Evaluation Of Image Detection Techniques

U.I. Bature
Department of Computer and Communications Engineering
Abubakar Tafawa Balewa University, P.M.B. 0248 ATBU Bauchi, Nigeria
4070usa@gmail.com,

Murtala A. B.
Department of Computer and Communications Engineering
Abubakar Tafawa Balewa University, P.M.B. 0248 ATBU Bauchi, Nigeria
limaminkobi@yahoo.com,

A. Y. Nasir
Department of Computer and Communications Engineering
Abubakar Tafawa Balewa University, P.M.B. 0248 ATBU Bauchi, Nigeria
aynasir@gmail.com

***Abstract*—Edge detection techniques are mathematical methods of detecting and localizing discontinuities in an image. Edge detection is a well-developed field within image processing and image analysis because it helps in reducing some significant amount of data and cleans out irrelevant information. Therefore it's pertinent to evaluate the differences, advantages and disadvantages of these edge detection techniques. In this paper we presented the evaluation of the conventional and commonly used edge detection techniques, that is, first and second order derivatives popularly known in image processing as gradient-based and Laplacian-based respectively. These edge detection techniques algorithms were developed using Matlab 7.7.0 version. The Prewitt, Sobel and Roberts operators are simple and quick to compute while the canny operator exhibits complexity in computations, implementations, and more execution time but since it employs probability method for error rate finding, well localization of edge points and response. Thus, the canny edge detection algorithm produce better edge detection even in noisy situations and signal to noise ratio (SNR) enhancement, and it is capable of finding best contours and unbroken edges.**

Keywords; Digital Image Processing, Edge Detectors, Evaluation

I. Introduction

Edges are boundaries between different textures, and points where image brightness varies swiftly are usually arranged into a set of line segments called edges. Edge can also be defined as discontinuities in image intensity from one pixel to another [1]. Edge detection is a mathematical method of detecting and localizing discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene [2], [3], [4]. Edge detection is a well-developed field within image processing in which region boundaries and edges are strongly related [5].
Segmentation is a process of abridging, editing and altering the representation of an image into a desired or targeted image that is more expressive and easy to evaluate. Images are subdivided into their integral components where discontinuities in their pixel intensities like edges are clearly shown, Segmentation tends to provide meaningful discontinuities in intensity values, this leads to the popularity of this method in image processing. Edge detection is a well-developed field within image processing in which region boundaries and edges are strongly related and frequently a sharp modification in intensity at the region boundaries exist [5]. Discontinuities in intensity values are better detected by the use of first and second order derivatives which are popularly known in image processing as gradient and Laplacian respectively.

II. Overview on Edge Detector

Variations in the properties of an image brightness is better expressed by detecting their changes in edge, this helps in revealing an important details or events that are useful for further processes. Applying an edge detection technique to an image may reveal or preserve the most important structural details of that image and subsequently reduces or filters out the less relevant ones. Edge detection is difficult in noisy images, since both the noise and the edges contain high frequency contents. Operators used on noisy images are typically larger in scope, so they can average enough data to discount localized noisy pixels [3]. It can be expressed under general assumptions for an image model that breaks in image brightness are probably to link with [6];

- discontinuities in depth,
- discontinuities in surface orientation,
- changes in material properties,
- variations in scene illumination,

The first order derivatives which are known as classical methods these are gradient operators (2D filters) like the Sobel, Prewitt and Robert operators, and these are designed to be sensitive to large gradient (discontinuities) in digital image brightness while zero values are returns in uniform regions. The Laplacian method searches for zero crossing in the second derivative of the image to find edges, and edge detection has applications like object recognition, motion analysis, and pattern recognition [2].

III. Methods of Edge Detection

Processes to perform edge detections are many, as earlier mentioned they are grouped into two categories; the gradient operators and Laplacian method and they are as follows:

A. *Edge Detection via the Gradient Operators*

Gradient can be defined as a vector which has certain magnitude and direction and this is rather described as follows [7]:

$$\Delta f = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix}$$

$$magn(\nabla f) = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} = \sqrt{M_x^2 + M_y^2} \qquad (1)$$

$$dir(\nabla f) = tan^{-1}\left(\frac{M_y}{M_x}\right) \qquad (2)$$

Alternatively the magnitude of gradient is represented by:

$$magn(\nabla f) \approx |M_y| + |M_x| \qquad (3)$$

Gradient has the following properties:

- Magnitude of gradient provides information about the strength of the edge.
- Gradient direction is always perpendicular to the direction of the edge. In order words, edge direction rotated with respect to the gradient direction by 90 degrees.

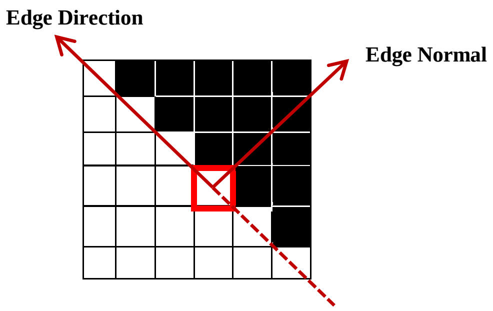

Fig. 1. *Description of normal and direction lines of edge*

1) *Robert Cross Operator:*

The Roberts Cross operator is a 2×2 convolution kernel as presented in Fig. 2, this operator executes fast and simple to compute, 2-D spatial gradient measurement on an image. One kernel is simply the other rotated by 90° [3]. The operator is designed to respond extremely to edges running at 45 degrees to the pixel of consideration one mask for each of the two perpendicular alignments. Values of the pixel at each point obtained after computation denotes the estimated absolute magnitude of the spatial gradient of the grey input image at that particular point.

$$M_x = \begin{bmatrix} +1 & 0 \\ 0 & -1 \end{bmatrix} \qquad M_y = \begin{bmatrix} 0 & +1 \\ -1 & 0 \end{bmatrix}$$

Fig. 2. *Roberts Cross convolution Kernels*

The gradient magnitude is best represented by (1) and (3) while the direction (Angle of orientation of the edge) is given by [3]:

$$dir(\nabla f) = tan^{-1}\left(\frac{M_y}{M_x}\right) - \frac{3\pi}{4} \qquad (4)$$

Robert Cross operator is very sensitive to noise due to the use of such a small mask of 2×2 orientations and produces very weak responses to actual edges except they are very shrill.

2) *Sobel Edge Detector:*

Sobel operator is 3×3 neighborhoods convolution kernels as shown in Fig. 3, one kernel for each of the vertical and horizontal orientation (M_y and M_x), one mask is simply the other rotated by 90°. It is usually used to detect the approximate absolute gradient magnitude at a particular point of interest in the grey input image. Sobel operator performs a 2-D spatial gradient measurement on a grey image and emphasizes on those regions with high spatial gradient. Sobel operator unlike Robert cross operator, it is less sensitive to noise and it is intended to react maximally to the edges running horizontally and vertically relative to the pixel grid. Each kernel can be apply separately to the input image to produce the corresponding measurements of the gradient components along particular orientation which can be combined to find the absolute magnitude of the gradient at every point.

$$M_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \qquad M_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Fig. 3. *Sobel Operator Kernels*

The Sobel gradient magnitude is computed using (1) or (3) while the direction is given by (2).

3) Prewitt Edge Detector:

The Prewitt operator is similar to the Sobel operator and it is used for detecting vertical and horizontal edges in images [8]. Prewitt operator is restricted to eight probable orientations and most direct orientation estimates are not exactly accurate [2]. Prewitt operator is 3×3 neighborhoods convolution kernels as shown in Fig. 4, one kernel for each of the vertical and horizontal orientation (M_y and M_x), one mask is simply the other rotated by 90°.

$M_x =$

-1	0	+1
-1	0	+1
-1	0	+1

$M_y =$

+1	+1	+1
0	0	0
-1	-1	-1

Fig. 4. *Prewitt Operator Kernels*

B. Second Derivative Operators

The first order derivatives or the classical method results in detection of too many edge points when a threshold is assumed. Therefore, a better edge detection approach would be the second order derivative which would find only the points that have the local maxima in gradient values and regarded them as edge points. Consequently, at first derivative there will be a peak at edge points while at the second derivative there will be a zero crossing.

Consider the properties of second derivative, hence, edge points may be detected on finding the zero crossings in the derivative of the image intensity. Laplacian operator falls under this second order derivatives.

1) Laplacian of Gaussian:

Laplacian is a two dimensional simplex isotropic operator which is defined for a function input image $f(x,y)$ for two variables (Rosenfeld and Kak [1982]), as shown in (5) [9]. Those who continued on this way were Berzins (1984), Shah, Sood and Jain (1986), Huertas and Medioni (1986) [10]. Edge points detected by considering a zero crossings of 2nd order derivatives of image intensity are always very sensitive to noise, as such, it is required that noise must first be filter out before the edge detection. Thus, the combination of Laplacian and Gaussian filters for edge detection popularly known as Laplacian of Gaussian (LoG) operator was invented in 1980 by Marr and Hildreth. The Laplacian is often applied to an image that has first been smoothed with something approximating a Gaussian Smoothing filter in order to reduce its sensitivity to noise [3], [10].

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (5)$$

Laplacian is a linear operator since derivatives of any order are linear operations and of course the input image is represented as a set of discrete pixels. Hence a discrete convolution mask that can approximate the 2nd derivatives is needed. Fig. 6 is always used as discrete approximations to the kernels for the Laplacian operation [8] [9] [11].

(a)

0	-1	0
-1	4	-1
0	-1	0

(b)

-1	-1	-1
-1	8	-1
-1	-1	-1

(c)

0	1	0
1	-4	1
0	1	0

(d)

1	1	1
1	-8	1
1	1	1

(c) and (d) are complementary of (a) and (b)

-1	2	-1
2	-4	2
-1	2	-1

Fig. 5. *Laplacian Kernels*

The LoG as earlier mentioned is a combination of Laplacian and Gaussian filters. Since convolution is associative, therefore, hybrid of these two filters convolve with an image would certainly produce a desired result. The filter is shown as [7]:

$$G(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (6)$$

(σ Determines the degree of smoothing, mask size increases with σ)

A discrete kernel which shows an approximation of a Gaussian function (for a Gaussian σ = 1.4) is shown in Fig. 6 [1].

0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	4	5	3	0	3	5	4	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	4	5	3	0	3	5	4	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

Fig. 6. *LoG discrete approximation with Gaussian* $\sigma = 1.4$

2) Canny Edge Detection

Canny edge operator was designed by John canny for his Master's thesis at MIT in 1983 [12]. John canny edge detector is the first derivative of a Gaussian and closely estimates the operator that optimizes the product of signal-to-noise ratio and localization. The Canny edge detector is widely considered to be the standard edge detection method in the industry [2]. Canny operator smoothest the input image with a Gaussian filters to enhances the desired result, and also estimates gradient magnitude direction at various pixels. John Canny developed an algorithm based on optimization and he considered edge detection as a signal processing optimization issues. The algorithm is summarized by the following steps:

- First, smooth the input image with a Gaussian function, this will filter out noise in the original input image, hence it will ease edge localization and detection. This is better represented in the following equations [7]:

Compute f_x and f_y

$$f_x = \frac{\partial}{\partial x}(f * G) = f * \frac{\partial}{\partial x} G = f * G_x$$

$$f_y = \frac{\partial}{\partial y}(f * G) = f * \frac{\partial}{\partial y} G = f * G_y$$

$$G_x(x, y) = \frac{-x}{\sigma^2} G(x, y) \quad (7)$$

$$G_y(x, y) = \frac{-y}{\sigma^2} G(x, y) \quad (8)$$

Where $G(x, y)$ is the Gaussian function, $G_x(x, y)$ is the derivative $G(x, y)$ with respect to x and $G_y(x, y)$ is the derivative $G(x, y)$ with respect to y.

The larger the width of the Gaussian mask, the lower is the detector's sensitivity to noise. The localization error in the detected edges also increases slightly as the Gaussian width is increased [3].

- John canny sets the second criterion after smoothing the image with Gaussian function by finding the edge strength (gradient magnitude) at every pixels, this will displays changes in the image intensity along the x-direction and y-direction and also this will indicates the presence of edges.

Consequently, edge points should be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum [2].

$$magn(i, j) = \sqrt{f_x^2 + f_y^2} \quad (9)$$

- The third criterion is the application of non-maxima suppression; John Canny realized that the first two criterions were not bold enough to entirely abolish the prospect of multiple responses to an edge. John canny sets this as to have only a response to a single edge. That is, only local maxima should be considered as edges [5].

Edges give rise to ridges in the gradient magnitude image, to find edge points, local maxima of the gradient magnitude must be found, broad ridges must be diluted or thinned and this permits only magnitudes at the points of greatest local change to remain and finally values found along the direction of the gradient which are not considered to be among the peak values of ridge are suppressed.

- The last criterion is the application of hysteresis thresholding or edge linking. Hysteresis is used as a means of eliminating streaking [13]. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold [3].

John canny sets a tracking process which demonstrates hysteresis governed by two thresholds. Since, the result of non-maxima suppression is bound to comprise of local maxima formed by noise. On setting a low threshold, some noise maxima will definitely be accepted whereas setting high threshold might cause the true maxima to be overlooked.

Therefore, thresholds T1 and T2 with a standard that T1 is greater than T2 (T1>T2) is deliberated. Tracking start at a point on a ridge that is higher than threshold T1 and it continues in both directions available from that particular point until the altitude of the ridge drops below T2. This hysteresis aids in ensuring that noisy edges are not shattered up into multiple edge fragments. Hence, this algorithm performs edge linking as by product of double-thresholding [7].

IV. Evaluation of Various Edge Detection Operators

This segment displays results of edge detection techniques performed on various images and this will surely guide us on easy evaluation of these techniques. Various edge detection techniques were performed on various images like Robert Cross edge detection, Sobel operator, Prewitt operator, Laplacian of Gaussian (LoG) edge detector and canny edge detector.

Fig. 7. *Image used for edge detection evaluation (Treeedges.jpg)*

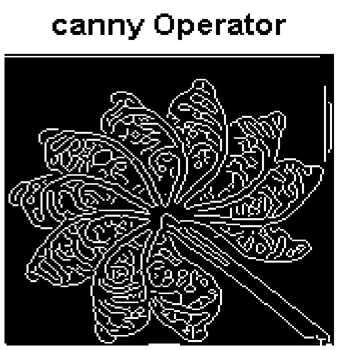

Fig. 8. *Result of canny edge detection on Fig. 7. This exhibits the best results*

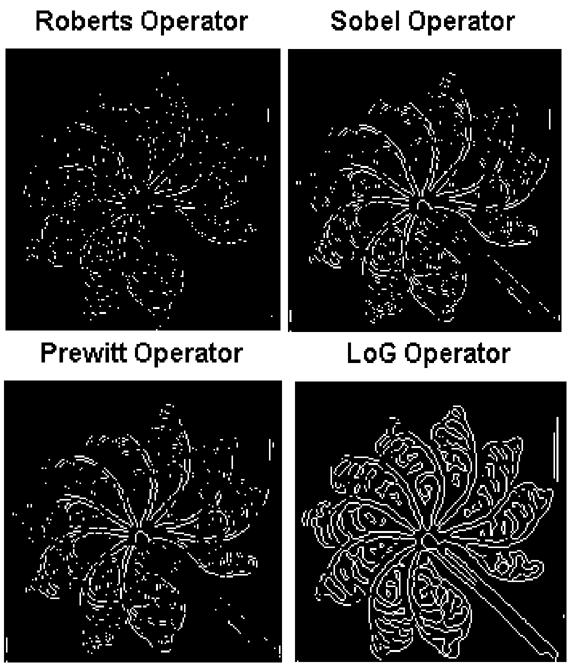

Fig. 9. *Results of various edge detection techniques on Fig. 7*

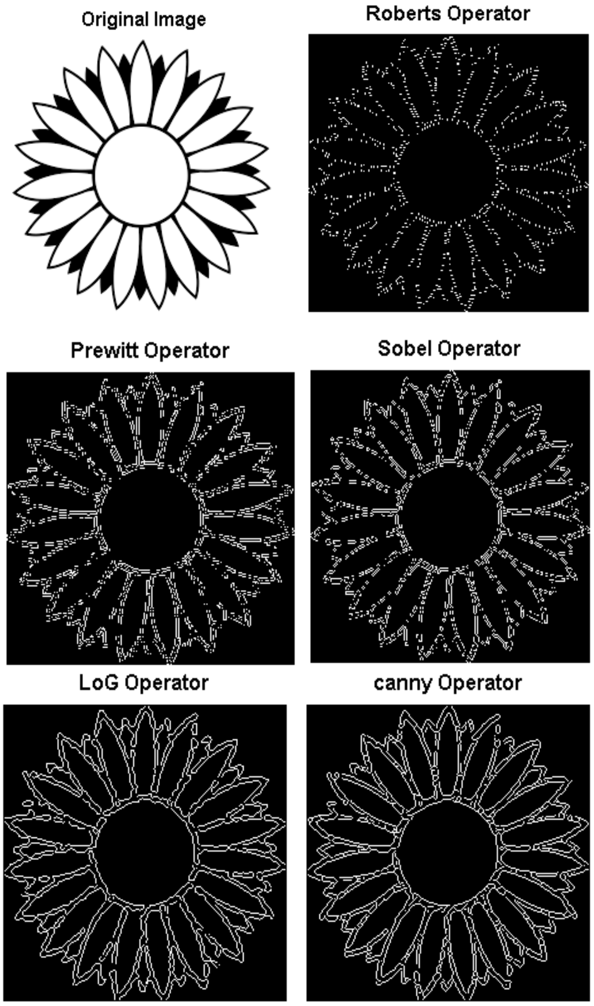

Fig. 10. *Evaluation of various edge Detection Techniques on the Original Image*

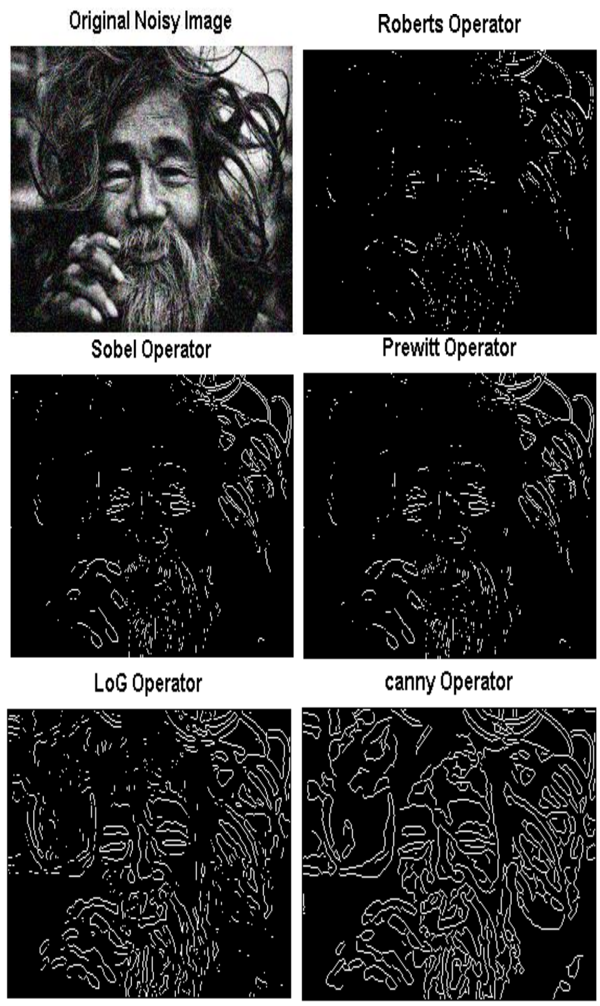

Fig. 11. *Evaluation of various edge Detection Techniques on the Noisy Image*

V. Discussion on Results

Edges are boundaries between different textures and also discontinuities in image intensity from one pixel to another. Therefore edge detection is of paramount important in computer vision, image processing, image pattern recognition and image analysis, its necessary to select edge detector that will produce desired results, edge detectors that will point out true boundaries, since this lead to the capture of an important details or events and changes in properties.

Hence, it is necessary to evaluate the performance of these edge detection techniques in order to point out the advantages and disadvantages of these methods and which method can best fit your proposed application, table 1 shows the summary of these advantages and disadvantages of these techniques [3], [4].

Table I: Summary of Advantages and Disadvantages of Various Edge Detection Techniques

Edge detectors (Operator)	Evaluation of Edge Detectors	
	Advantages	***Disadvantages***
Classical methods (Sobel, Roberts, Prewitt, …)	Simplicity and quick to compute, detection of region of high spatial gradient which often correspond to edges and orientations	Sensitive to noise and erroneous results
Zero Crossing (Laplacian of Gaussian (LoG), Marr-Hildreth)	Detects exact areas of edges, testing wider area around the pixel, detects features	Strongly influenced by the size of the Gaussian kernel used for the smoothing stage and orientation of edges often omitted, glitches at the corners, curves and areas where the grey intensity level functions varies
Gaussian (Canny, Deriche, Shen-Castan)	Employs probability method for error rate finding, well localization of edge points and response. Better edge detection even in noisy situations and signal to noise ratio enhancement, capable of finding best contours and unbroken edges	Complex computations and implementations, more execution time and false zero crossing

VI. Conclusions

The product of edge detection system yields a gray scale image that had bright intensities for the strong edges, less bright intensities for the weaker edges, and of course black for the product with no edges. It has been noticed that the gradient based edge detectors have great sensitivity to noise also the size and coefficients of the filter are fixed and cannot be adjusted for a particular input image. The Laplacian based technique of the canny relies solely on varying parameters, and this allows adaptive edge detection which provides the most robust solution.

Lastly, this paper is aimed at providing academics that are new in this field with a clear understanding of this idea of edge detection, since edge detector performance measure and approaches of evaluation provides the most effective ways of understanding the success of these developed models.

VII. Acknowledgment

The authors are really thankful to Dr. Syed Abd. Rahman Al-Attas and Dr Musa Mohd Mokji all of Computer Vision, Video and Image Processing Research Group (CvviP) Universiti Teknologi Malaysia (UTM) and Department of Computer and Communications Engineering, Abubakar Tafawa Balewa University (ATBU) Bauchi, Nigeria for the provision of encouraging facilities, materials and enough time that make this work into completion.

VIII. References

[1] Mamta Juneja, Parvinder Singh Sandhu. “Performance Evaluation of Edge Detection Techniques for Images in Spatial Domain” International Journal of Computer Theory and Engineering, Vol. 1, No. 5, December, 2009

[2] M. Sushma sri, m. Narayana. “Edge detection by using lookup table” (ijret), volume: 02 issue: 10, oct-2013.

[3] R.Maini, H. Aggarwal. “Study and comparison of various image edge detection techniques,” International Journal of Image processing (IJIP), volume (3), issue (1) 2009.

[4] Ireyuwa. E. Igbinosa. “Comparison of edge Detection technique in image processing techniques” International Journal of Information Technology and Electrical Engineering (ITEE), Volume 2, Issue 1, February 2013.

[5] Ms. Pawar P. H, Prof. Patil R. P. “Image edge detection techniques using MATLAB Simulink” (IJERT) Vol. 3 Issue 6, June, 2014.

[6] H.G. Barrow and J.M. Tenenbaum (1981) "Interpreting line drawings as three-dimensional surfaces", Artificial Intelligence, vol 17, issues 1-3, pages 75-116.

[7] Trucco, *“Edge detection”* Chapt 4 AND Jain et al., Chapt 5.

[8] R. C. Gonzalez and R. E. Woods. “Digital image processing”. 2nd ed. Prentice Hall, 2002.

[9] R. C. Gonzalez and R. E. Woods. “Digital image processing”. 3rd ed. Pearson International Edition, Prentice Hall, 2008.

[10] F. M. Abubakar, “Study of image segmentation by using edge detection techniques”.

International Journal of Engineering Research & Technology (IJERT), Vol. 1 Issue 9, November, 2012.

[11] Musa Mohd Mokji. “MEL 1233: Digital image processing”. Image Enhancement: Neighborhood Processing. University of Malaysia (UTM), 2014.

[12] P. Sharma et al. “Different Techniques of Edge Detection in Digital Image Processing” International Journal of Engineering Research and Applications (IJERA), pp.458-461, Vol. 3, Issue 3, May-Jun 2013.

[13] J. F. Canny. “A computational approach to edge detection”. IEEE Transactions in pattern analysis and machine intelligence vol. 8 pp. 679-698, 1986.