Journal of Mathematical Problems, Equations and Statistics

E-ISSN: 2709-9407
P-ISSN: 2709-9393
JMPES 2024; 5(1): 137-142

www.mathematicaljournal.com
Received: 15-01-2024
Accepted: 21-02-2024

K Revathi
Research Scholar, Department of Mathematics, Periyar University, Salem, Tamil Nadu, India

Dr. P Sundararajan
Associate Professor, PG & Research, Department of Mathematics, Arignar Anna Govt. Arts College, Namakkal, Tamil Nadu, India

Corresponding Author:
K Revathi
Research Scholar, Department of Mathematics, Periyar University, Salem, Tamil Nadu, India

Application of intuitionistic robust fuzzy matrix in computer vision

K Revathi and Dr. P Sundararajan

DOI: https://doi.org/10.22271/math.2024.v5.i1b.132

Abstract
Many techniques have been employed for the development of an optimum corner detection algorithm. Each effort is guided by the motivation to overcome the limitations in previous methodologies. The conventional techniques incorporate the use of linear time invariant filters. These filters recognize a corner as an abrupt change of grey scale pixel intensities. The techniques are well established and computationally efficient. Harris, SUSAN, Robert, Prewitt, canny, SOBOL, SIFT, FAST, SFAST are based on the concept of spatial differential filters utilizing local gradient. These filters process the data in a relatively short time and are computationally optimized, however, they are susceptible to noise. In this paper a new intuitionistic robust fuzzy matrix, corner-based detection method is proposed, namely Intuitionistic Robust Fuzzy Matrix Corner Detection (IRFMCD) and its performance is studied by using real images with error tolerance.

Keywords: Intuitionistic robust fuzzy matrix, computer vision, canny, sift, fast

1. Introduction

Corners in an image are contours generated because of sudden or abrupt change in any of the (multiple) characteristics at pixel level. These changes could be observed due to alteration in colour, texture, shade, or light absorption. These characteristics could further lead in estimating the orientation, size, depth, and surface features in an image. Corner detection has numerous applications in the field of robotics, medical image analysis, geographical science, pattern recognition, and military technology etc., it is often the case that these images embody high frequency noise or irrelevant data which inhibits the detection of continuous corner points, since corner itself is a composition of high frequency data. The noise generates false flags as they often mislead the algorithms for a corner.

Section 2 deals with the corner detection concepts and the various types of corner and detection methods along with their features belongs to section 3. Section 4 provides the features of the proposed IRFMCD procedure in the context of corner detection. Finally, section 5 presents the results of experimental study of IRFMCD along with the existing methods.

2. Corner Detection Techniques

Fuzzy Set theory is another technique that has been employed for corner detection. The method performs mathematical and logical reasoning based on approximations rather than crisp values. Therefore, the technique significantly reduces the complexity of problems where fixed values cannot be attained or predicted. Kim *et al*. A theoretical perception suggests that higher order fuzzy rules set would compensate for other limitations and effectively represent uncertainties. Unfortunately, the complexity of representation of model in fuzzy type-2 increases multi-fold.

The developed corner detection technique for noisy images is based on fuzzy logic. A 3x3 window mask was designed to take the grayscale values of neighborhood pixels from the input image. The grayscale values of the neighborhood pixels obtained from the mask were pre-processed prior to the fuzzy inference system.

3. Typers of Corner Detection

3.1 HARRIS operator

Harris proposed a method to solve the problems of noisy response due to a binary window function, namely Harris detector by applying the Gaussian noisy filter is defined as,

$C(x,y) = \Sigma w[I(xi,yi) - I(xi+\Delta x, yi+ \Delta y)]2$

where the image function I(.) is approximated by a Taylor expansion truncated to the first order terms. C(x, y), the auto-correlation matrix which captures the intensity structure of the local neighborhood. Finally, find the operator points as local maxima of the operator response by characterizing operator by Eigen values of C(x, y).

3.2 SUSAN operator

Smith and Brady entrenched a new approach to low level image processing, in particular, corner and operator detection and structure preserving noise reduction.

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

The SUSAN detectors are based on the minimizing of the local image region, and the noise reduction method uses the region in the smoothing the neighborhood.

3.3 Robert operator

The calculation of the gradient magnitude of an image is obtained by the partial derivatives Gx and Gy at every pixel location. The simplest way to implement the first order partial derivative is by using the Roberts cross gradient operator is given,

$Gx= f (i, j) - f (i+1, j+1)$ $Gy= f (i+1, j) - f (i,j+1)$ (1)

The above partial derivatives can be implemented by approximating them to two 2x2 masks.

3.4 Prewitt operator

The Prewitt corner detector is a much better operator than Roberts's operator. This operator having a 3 x 3 masks deals better with the effect of noise. An approach using the masks of size 3 x 3 is given below, the arrangement of pixels about the pixels [i,j].

3.5 Canny operator

Canny technique is a very important method to find corners by isolating noise from the image before finding the corners of the image, without affecting the features of the corners in the image and then applying the tendency to find the corners and the critical value for threshold. The algorithmic steps for canny corner detection technique are followed.

1. Convolve image f (r, c) with a Gaussian function to get smooth image,

   $f(r,c).f(r,c)=f(r,c)*G(r,c,6)$

2. Apply the first difference gradient operator to compute corner strength, then the corner magnitude and direction are obtained as before.
3. Apply non-maximal or critical suppression of the gradient magnitude.
4. Apply threshold of the non-maximal suppression image.

3.6 Sobel Operator

The Sobel corner detector is very much like the Prewitt corner detector. The difference between both is that the weight of the central coefficient is 2 in the Sobel operator. The partial derivatives of the Sobel operator are calculated as.

$Gx = (a_6 +2a_5+ a_4) - (a_0+2a_1+ a_2)$ & $Gy = (a_2 +2a_3+ a_4) - (a_0+2a_7+ a_6)$

3.7 SIFT

SIFT (Scale Invariant Feature Transform) corner detection is realized by extracting a distinctive invariant feature from image algorithm was proposed by Lowe (2004) [13]. 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, one can find the same corners.

3.8 FAST

FAST (Features from Accelerated Segment Test) algorithm was proposed and extended by Rosten and Tom (2006, 2010). Several feature detectors have been established and many of them are good. But when looking for a real-time application point of view, they are not fast enough. 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.

These are labeled using a straightforward implementation of the segment test criterion for n and a convenient threshold. Second, 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.

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

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 many correspondence errors if the motion is large.

3.9 SFAST

SFAST (S estimator FAST) algorithm was proposed and extended by Ravi and Muthukrishnan (2016). Edges are the key points in the corners and curve extraction is the one of the most powerful tools in the edges. The edges are generated by 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 give imperfect edges. FAST algorithm uses the least squares procedure.

The 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 do not verify 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.

4. Intuitionistic robust fuzzy matrix corner detection (IRFMCD)

In this proposed method four inputs and one output is given to the Intuitionistic robust fuzzy matrix system with the help of a canny corner detection algorithm, Called Intuitionistic robust fuzzy matrix Corner Detection (IRFMCD). The four inputs are the four pixels of the 3x3 masking window. For inputs and output, the triangular membership function is used. Two super fuzzy matrix sets are used for the input -Black & White and two super fuzzy matrices are used for the output- Corner&. Non-Corner. The IRFMCD was created to represent each variable's intensities; these sets are associated to the linguistic variables "black" and "white" for input and "corner" and "non corner" for output. The adopted membership functions for the super fuzzy matrix sets associated to the input and output are triangular. The work of this section is concerned with the development of super fuzzy matrix rules-based algorithm for the detection of image corners. For applying robust fuzzy matrix on an image to detect the corners, image preprocessing is done. Robust fuzzy matrix image processing is the collection of all approaches that understand, represent, and process the images, their segments and features as super fuzzy matrix sets.

The IRFMCD method, which extracts the corners of the image from the contours of the corner detected image. The IRFMCD corner detector works as follows.

Algorithm

- Input image.
- Guassian Smoothing.
- Grandient in x & y.
- Non-maximum suppression.
- IRFMCD conversion until n^{th} term.
- Hysteresis threshold (T) value.
- Fill small gaps in corner contours. When the gap forms a T=1.96 $\hat{\sigma}_s$.-junction, mark it as a T-corner.
- Compute curvature on the corner contours at a high scale.
- Compare T-corners to the corners found using the IRFMCD procedure and remove very close corners.

5. Experimental Result

All algorithms were implemented in MATLAB software. The performance of the various methods of corner detection is tested with different images (with and without Noise). The various corner detectors are used for the extraction of corner points and gaps of 1.5 pixel wide are filled for the detection of corner features.

The processing time for detection of corners in images is summarized in Table 1 and 2. Experimental actual images shown in Figure 1.

Table 1: Number of true/false corners detected with time taken (Without noise)

Number of Corners (without noise)											
Image	**Type of Image**	**HARRIS**	**SUSAN**	**ROBERT**	**PREWITT**	**CANNY**	**SOBEL**	**SIFT**	**FAST**	**S-FAST**	**IRFMCD**
Shapes	JPEG	51	53	55	57	59	58	60	61	63	65
		[22]	[20]	[19]	[19]	[18]	[19]	[16]	[15]	[14]	[12]
		(0.717)	(0.589)	(0.424)	(0.415)	(0.378)	(0.361)	(0.351)	(0.344)	(0.331)	(0.322)
	GIF	50	51	52	54	56	57	60	60	62	63
		[21]	[21]	[20]	[19]	[19]	[18]	[17]	[16]	[15]	[13]
		(0.700)	(0.577)	(0.411)	(0.400)	(0.398)	(0.381)	(0.371)	(0.369)	(0.351)	(0.342)
	PNG	51	52	54	57	59	58	60	61	63	65
		[22]	[21]	[19]	[19]	[18]	[19]	[16]	[15]	[14]	[12]
		(0.701)	(0.599)	(0.404)	(0.405)	(0.361)	(0.321)	(0.311)	(0.300)	(0.299)	(0.297)
	BMF	50	51	52	54	56	57	60	60	62	63
		[21]	[21]	[20]	[19]	[19]	[18]	[17]	[16]	[15]	[13]
		(0.700)	(0.577)	(0.411)	(0.400)	(0.398)	(0.381)	(0.371)	(0.369)	(0.351)	(0.342)
Apple	JPEG	30	32	33	35	37	37	38	39	40	41
		[14]	[13]	[11]	[11]	[10]	[9]	[8]	[8]	[7]	[6]
		(0.617)	(0.519)	(0.414)	(0.405)	(0.348)	(0.331)	(0.311)	(0.304)	(0.301)	(0.300)
	GIF	32	33	32	35	36	38	37	39	41	42
		[13]	[12]	[11]	[10]	[10]	[9]	[7]	[7]	[7]	[5]
		(0.707)	(0.509)	(0.404)	(0.405)	(0.308)	(0.301)	(0.301)	(0.304)	(0.301)	(0.301)
	PNG	31	34	31	34	37	37	37	39	40	41
		[14]	[14]	[12]	[11]	[11]	[10]	[9]	[8]	[7]	[6]
		(0.617)	(0.589)	(0.444)	(0.425)	(0.398)	(0.361)	(0.351)	(0.344)	(0.331)	(0.322)
	BMF	33	33	34	36	38	38	38	39	41	42
		[15]	[14]	[13]	[12]	[12]	[11]	[11]	[11]	[10]	[9]
		(0.717)	(0.589)	(0.424)	(0.415)	(0.378)	(0.361)	(0.351)	(0.344)	(0.331)	(0.322)
Cameraman	JPEG	108	109	111	113	114	116	118	119	120	121
		[24]	[24]	[23]	[22]	[21]	[20]	[19]	[19]	[18]	[17]
		(0.734)	(0.637)	(0.611)	(0.523)	(0.498)	(0.413)	(0.381)	(0.374)	(0.360)	(0.332)
	GIF	107	109	111	113	114	116	118	119	120	121
		[25]	[22]	[20]	[19]	[18]	[18]	[17]	[16]	[16]	[15]
		(0.714)	(0.623)	(0.511)	(0.423)	(0.398)	(0.363)	(0.351)	(0.344)	(0.310)	(0.212)
	PNG	107	109	111	113	114	116	118	119	120	121
		[25]	[22]	[20]	[19]	[18]	[18]	[17]	[16]	[16]	[15]
		(0.634)	(0.631)	(0.561)	(0.523)	(0.408)	(0.333)	(0.321)	(0.314)	(0.300)	(0.222)
	BMF	108	108	110	114	114	116	118	119	120	121
		[24]	[21]	[21]	[19]	[18]	[18]	[17]	[16]	[16]	[15]

		(0.614)	(0.621)	(0.521)	(0.513)	(0.428)	(0.323)	(0.311)	(0.346)	(0.301)	(0.225)
Leena Image	JPEG	101	103	106	107	109	111	113	113	115	117
		[27]	[24]	[23]	[21]	[20]	[19]	[18]	[16]	[15]	[14]
		(0.634)	(0.631)	(0.561)	(0.523)	(0.408)	(0.333)	(0.321)	(0.314)	(0.300)	(0.222)
	GIF	102	104	107	108	109	112	113	114	115	117
		[26]	[24]	[23]	[21]	[20]	[19]	[18]	[16]	[15]	[14]
		(0.631)	(0.621)	(0.551)	(0.513)	(0.404)	(0.323)	(0.320)	(0.310)	(0.245)	(0.220)
	PNG	101	104	107	108	109	112	113	114	115	117
		[25]	[24]	[23]	[21]	[20]	[19]	[18]	[16]	[15]	[14]
		(0.621)	(0.620)	(0.540)	(0.503)	(0.400)	(0.313)	(0.310)	(0.300)	(0.235)	(0.217)
	BMF	103	103	105	106	108	110	112	113	116	118
		[26]	[23]	[22]	[21]	[20]	[18]	[18]	[17]	[16]	[15]
		(0.624)	(0.621)	(0.567)	(0.513)	(0.411)	(0.303)	(0.311)	(0.304)	(0.260)	(0.222)
House	JPEG	118	119	121	122	124	126	128	129	130	131
		[24]	[24]	[23]	[22]	[21]	[20]	[19]	[19]	[18]	[17]
		(0.714)	(0.617)	(0.611)	(0.513)	(0.418)	(0.413)	(0.311)	(0.310)	(0.310)	(0.310)
	GIF	117	120	121	123	124	127	130	130	132	133
		[23]	[21]	[20]	[20]	[21]	[20]	[19]	[19]	[18]	[17]
		(0.714)	(0.617)	(0.611)	(0.513)	(0.418)	(0.413)	(0.311)	(0.310)	(0.310)	(0.310)
	PNG	118	121	122	123	124	127	130	130	132	133
		[21]	[22]	[21]	[21]	[20]	[19]	[19]	[17]	[16]	[16]
		(0.714)	(0.617)	(0.611)	(0.513)	(0.418)	(0.413)	(0.311)	(0.310)	(0.310)	(0.310)
	BMF	119	120	121	122	123	126	131	132	132	132
		[21]	[22]	[21]	[21]	[20]	[19]	[19]	[17]	[16]	[16]
		(0.724)	(0.623)	(0.614)	(0.523)	(0.428)	(0.403)	(0.341)	(0.311)	(0.306)	(0.301)

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

Table 2: Number of true/false corners detected with time taken (With noise)

Number of Corners (without noise)											
Image	**Type of Image**	**HARRIS**	**SUSAN**	**ROBERT**	**PREWITT**	**CANNY**	**SOBEL**	**SIFT**	**FAST**	**S-FAST**	**IRFMCD**
Shapes (71)	JPEG	50	51	52	54	56	57	58	60	61	63
		[26]	[25]	[21]	[20]	[20]	[19]	[18]	[17]	[16]	[15]
		(0.737)	(0.599)	(0.444)	(0.435)	(0.387)	(0.377)	(0.361)	(0.354)	(0.341)	(0.332)
	GIF	50	51	52	54	56	57	60	60	62	63
		[23]	[23]	[21]	[20]	[21]	[19]	[18]	[17]	[16]	[15]
		(0.710)	(0.587)	(0.431)	(0.420)	(0.399)	(0.391)	(0.381)	(0.379)	(0.361)	(0.352)
	PNG	51	52	54	57	59	58	60	61	63	65
		[24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[15]	[14]
		(0.701)	(0.599)	(0.404)	(0.405)	(0.361)	(0.321)	(0.311)	(0.300)	(0.299)	(0.297)
	BMF	50	51	52	54	56	57	60	60	62	63
		[23]	[22]	[21]	[19]	[19]	[18]	[17]	[16]	[15]	[13]
		(0.730)	(0. 707)	(0.411)	(0.410)	(0.378)	(0.361)	(0.361)	(0.359)	(0.351)	(0.350)
Apple (45)	JPEG	30	32	33	35	37	37	38	39	40	41
		[18]	[17]	[15]	[14]	[13]	[12]	[10]	[9]	[6]	[6]
		(0.637)	(0.579)	(0.494)	(0.485)	(0.398)	(0.381)	(0.361)	(0.354)	(0.341)	(0.330)
	GIF	32	33	32	35	36	38	37	39	41	42
		[17]	[17]	[16]	[15]	[13]	[12]	[9]	[9]	[8]	[7]
		(0.707)	(0.549)	(0.484)	(0.475)	(0.388)	(0.371)	(0.351)	(0.344)	(0.331)	(0.321)
	PNG	31	34	31	34	37	37	37	39	40	41
		[19]	[18]	[16]	[15]	[14]	[13]	[11]	[9]	[7]	[6]
		(0.697)	(0.599)	(0.494)	(0.485)	(0.398)	(0.391)	(0.381)	(0.374)	(0.361)	(0.352)
	BMF	33	33	34	36	38	38	38	39	41	42
		[16]	[15]	[14]	[13]	[12]	[11]	[11]	[11]	[10]	[9]
		(0.707)	(0.589)	(0.464)	(0.455)	(0.398)	(0.381)	(0.371)	(0.364)	(0.351)	(0.342)
Cameraman (130)	JPEG	108	109	111	113	114	116	118	119	120	121
		[30]	[28]	[27]	[27]	[26]	[25]	[24]	[23]	[20]	[19]
		(0.784)	(0.697)	(0.671)	(0.593)	(0.514)	(0.493)	(0.391)	(0.390)	(0.380)	(0.372)
	GIF	107	109	111	113	114	116	118	119	120	121
		[29]	[27]	[27]	[26]	[24]	[20]	[19]	[18]	[18]	[17]
		(0.774)	(0.673)	(0.571)	(0.473)	(0.378)	(0.373)	(0.371)	(0.364)	(0.360)	(0.252)
	PNG	107	109	111	113	114	116	118	119	120	121
		[30]	[32]	[31]	[28]	[27]	[25]	[24]	[22]	[19]	[18]
		(0.694)	(0.681)	(0.591)	(0.583)	(0.498)	(0.393)	(0.381)	(0.374)	(0.350)	(0.292)
	BMF	108	108	110	114	114	116	118	119	120	121
		[28]	[28]	[26]	[24]	[22]	[21]	[20]	[19]	[18]	[17]
		(0.674)	(0.671)	(0.571)	(0.563)	(0.478)	(0.373)	(0.361)	(0.356)	(0.351)	(0.325)
Leena Image (121)	JPEG	101	103	106	107	109	111	113	113	115	117
		[35]	[34]	[33]	[31]	[30]	[29]	[28]	[26]	[25]	[22]
		(0.694)	(0.691)	(0.591)	(0.583)	(0.498)	(0.393)	(0.391)	(0.384)	(0.350)	(0.322)
	GIF	102	104	107	108	109	112	113	114	115	117

		[36]	[34]	[33]	[31]	[30]	[29]	[28]	[26]	[15]	[14]
		(0.681)	(0.621)	(0.551)	(0.513)	(0.404)	(0.323)	(0.320)	(0.310)	(0.245)	(0.220)
	PNG	101	104	107	108	109	112	113	114	115	117
		[35]	[34]	[33]	[31]	[30]	[29]	[27]	[26]	[24]	[23]
		(0.671)	(0.670)	(0.570)	(0.563)	(0.480)	(0.393)	(0.380)	(0.370)	(0.295)	(0.287)
	BMF	103	103	105	106	108	110	112	113	116	118
		[36]	[33]	[32]	[31]	[30]	[28]	[28]	[27]	[26]	[25]
		(0.664)	(0.661)	(0.597)	(0.583)	(0.491)	(0.393)	(0.381)	(0.374)	(0.290)	(0.282)
House (141)	JPEG	118	119	121	122	124	126	128	129	130	131
		[34]	[34]	[33]	[32]	[31]	[30]	[29]	[28]	[28]	[27]
		(0.774)	(0.667)	(0.651)	(0.573)	(0.488)	(0.473)	(0.391)	(0.380)	(0.370)	(0.360)
	GIF	117	120	121	123	124	127	130	130	132	133
		[33]	[31]	[30]	[30]	[31]	[30]	[29]	[29]	[28]	[27]
		(0.764)	(0.697)	(0.681)	(0.583)	(0.498)	(0.48)	(0.391)	(0.382)	(0.360)	(0.360)
	PNG	118	121	122	123	124	127	130	130	132	133
		[31]	[32]	[31]	[31]	[30]	[29]	[29]	[27]	[26]	[26]
		(0.754)	(0.667)	(0.651)	(0.563)	(0.468)	(0.453)	(0.371)	(0.360)	(0.350)	(0.340)
	BMF	119	120	121	122	123	126	131	132	132	132
		[33]	[32]	[32]	[31]	[30]	[29]	[29]	[27]	[26]	[26]
		(0.704)	(0.653)	(0.644)	(0.553)	(0.448)	(0.433)	(0.351)	(0.351)	(0.336)	(0.331)

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

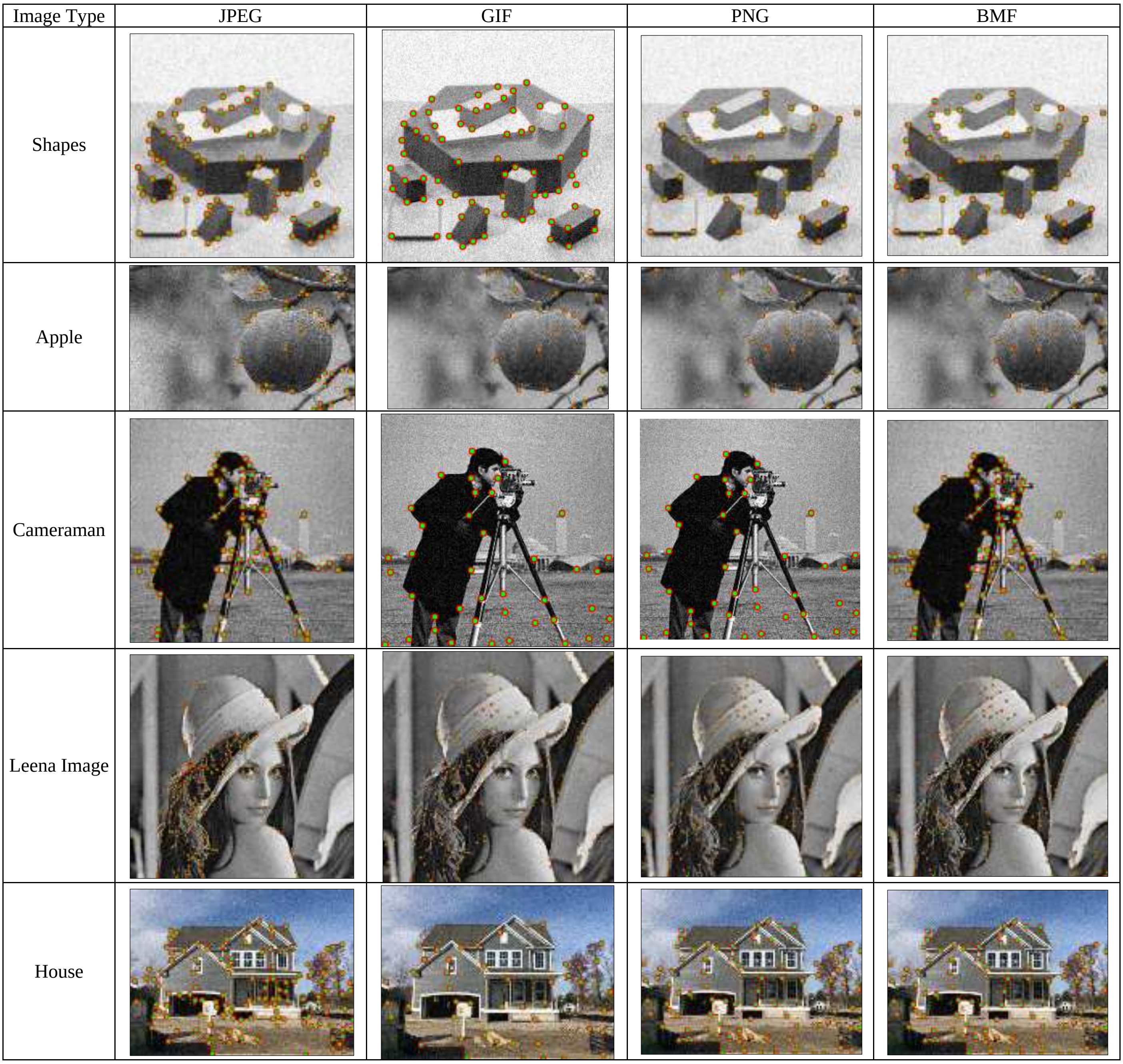

Fig 1: Corner Detected under IRFMCD (With noise)

Conclusion

It is observed from table 1 and 2, the performance of the proposed IRFMCD is better than the other corner detection methods. It is noted that corner detection and curve extraction timings are considerably low in IRFMCD procedure since the proposed method works with 36 pixels.

The above data Simulated in TLS MATLAB toll box. It is randomly generated data based on the sample size. It is also to fix the threshold values is 2 in the toolbox and add 2% of salt and pepper Gaussian noise in the image.

Table 1 and 2 indicate that the proposed IRFMCD procedure detects a greater number of true corners and a smaller number of false corners when compared with the others existing models. Note that the actual number of corners are 71, 45, 130, 121 and 141 for the image's shape, apple, cameraman, Lena, and house respectively. It is concluded that IRFMCD procedure outperforms Harris, SUSAN, Rober, Prewitt, Canny, SOBEL, SIFT, FAST and SFAST by considering the processing time, curve and edge extracted and number of true corners detected.

References

1. Adak AK, Bhowmik M, Pal M. Some properties of generalized intuitionistic fuzzy nilpotent matrices over distributive lattice. Fuzzy Information and Engineering. 2012;4(4):371-387.
2. Atanassov K. Intuitionistic fuzzy sets VII ITKR's session. Sofia; c1983 Jun.
3. Atanassov K, Gargov G. Elements of intuitionistic fuzzy logic. Part-I. Fuzzy Sets and Systems. 1998;95(1):39-52.
4. Dengfeng L, Chuntian C. New similarity measures of intuitionistic fuzzy sets and application to pattern recognitions. Pattern Recognition Letters. 2002;23(1-3):221-225.
5. Deschrijver G, Kerre EE. On the relationship between some extensions of fuzzy set theory. Fuzzy Sets and Systems. 2003;133(2):227-235.
6. Deschrijver G, Kerre EE. On the position of intuitionistic fuzzy set theory in the framework of theories modelling imprecision. Information Sciences. 2007;177(8):1860-1866.
7. Ravi J, *et al*. A new robust distance machine learning approach to edge and corner detection: RD-FAST. Journal of Statistics and Mathematical Engineering (JSME). 2018;4(3):31-44.
8. Ravi J, *et al*. SFAST: A new robust regression based combined edges and corner detection. International Journal of Advanced Research in Computer Science, 2017, 8(7).
9. Ravi J. A robust measure of pairwise distance estimation approach: RD-RANSAC. International Journal of Statistics and Applied Mathematics. 2017;2(2):31-34.
10. Liu H-W. New similarity measures between intuitionistic fuzzy sets and between elements. Mathematical and Computer Modelling. 2005;42(1):61-70.
11. Zadeh LA. The concept of a linguistic variable and its application to approximate reasoning-III. Information Sciences. 1975;9(1):43-80.
12. Zeng W, Li D, Yin Q. Distance and similarity measures of Pythagorean fuzzy sets and their applications to multiple criteria group decision making. International Journal of Intelligent Systems. 2018;33(11):2236-2254.
13. Lowe DG. Distinctive image features from scale-invariant keypoints. International journal of computer vision. 2004 Nov;60:91-110.