International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 3, March 2014)

An Improved Approach for Digital Image Edge Detection

Mahbubun Nahar[1], Md. Sujan Ali[2]

[1]MS Student, [2]Assistant Professor, Department of Computer Science and Engineering, Jatiya Kabi Kazi Nazrul Islam University, Trishal, Mymensingh,Bangladesh,
E-mail:mabuba.knu@gmail.com,msujanali@gmail.com

***Abstract*— Before objects detection and image segmentation edge detection is the preliminary and major step. The problems of edge detection are: false edge detection, missing true edges, producing thin or thick lines, noise removal etc. There are many edge detection algorithms have been developed for detecting the true edges from an image. These algorithms use different types of masks such as Robert, Prewitt, Sobel, Laplacian masks etc. Though the algorithms have some advantages and disadvantages, some of these perform better than others for specific regions and specific type of images. This paper proposed a new algorithm which works with a new mask. By comparing with others the proposed algorithm performs better.**

***Keywords*—Edge detection, false edges, true edges, image segmentation, object detection, edge detection algorithms, filters, masks, proposed algorithm.**

I. INTRODUCTION

In computer vision, edge detection is one of the most commonly used techniques in digital image processing [1]. Edges are sets of connected edge pixels, where the edge pixels are pixels at which the intensity of an image function changes abruptly [2]. Edge detection is a process which attempts to capture the significant properties of objects in the image. These properties include discontinuities in the photometrical, geometrical and physical characteristics of objects. Such information give rise to variations in the grey level image; the most commonly used variations are discontinuities (step edges), local extrema (line edges), junction [3]. There are an extremely large number of edge detection operators available, each designed to be sensitive to certain types of edges. Variables involved in the selection of an edge detection operator include Edge orientation, Noise environment and Edge structure. The geometry of the operator determines a characteristic direction in which it is most sensitive to edges. Operators can be optimized to look for horizontal, vertical, or diagonal edges. Edge detection is difficult in noisy images, since both the noise and the edges contain high frequency content. Attempts to reduce the noise result in blurred and distorted edges. Operators used on noisy images are typically larger in scope, so they can average enough data to discount localized noisy pixels. This results in less accurate localization of the detected edges. Not all edges involve a step change in intensity. The operator needs to be chosen to be responsive to such a gradual change in those cases. So, there are problems of false edge detection, missing true edges, edge localization, high computational time and problems due to noise etc. Here our goal is to analyse the performance of the various edge detection techniques in different conditions and propose a new way that will perform better than other methods.

II. EXISTING ALGORITHMS AND MASKS

There are many ways are exist to perform edge detection. However, the majority of different methods may be grouped into two categories:

A. Gradient Based Edge Detection:

The gradient method detects the edges by looking for the maximum and minimum in the first order derivative of the image. In a word this type of technique works with first derivative. For instance, using Sobel, Prewitt, and Robert's operator. For a continuous image $f(x, y)$, where x and y are the row and column coordinates respectively, we typically consider the two directional derivatives $\partial_x f(x, y)$ and $\partial_y f(x, y)$. The first derivative of gray level is positive at the beginning of the ramp edge and at points on the ramp; zero in the areas of constant gray levels, [2] and otherwise negative. The presence of an edge in the image is detected by the magnitude of the first derivative. The gradient magnitude is defined as

$$mag\Delta f(x,y) = \sqrt{\left(\partial_x f(x,y)^2 + \partial_y f(x,y)^2 \right)} \qquad (1)$$

In an image edge pixels have higher intensity values than its surrounding pixels. If the value of the gradient exceeds some threshold value then the pixel position is identified as an edge position. The convention for 3 by 3 regions to denote image points of an input image is shown in Fig.1.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 3, March 2014)

Z_1	Z_2	Z_3
Z_4	Z_5	Z_6
Z_7	Z_8	Z_9

Fig. 1. A 3 by 3 region of an image

Where Z_5= f(x, y), Z_1= f(x-1, y-1), Z_2= f(x-1, y),Z_3= f(x-1, y+1), Z_4= f(x, y-1), Z_6 = f(x, y+1), Z_7=f(x+1, y-1), Z_8= f(x+1,y), Z_9= f(x+1, y+1)

1) Robert edge detection: The mask of Robert operator is one of the earliest attempts to use 2-D masks with diagonal preference [2]. It is very simple in computation and swiftly measures the gradient components of an image. It thus highlights regions of high spatial frequency which often correspond to edges. In its most common usage, the input to the operator is a greyscale image, as is the output. Pixel values at each point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point [1]. The Robert masks are shown in Fig. 2. The Robert operator is given by the equations [2], and [3]:

$$\partial_x f(x,y) = Z_9 - Z_5 \quad (2)$$

$$\partial_y f(x,y) = Z_8 - Z_6 \quad (3)$$

(a)

0	-1
+1	0

(b)

-1	0
0	+1

Fig. 2. a) Robert mask for vertical direction b) Robert mask for horizontal direction

2) Sobel edge detection: The Sobel operator consists of a pair of 3 by 3 convolution kernels as shown in Fig. 3. One kernel is simply the other rotated by 90^0[6]. The Sobel operator is given by the equations [4], [5]:

$$\partial_x f(x,y) = (Z_7 + 2Z_8 + Z_9) - (Z_1 + 2Z_2 + Z_3) \quad (4)$$

$$\partial_y f(x,y) = (Z_3 + 2Z_6 + Z_9) - (Z_1 + 2Z_4 + Z_7) \quad (5)$$

Where Z1 to Z9 are the pixel values of the image region shown in Fig. 1.

(a)

-1	0	+1
-2	0	+2
-1	0	+1

(b)

-1	-2	-1
0	0	0
+1	+2	+1

Fig. 3. a) Sobel mask for vertical direction b) Sobel mask for horizontal direction

To produce the separate measurements of the gradient component horizontal and vertical direction the kernels are used separately with the input image. Then the outputs are taking together to generate the absolute magnitude of the gradient at each point and the orientation of that gradient [7].

3) Prewitt edge detection: The magnitude of an edge can measure efficiently by the Prewitt operator. Prewitt operator [2] is very similar to Sobel operator and is used for detecting vertical and horizontal edges in images. The Robert masks are shown in Fig. 4. Prewitt operator is given by the equations [6], and [7]:

(a)

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

(b)

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

Fig. 4. a) Prewitt mask for vertical direction b) Prewitt mask for horizontal direction

$$\partial_x f(x,y) = (Z_7 + Z_8 + Z_9) - (Z_1 + Z_2 + Z_3) \quad (6)$$

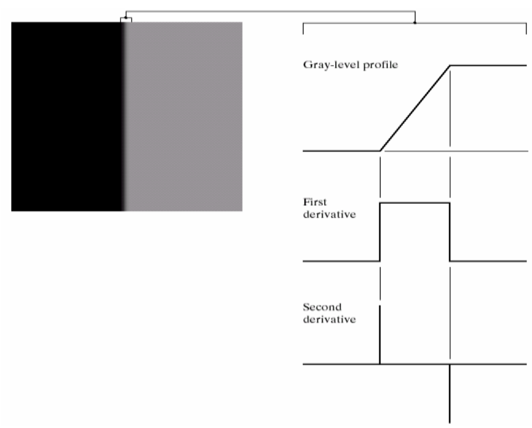
$$\partial_y f(x,y) = (Z_3 + Z_6 + Z_9) - (Z_1 + Z_4 + Z_7) \quad (7)$$

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 3, March 2014)

B. Laplacian Based Edge Detection

Laplacian method works on zero crossings in the second order derivative of the image to detect edges. An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location. Canny edge detector and Laplacian of Gaussian (LoG) etc. are the examples of Laplacian based method.

1) Laplacian of Gaussian (LoG): The edge points of an image can be detected by finding the zero crossings of the second derivative of the image intensity. The idea is illustrated in Fig.5 [2]. However, calculating the 2nd derivative of image intensity is very sensitive to noise. Before edge detection, this noise should be filtered out. The "Laplacian of Gaussian" is used to achieve this. This method combines Gaussian filtering with the Laplacian for edge detection [4]. It is sometimes called Marr-Hildreth edge detector or Mexican hat operator.



(a) (b)

Fig. 5. (a) Two regions separated by a vertical edge (b) Detail near the edge, showing a gray level profile and, the first and second derivative of the profile.

In Laplacian of Gaussian edge detection there are mainly three steps [4]:

1. Filtering
2. Enhancement
3. Detection.

The Laplacian is often applied to an image which is firstly smoothed with something approximating a Gaussian Smoothing filter in order to reduce the noise [6]. The detection criterion is the presence of a zero crossing in the second derivative with the corresponding large peak in the first derivative. In this approach, firstly noise is reduced by convoluting the image with a Gaussian filter. Isolated noise points and small structures are filtered out. With smoothing; however; edges are spread. Those pixels, that have locally maximum gradient, are considered as edges by the edge detector in which zero crossings of the second derivative are used. To avoid detection of insignificant edges, only the zero crossings, whose corresponding first derivative is above some threshold, are selected as edge point. The edge direction is obtained using the direction in which zero crossing occurs [4].

Since the input image is represented as a set of discrete pixels, we have to find a discrete convolution kernel that can approximate the second derivatives in the definition of the Laplacian [2]. Three commonly used small kernels are shown in Fig. 6.

1	1	1
1	-8	1
1	1	1

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

0	1	0
1	-4	1
0	1	0

Fig. 6. Three commonly used discrete approximations to the Laplacian filter.

In Laplacian method, at first smooth the input image by convolution with a Gaussian function which is equation (8) to suppress the noise.

$$G_\sigma(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp[-\frac{x^2+y^2}{2\sigma^2}] \tag{8}$$

The equation for Laplacian edge detection is follows, where ∇^2 is the Laplacian operator:

$$\nabla^2[G_\sigma(x,y) * f(x,y)] = [\nabla^2 G_\sigma(x,y)] * f(x,y) = LoG * f(x,y) \tag{9}$$

At first compute the *Laplacian of Gaussian* $\Delta G_\sigma(x,y)$ and then convolve it with the input image using the following equations,

$$\frac{\partial}{\partial x} G_\sigma(x,y) = \frac{\partial}{\partial x} e^{-\frac{x^2+y^2}{2\sigma^2}} = -\frac{x}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{10}$$

and

$$\frac{\partial^2}{\partial^2 x} G_\sigma(x,y) = \frac{x^2}{\sigma^4} e^{-(x^2+y^2)/2\sigma^2} - \frac{1}{\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \frac{x^2-y^2}{\sigma^4} e^{-(x^2+y^2)/2\sigma^2} \tag{11}$$

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 3, March 2014)

The normalizing coefficient $1/\sqrt{2\pi\sigma^2}$ is ignored for simplicity. By the same way we get,

$$\frac{\partial^2}{\partial^2 y} G_\sigma(x,y) = \frac{y^2-\sigma^2}{\sigma^4} e^{-(x^2+y^2)/2\sigma^2} \tag{12}$$

Finally, the LoG operator or convolution kernel is defined as:

$$LoG = \nabla^2 G_\sigma(x,y) = \frac{\partial^2}{\partial x^2} G_\sigma(x,y) + \frac{\partial^2}{\partial y^2} G_\sigma(x,y) = \frac{x^2+y^2-2\sigma^2}{\sigma^4} e^{-(x^2+y^2)/2\sigma^2} \tag{13}$$

2) Canny Edge Detection Algorithm: In many computer vision algorithms, the Canny edge detector is a very popular and effective edge feature detector that is used as a pre-processing step. It is a multi-step detector which performs smoothing and filtering, non-maxima suppression, followed by a connected-component analysis stage to detect "true" edges, while suppressing "false" non edge filter responses [8]. The Canny edge detection algorithm is very optimal edge detector. In this algorithm a list of criteria is followed [9] for the betterment of correct edge detection. The first one is, Low error rate. That is, the edges detected must be as close as possible to the true edges. The second is, Edge points should be well localized. That is, the distance between a point marked as an edge by the detector and the center of the true edge should be minimum. A third criterion is then added to ensure that the detector has only one response to a single edge [14]. That means every edge point should have single response.

The Canny edge detection algorithm follows the following basic steps:

Step 1: Passing the input image through a Gaussian filter to eliminate the noise and smooth the image.

Step 2: To find out the highlight regions compute image gradient using a mask and also compute the angle images. In angle, if x gradient=0, set edge direction=90^0; if y gradient=0, set edge direction=0^0.

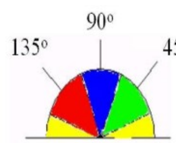


Fig. 7. Angle computation

Any edge direction falling within the yellow range (0 to 22.5 & 157.5 to 180 degrees) is set to 0 degrees. Any edge direction falling in the green range (22.5 to 67.5 degrees) is set to 45 degrees. Any edge direction falling in the blue range (67.5 to 112.5 degrees) is set to 90 degrees. And finally, any edge direction falling within the red range (112.5 to 157.5 degrees) is set to 135 degrees [6].

Step 3: Suppress the nonmaxima of the gradient magnitude image. In other words, suppressed any pixel that is not at the maximum, i.e. suppressed any pixel value (sets it equal to 0) that is not considered to be an edge.

Step 4: Perform hysteresis thresholding [10]. That is, Hysteresis [9] is used to reduce the pixels that have not been suppressed. Hysteresis uses two thresholds T1 and T2. If the magnitude<T1, made a non edge (set zero); if the magnitude>T1, made an edge; if T1<magnitude<T2, made a non edge (set zero); When the magnitude>T2 , the path considered as an edge. T2 is also used to detect and reduce the edge look like a dashed line.

III. LIMITATIONS OF ABOVE ALGORITHMS AND TECHNIQUES

A. First Derivative

The classical operators such as Prewitt, Sobel and Robert Cross which uses first derivative has very simple calculations to detect edges but its limitations are inaccurate detection and sensitivity to noise [12], discriminating between too close edges [13]. Prewitt's method has bad approximation to the gradient operator.

B. Canny

Although Canny's method is known to many as the better edge detector than the gradient-based, LoG, and zero-crossing methods, it still suffers from some practical limitations. Firstly, close edges may affect each other in the process especially when the standard deviation of the Gaussian function is too large which results in inaccurate edge locations and some edge losses. Secondly, the hysteresis thresholding requires not only the trial and error adjustment of two thresholds to produce a satisfactory edge result for each different input image but also the control of the imaging environment to assure the validity of the pre-adjusted thresholds [11]. There are also some limitations of Canny's method. After detection the edges look like a loop, expensive and complex computation, false zero crossing, time consuming. The corner pixels look in the wrong directions for their neighbors, leaving open ended edges, and missing junctions.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 3, March 2014)

C. Laplacian of Gaussian

The disadvantage is sensitivity to the noise. In detecting the edges and their orientations are increased in the noise to the image this will eventually degrade the magnitude of the edges. The second disadvantage is that, the operation gets diffracted by some of the existing edges in the noisy image [16]. It reduces the accuracy in finding out the orientation of edges and malfunctioning at the corners, curves, where the gray level intensity function variations.

IV. PROPOSED ALGORITHM AND MASK

Some limitations are solved previously by wavelet transform and others techniques. The proposed algorithm mainly solves the inaccuracy. That mean, the proposed algorithm detects more true edges by eliminating the false edges. The proposed mask is very similar to Sobel operator and is used for detecting vertical and horizontal edges in images.

The proposed algorithm follows the following steps:

Step 1: Take an image as input.

Step 2: Compute the gradient magnitudes (vertical and horizontal) of the input image using the proposed mask. The kernels of the proposed mask have shown in the Figure-8.

Step 3: Combine the gradient magnitudes and detect the edges of the input image.

Step 4: Passing the output image through the median filter to eliminate the false edges and noise.

-1	0	+1
-2.5	0	+2.5
-1	0	+1

(a)

-1	-2.5	-1
0	0	0
+1	+2.5	+1

(b)

Fig. 8. a) Proposed mask for vertical direction b) Proposed mask for horizontal direction

V. EXPERIMENTAL RESULTS

In this paper an image dataset is taken from *Berkeley Segmentation Dataset* containing a set of images with training and corresponding ground truth images for experiment and objective evaluations. In this experiment, the MATLAB 7.10.0 (R2010a) is used to process the images. In pattern recognition and information retrieval, precision is the fraction of retrieved instances that are relevant [4].

$$Precision = \frac{True\ Positive\ (TP)}{True\ positive\ (TP)+False\ Positive(FP)} \quad (14)$$

Sensitivity (also called recall rate in some fields) measures the proportion of actual positives which are correctly identified.

Sensitivity = (True Positive (TP))/(True positive (TP)+False Negative (FN)) (15)

Balanced Error Rate (BER) is also calculated for each submitted result. BER is the average of the proportion of wrong classifications in each class.

Balanced Error Rate (BER) = 100 * (1 - BCR) (16)

Where, Balanced Classification Rate (BCR) = 0.5 * (Sensitivity + Specificity).

The traditional F-measure or balanced F-score (F1 score) is the harmonic mean of precision and recall.

$$F\text{-}measure = 2.\frac{Precision.Recall}{Precision+Recall} \quad (17)$$

Mean Square Error (MSE) is defined as:

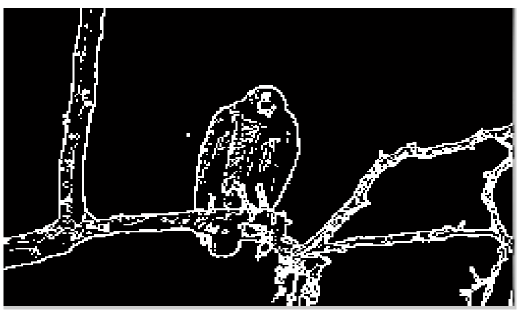
$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j) \quad (18)$$

Peak Signal to Noise Ratio (PSNR) is defined as:

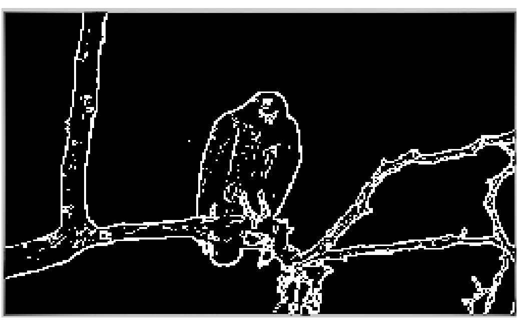
PSNR= 10.log10 (L2/Error) (19)

In this experiment, the precision, sensitivity, specificity, F-measure, MSE, PSNR and balanced error rate of the edge images comparing with the ground truth images are measured. The comparisons between different edge detection techniques are show in the Table 1. According to the comparisons the proposed algorithm has the best overall performance with the above parameters. Fig. 9(a) shows the original image, and (b)-(g) show the edge images f the original image using different operators with proposed one.

(a) Original



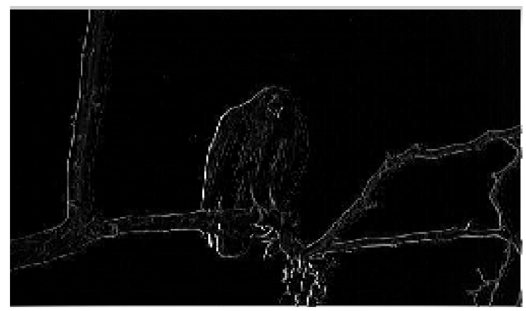
(b) Sobel



(c) Prewitt

(d) Proposed

(e) Canny



(f) Robert

(g) LoG

Figure-9: Edge images on the original image using different operators and proposed algorithm.

TABLE I
THE COMPARISONS BETWEEN DIFFERENT EDGE DETECTION TECHNIQUES

Techniques	Precision	Sensitivity	F-measure(%)	MSE	PSNR	BER (%)
Canny	0.91668	0.86844	89.19096	0.18010	7.44496	29.96208
LoG	0.94316	0.86901	90.45709	0.15688	8.04440	22.06268
Robert	0.94998	0.64405	76.76595	0.33356	4.76826	27.84354
Prewitt	0.93730	0.88298	90.93283	0.15066	8.22003	23.34894
Sobel	0.94236	0.86852	90.39341	0.15795	8.01492	22.31153
Proposed	**0.94672**	**0.89257**	**91.88473**	**0.13490**	**8.70003**	**20.25245**

VI. CONCLUSION

The proposed method detects maximum edges successfully than other methods by using the proposed mask. The experimental image is taken from *Berkeley Segmentation Dataset*. During detection of maximum edges some false edges may appear. To eliminate these false edges and noise we passed the output image through a median filter after edge detection. From the investigation, it has been shown the efficiency of proposed method is better than others. The proposed mask is based on convolving the image with a small, separable, filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations.

References

[1] N. Senthilkumaran and R. Rajesh, "Edge Detection Techniques for Image Segmentation - A Survey", Proceedings of the International Conference on Managing Next Generation Software Applications (MNGSA-08), 2008, pp.749-760.

[2] R. C. Gonzalez, R. E. Woods, "Digital Image Processing", 3rd edition, Upper Saddle River, New Jersey, Prentice-Hall Inc. (2008).

[3] Djemel Ziou and Salvatore Tabbone, "Edge detection Techniques – An overview", University of Sherbrooke, Canada.

[4] Md. Sujan Ali, "A Survey of Image Edge Detection Techniques" Proceeding of the International Conference on Electrical, Computer and Telecommunication Engineering 28—28 December 2012 (ICECTE2012).

[5] Roberts, L. G., Tippet, J. T., Machine, "Perception of Three-Dimensional Solids", Cambridge, Mass, MIT Press (1965).

[6] Raman Maini & Dr. Himanshu Aggarwal "Study and Comparison of Various Image Edge Detection Techniques", International Journal of Image Processing (IJIP), Volume (3) : Issue (1) .

[7] L.G. Roberts,"Machine Perception of 3D Solids", Optical and Electro-optical Information Processing, MIT Press, 1986.

[8] Yuancheng "Mike" Luo and Ramani Duraiswami, "Canny Edge Detection on NVIDIA CUDA", University of Maryland, College Park.

[9] J. Canny, Finding Edges and Lines in Image, Master's Thesis, MIT, 1983.

[10] Er. Snigdha Mohanty and Er. Mahesh Prasad Sahoo "EDGE DETECTION: A COMPARISON", Bhubaneswar.

[11] S. Wenchang, S. Jianshe, Z. Lin, "Wavelet Multi-scale Edge Detection Using Adaptive threshold,"IEEE, 2009.

[12] Ireyuwa. E. Igbinosa "Comparison of Edge Detection Technique in Image Processing Techniques", ITEE Journal, Volume 2, Issue 1, ISSN: - 2306-708X, February 2013.

[13] G. Seshikala, Dr.Umakanth Kulkarni, Dr.M.N. Giriprasad, "Palm Print Feature Extraction Using Multi Scale Wavelet Edge Detection Method", IJAREEIE, Vol. 1, Issue 1, July 2012.

[14] JOHN CANNY, MEMBER, IEEE, "A Computational Approach to Edge Detection", IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. PAMI-8, NO. 6, NOVEMBER 1986.

[15] P. Vidya, S. Veni and K.A. Narayanankutty," Performance Analysis of Edge Detection Methods on Hexagonal Sampling Grid", International Journal of Electronic Engineering Research, ISSN 0975 - 6450 Volume 1 Number 4 (2009) pp. 313–328.

[16] G.T. Shrivakshan, Dr.C. Chandrasekar, "A Comparison of various Edge Detection Techniques used in Image Processing", IJCSI, Vol. 9, Issue 5, No 1, September 2012.