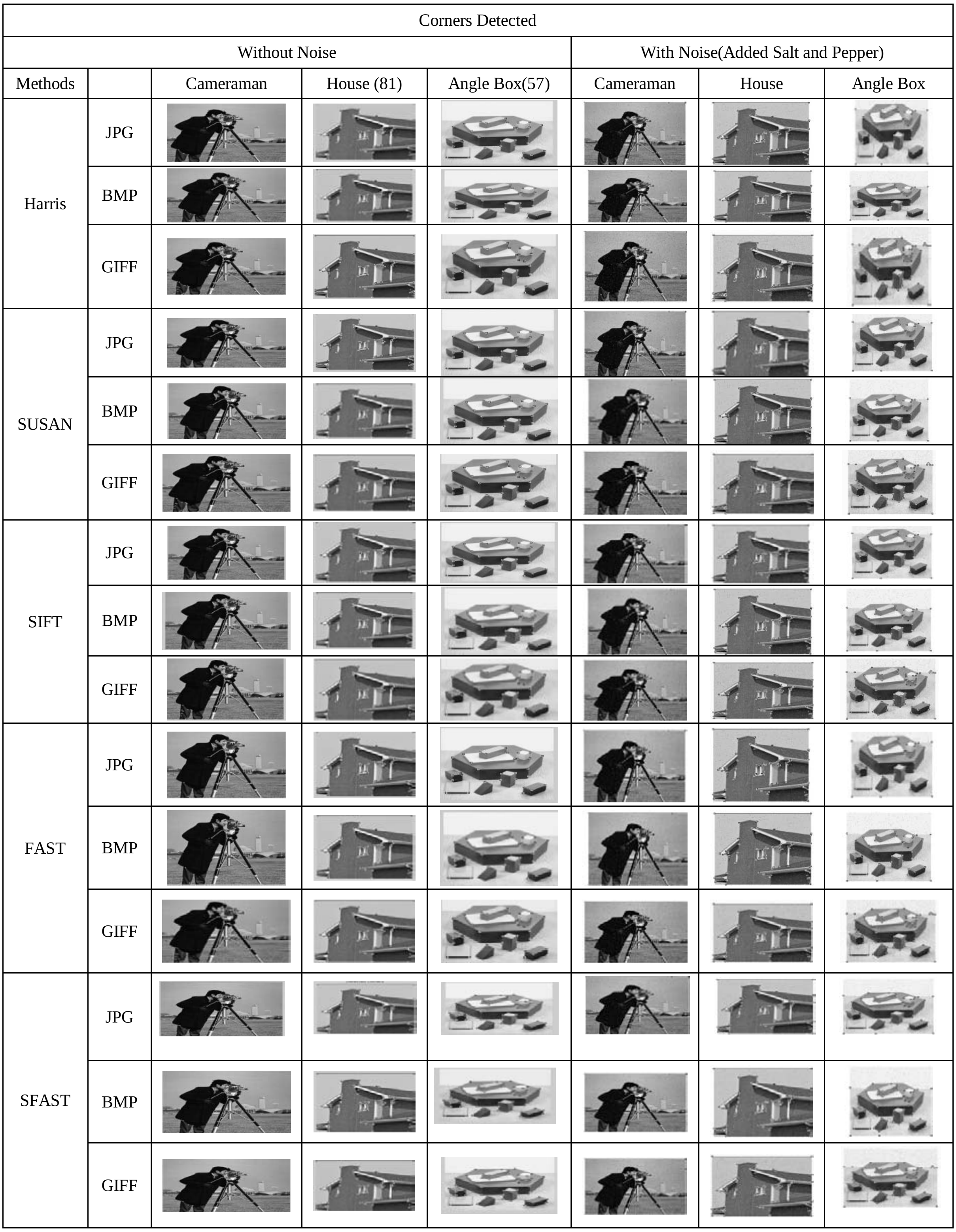


Fig 4.2: Corners detected in images with various types (with and without noise)