



Development of a Novel Algorithmic Approach for Several Digital Image Processing and Analysis

Kalyan Kumar Jena*Dept. of CSE, PMEC, Berhampur
BPUT, Odisha, India**Sasmita Mishra**Dept. of CSEA, IGIT, Sarang
BPUT, Odisha, India**Sarojananda Mishra**Dept. of CSEA, IGIT, Sarang
BPUT, Odisha, India

Abstract— Edges of an image can be considered as a type of very crucial information that can be extracted by applying several edge detectors. Edge detection of Image is a basic and important concern in computer vision and image processing. Here a new approach to edge detection of several images using fuzzy method is presented. This new approach is based on the concept of Lower Construction-Laplacian of Gaussian (LOG) Edge Detection Operator. The proposed method works better for darker (low contrast) images. Fuzzy Image processing is a very well known technique used in image processing and it gives better result in many such situations as compared to the several conventional image processing techniques. Different Gray-scale images can be processed according to fuzzy topology and likewise different color images can be processed by considering the color image as multi-dimensional gray-scale images. In this concern, a new construction method focusing on interval-valued fuzzy relations from fuzzy images by vicinity is presented. This construction methodology is related to the concepts of triangular norm (t -norm). We analyze the effect of using t -norm. Again we examine the results extracted using some traditional edge detection operators such as Sobel, Prewitt, Canny and Robert. Finally, we apply the construction method to image processing and compare the result of our approach with those obtained by means of other.

Keywords— Edge Detection, Computer Vision, Image Processing, Lower Construction Method, Laplacian of Gaussian (LOG) Edge Detection operator, Fuzzy Image Processing, Gray-Scale Images, Color Images, Interval-valued Fuzzy Relations, Triangular Norm (t -norm), Sobel, Prewitt, Robert.

I. INTRODUCTION

Nowadays image processing is widely applied in many fields. The automatic processing of images is needed to suit the real world scenario [5]. An edge is a set of connected pixels that lies on the boundary between two regions which differs in their intensity value. Edges often occur at points where there is a significant variation in the intensity values on an image. Furthermore, changes in some physical and surface properties, such as illumination (e.g. shadows), geometry (orientation or depth) and reflectance can also signal an object boundary in the scene. To extract the edges from the images, several edge detection operators, such as Sobel operator, Prewitt operator, Canny operator, Roberts operator and LOG operator are commonly used and they belong to the high pass filtering [11]. The pixels which lie on the edge are known as edge pixels or edge points. Lakshmi and Sankaranarayanan (2010) and Senthilkumaran and Rajesh (2009) describe the edge categories based upon the changes in their gray value. The different types of edges are step edge, ramp edge, spike edge, roof edge.

Edge detection as a fundamental image processing method has been studied for decades and edge detection has become serious challenge to image processing scientists. A process of identifying sharp changes in pixels values with respect to neighboring pixels values is edge detection. This method is used to detect outline or contour of the objects. Edge detection algorithms are of fundamental importance for image processing applications because it's simply can determine within a short time the boundaries of objects in the image. Tested algorithms are Sobel, Prewitt, Canny, Roberts and LOG edge detection algorithms which are still maintains its popularity today [1]. In general, the aim of edge detection is to significantly reduce the amount of data in an image, while retaining the structural properties to be used for further image processing. The state-of-the-art gradient-based edge detectors lack scalability in the filter size. Small-scaled filters are sensitive to edge signals but also prone to noise, whereas large-scaled filters are robust to noise but can filter out fine details. Multi-scale edge detection face a runtime issue that when scale increases there will be a linear or quadratic increase of time consumption [3]. Many researchers have been done in the field of gray image edge detection. However, color image edge detection algorithm is not studied adequately [4].

There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories, Gradient and Laplacian. The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image. The Laplacian method searches for zero crossings in the second derivative of the image to find edges. An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location. Detection of edges for an image may help for image segmentation, data compression, and also help for well matching such as image reconstruction and so on. 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. The operator needs to be chosen to be responsive to such a gradual change in those cases. There are problems of false edge detection, missing true edges, edge localization, high computational time and problems due to noise etc. Therefore, the objective is to do the comparison of various edge detection techniques and analyze the performance of the various techniques in different conditions [7].

The paper is organized as follows: Comparison study of different edge detectors and principle of edge detection are discussed in the section II and III respectively. Several significant edge detectors, that are used in this work, are discussed in the section IV to IX. The results emerged by the selected edge detectors using the proposed technique are compared in Section X. We discuss the performance of the method and conclude with Section XI.

II. COMPARISON STUDY OF SOBEL, PREWITT, CANNY, ROBERT AND LOG OPERATORS

LOG edge detection algorithm is better than other edge detection algorithms in determining texture analysis.

Robert Operator has 2x2 mask and simple structure. Prewitt Operator and Sobel Operator have the similarity with property. In complex images, Prewitt Operator is better than Robert. LOG has very successful in image processing. Both LOG and Canny provide better result in the image with noise [1]. In gray level image, each pixel is represented by 8 bit. So, gray level values vary from 0 to 255 where 0 value stands for black color and 255 value stands for white color [2]. Prewitt, Sobel and LOG Operators can be applied for color images [3]. Canny is widely used for gray image processing. This algorithm is unable to deal with color images. Different operators such as Sobel, Robert are used for edge detection because of easy computations and these are sensitive to noise and edge determined is discontinuous [4].

Robert, Prewitt and Sobel are 1st order derivative operators. They detect edges, but the detected edges are thick edges, sensitive to noise and inaccurate. LOG is a 2nd derivative operator. It detects the thin edges. Canny is combined type of operator and it also detects thin edges and the error rate is low [5]. Lower Constructor provides lower bound of each interval and it provides better result for darker images. Upper Constructor provides upper bound of each interval and it provides better result for brighter images [6]. Sobel, Prewitt and Robert operators are computationally expensive [7]. The performance of Canny's detector is dependent on the parameters selected and it is not better in computation time as compared to other methods [8]. To extract the edges from the images, derivative edge detection operators, such as Sobel operator, Roberts operator, and LOG operator, are commonly used. These operators are computationally simple. Sobel performs worst compared with others. It extracts too many false edges and that the extracted edges are thick and discontinuous. Canny edge detector can extract fine edges while it also acquires some false edges [10].

Sobel, Prewitt, Canny, Robert and LOG operators are high pass filters, that means they provide better result for high spatial frequency regions. Sobel and Prewitt Operators are not fit for noise medical image edge detection. LOG can detect the image edge successfully. Sobel fails to detect the outer edge of body [11]. There is no single edge detection method that performs well in every possible image context. Information that could be missed by one detector may be captured by another. LOG can be used to reduce noise. Sobel, Robert and LOG are edge enhancement operators [12]. Sobel and Prewitt edge detectors detect edges having certain orientations and performed poorly when the edges are blurred and noisy. LOG is widely used smoothing filter in edge detection. It can track zero crossings over a range of scales and also gives ability to recover the entire signal at sufficiently small scales [18]. LOG does not generally create zero crossing as the scale increases but others create and it works better for multiple scales. LOG is used to filter out and remove the noise and it provides thin, sharp edges and keep the distortion small. LOG may suffer from problem of missing edges. Canny detector is more sensitive to weak edges and it requires proper sensibility, localization and local unicity of images [13]. Canny does not provide better result under noise. LOG can be used to remove noise before differentiation [14]. Sobel, Prewitt and LOG can be applied for color images [15].

From the comparison study, we concluded that LOG operator is the most successful operator among them and it is less affected by noise. But all these operators provide better results for brighter or high contrast images and unable to provide better results for darker or low contrast images. So it is very serious challenge to us to detect the edge of darker or less contrast images perfectly using these operators. Lower Construction [6] method makes the darker or low contrast image suitable for detection of edge more accurately using these operators. As LOG operator is the most successful operator, that is why we have implemented the LOG operator after implementing Lower Construction method on the darker or low contrast images in order to get accurate results.

Initially, it is required to fuzzify the darker or low contrast image and then the Lower Construction method is implemented on that image, because Lower Construction method cannot be directly implemented on an image. Finally, the defuzzification technique is applied on that particular image in order to make the image suitable for implementation of LOG operator to generate edge more accurately.

III. THE PRINCIPLE OF EDGE DETECTION

The goal of edge detection is to mark the points in a digital image at which the luminous intensity changes sharply. Sharp changes in image properties usually reflect important events and changes in properties of the world. These include discontinuities in depth, discontinuities in surface orientation, changes in material properties and Variations in scene illumination. Edge detection is a research field within image processing and computer vision, in particular within the area of feature extraction.

Edge detection of an image reduces significantly the amount of data and filters out information that may be regarded as less relevant, preserving the important structural properties of an image. There are many methods for edge detection, but most of them can be grouped into two categories [16]:

- Derivative Approach – Edge pixel or edge elements are detected by taking derivative followed by thresholding (e.g. Robert operator and 4 neighbor operator) they occasionally incorporate noise cleaning scheme(e.g. Prewitt operator Sobel operator) two dimensional derivative are computed by means of what we call is edge mask.
- Pattern Fitting Approach - A series of edge approximating functions in the form of edge template over a small neighborhood are analyzed. Parameters along with their properties corresponding to best fitting function are determined. Based on this information whether or not edge is present, is decided. We also call them edge filter.

Both the approaches have advantage and disadvantage. However it is our common experience that the second approach gives better result compared to derivative approach the reason could be followed from the primary aim of latter which is to detect derivative of image intensity and at the same time and to make the process robust to noise.

A. Edge Properties

Edges may be viewpoint dependent - these are edges that may change as the viewpoint changes, and typically reflect the geometry of the scene, objects occluding one another and so on, or may be viewpoint independent - these generally reflect properties of the viewed objects such as surface markings and surface shape. In two dimensions, and higher, the concept of perspective projection has to be considered[9].

A typical edge might be (for instance) the border between a block of red color and a block of yellow; in contrast a line can be a small number of pixels of a different color on an otherwise unchanging background. There will be one edge on each side of the line. Edges play quite an important role in many applications of image processing. During recent years, however, substantial (and successful) research has also been made on computer vision methods that do not explicitly rely on edge detection as a pre-processing [17].

B. Detecting An Edge

Taking an edge to be a change in intensity taking place over a number of pixels, edge detection algorithms generally compute a derivative of this intensity change. To simplify matters, we can consider the detection of an edge in one dimension. In this instance, our data can be a single line of pixel intensities. For instance, we can intuitively say that there should be an edge between the 4th and 5th pixels in the following 1-dimensional data[16]:

5	7	6	4	152	148	149
---	---	---	---	-----	-----	-----

To firmly state a specific threshold on how large the intensity change between two neighboring pixels must be for us to say that there should be an edge between these pixels is, however, not always an easy problem. Indeed, this is one of the reasons why edge detection may be a non-trivial problem unless the objects in the scene are particularly simple and the illumination conditions can be well controlled.

C. Approaches To Edge Detection

There are many methods for edge detection[10], but most of them can be grouped into two categories, search-based and zero-crossing based. The search-based methods detect edges by first computing a measure of edge strength, usually a first-order derivative expression such as the gradient magnitude, and then searching for local directional maxima of the gradient magnitude using a computed estimate of the local orientation of the edge, usually the gradient direction. The zero-crossing based methods search for zero crossings in a second-order derivative expression computed from the image in order to find edges, usually the zero-crossings of the Laplacian or the zero-crossings of a non-linear differential expression, as will be described in the section on differential edge detection following below. As a pre-processing step to edge detection, a smoothing stage, typically Gaussian smoothing[7], is almost always applied.

The edge detection methods that have been published mainly differ in the types of smoothing filters that are applied and the way the measures of edge strength are computed[3]. As many edge detection methods rely on the computation of image gradients, they also differ in the types of filters used for computing gradient estimates in the x- and y-directions.

IV. SOBEL EDGE DETECTOR

The Sobel edge detector (Sobel, 1970) is a gradient based edge detection algorithm. This edge detector performs a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial frequency that correspond to edges. Generally, it is used to find the approximate absolute gradient magnitude at each point in an input gray scale image [2]. This operator consists of a pair of 3x3 convolution kernels as shown in Figure 1. One kernel is simply the other rotated by 90°. It uses maximum points during the edge detection process[16].

-1	0	+1		+1	+2	+1
-2	0	+2		0	0	0
-1	0	+1		-1	-2	-1
G_x			G_y			

Fig. 1 Sobel Operator Mask [22]

These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation (call these G_x and G_y)[19]. These can then

be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient. The gradient magnitude is given by:

$$G = \sqrt{G_x^2 + G_y^2}$$

Typically, an approximate magnitude is computed using:

$$G = |G_x| + |G_y|$$

This is much faster to compute. The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by:

$$\theta = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

In this case, orientation θ is taken to mean that the direction of maximum contrast from black to white runs from left to right on the image, and other angles are measured anti-clockwise from this. Often, this absolute magnitude is the only output the user sees the two components of the gradient are conveniently computed and added in a single pass over the input image using the pseudo-convolution operator shown in Figure 2.

P ₁	P ₂	P ₃
P ₄	P ₅	P ₆
P ₇	P ₈	P ₉

Fig. 2 Pseudo-convolution kernels used to quickly compute approximate gradient magnitude

V. PREWITT EDGE DETECTOR

The Prewitt operator (Prewitt, 1970) is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function. At each point in the image, the result of the Prewitt operator is either the corresponding gradient vector or the norm of this vector. The Prewitt operator is based on convolving the image with a small, separable, and integer valued filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations. On the other hand, the gradient approximation which it produces is relatively crude, in particular for high frequency variations in the image. The Prewitt operator is named for Judith Prewitt [1].

Prewitt operator shows many similarities with the property of Sobel operator. It has two pieces 3X3 kernels shown in Figure 3. It is gradient based edge detection operator and it has gradient features. Compared to the success of edge detection in complex image, success of Prewitt operator is greater than Roberts's operator [19].

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

$$G_x$$

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

$$G_y$$

Fig. 3 Prewitt Operator Mask [22]

Mathematically, the operator uses two 3x3 kernels which are convolved with the original image to calculate approximations of the derivatives – one for horizontal changes, and one for vertical. If we define **A** as the source image, and **G_x** and **G_y** are two images which at each point contain the horizontal and vertical derivative approximations, the latter are computed as:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix} * A$$

$$G_y = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} * A$$

Since the Prewitt kernels can be decomposed as the products of an averaging and a differentiation kernel, they compute the gradient with smoothing. For example, **G_x** can be written as

$$\begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} * \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

The x -coordinate is defined here as increasing in the “right”-direction, and the y -coordinate is defined as increasing in the “down”-direction[20]. At each point in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using:

$$\sqrt{G_x^2 + G_y^2}$$

Using this information, we can also calculate the gradient’s direction:

$$\Theta = \text{atan2}(G_y, G_x)$$

where, for example, Θ is 0 for a vertical edge which is darker on the right side.

VI. ROBERT EDGE DETECTOR

Robert edge detection operator (Roberts, 1963) has a fast and simple structure. It has 2x2 convolution kernels and as shown in figure 4 these two convolution kernels is rotated 90° to each other. The kernels are applied to sequence on the image and then edge images are obtained by sum of calculated two results [1].

+1	0
0	-1

0	+1
-1	0

Fig. 4 Robert Operator Mask [22]

$$d_1 = g_0 - g_1$$

$$d_2 = g_0 - g_3$$

The differential along diagonal of a 2 X 2 mask is used and edge value after the convolution correspond to central point $(\frac{r-1}{2}, \frac{c-1}{2})$.

VI. CANNY EDGE DETECTOR

Canny edge detection algorithm (Canny, 1986) known as optimal edge detection algorithm and the most commonly used edge detection algorithm in practice. Canny has worked to improve edge detection algorithms that used popularly in those days by specifying a number of criteria. As a result of this work, Canny found that canny edge detection algorithm. The most important of criteria which are selected by Canny is to catch the low error rate. Another important criterion is that the location of identified edges should be correct. The Canny edge detector [4] was designed to be an optimal edge detector according to particular criteria. The operator works in a multi-stage process. First of all the image is smoothed by Gaussian convolution. Then a simple 2-D first derivative operator is applied to the smoothed image to highlight regions of the image with high first spatial derivatives. Edges give rise to ridges in the gradient magnitude image. Canny edge detection algorithm is composed of a few simple steps which are given as follows:

Step 1

- Noise is filtered out – usually a Gaussian filter is used
- Width is chosen carefully

Step 2

- Edge strength is found out by taking the gradient of the image
- A Sobel mask can be used

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

G_x

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

G_y

Step 3

- Find the edge direction

$$\theta = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

Step 4

- Resolve edge direction

Step 5

- Non-maxima suppression – trace along the edge direction and suppress any pixel value not considered to be an edge. Gives a thin line of edge

Step 6

- Use double / hysteresis thresholding to eliminate streaking

VIII. LOG EDGE DETECTOR

Laplacian of an image reveals that fast changing points of density in the image. Because of this property, it can used edge detection. LOG edge detector (Maini & Aggarval, 2009) take second derivative in the image and try to find zero crossing points. Since the second derivative of the image is used, this filter is very sensitive to noise. To overcome this problem, firstly the noise should be reduced by applying Gaussian smooth filter. And then LOG filter must be implemented to image [1]. Commonly used 5X5 kernels to the LOG filter which is very successful in image processing applications that has less noise level shown in Figure 5 ([20],[21]).

0	0	1	0	0
0	1	2	1	0
1	2	-16	2	1
0	1	2	1	0
0	0	1	0	0

Fig. 5 LOG Operator Mask [22]

IX. LOWER CONSTRUCTION-LOG EDGE DETECTOR

A. Lower Construction Method

The lower constructor is a generalization of the tn(t-norm) processing

A t-norm $T: [0, 1]^2 \rightarrow [0, 1]$ is an associative, commutative, increasing function, such that, $T(1, x) = x$ for all $x \in [0, 1]$.

At-norm T is called idempotent if $T(x, x) = x$ for all $x \in [0, 1]$.

The four basic t-norms [6] are as follows

1. The minimum $T_M(x,y)=\min(x,y)$.
2. The product $T_P(x, y) = x \cdot y$.
3. The Łukasiewicz t-norm

$$T_L(x, y) = \max(x + y - 1, 0)$$
4. The nilpotent minimum t-norm
5. $T_{nM}(x, y) = \min(x, y)$, if $x + y > 1$
 0 , otherwise

• Let $R \in F(X \times Y)$ be an FR. Consider two t-norms $T1$ and $T2$ and two values $n,m \in \mathbb{N}$ so that $n \leq P - 1/2$, and $m \leq Q - 1/2$. We define the lower constructor associated with $T1$, $T2$, n , and m in the following way [6]:

- $L^{n,m}T1, T2 : F(X \times Y) \rightarrow$ given by

$$L^{n,m}T1, T2 [R](x, y) = \bigwedge_{i=-n}^i \bigwedge_{j=-m}^j T1^{m,n}(T2 (R(x - i, y - j), R(x,y)))$$
for all $(x,y) \in (X,Y)$

The Algorithm begins with reading an $M \times N$ image. The first set of nine pixels of a 3×3 window are chosen with central pixel having values (2,2) i.e for each pixel (i,j) we are taking the 8 neighbourhood of (i,j) . After the initialization, the pixel values are initially marked as edge pixel after an observation to the 8 neighbourhood. After the subjection of the pixel values the algorithm generates an intermediate image using a construction method stated below. It is checked whether all pixels have been checked or now, if not then first the horizontal coordinate pixels are checked. If all horizontal pixels have been checked the vertical pixels are checked else the horizontal pixel is incremented to retrieve the next set of pixels of a window. In this manner the window shifts and checks all the pixels in one horizontal line then increments to check the next vertical location.

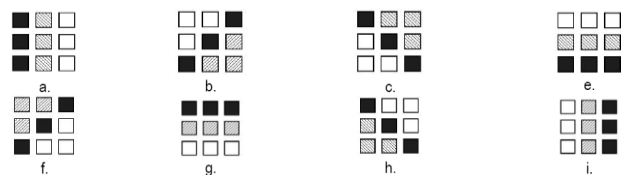


Fig. 6 Initial conditions to check 8 neighborhood [9]

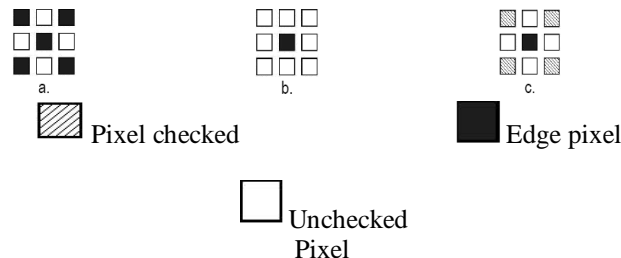


Fig. 7: (a,b)Type of unwanted edge pixels (c) condition for removal of unwanted edge pixels [9]

After edge highlighting image is subjected to another set of condition with the help of which the unwanted parts of the output image of type shown in Fig.6(a-b) are removed to generate an image which contains only the edges associated with the input image. Let us now consider the case of the fuzzy condition displayed in Fig.6 (g). For an input image A and an output image B of size $M \times N$ pixels respectively we have the following set of conditions that are implemented to detect the edges pixel values.

Input: An image A of M x N pixels (Phase 1)
 Output: An image B of M x N pixels
 Initial Edge Detection (A, B) using Min Construction
 For I←2 to M-1

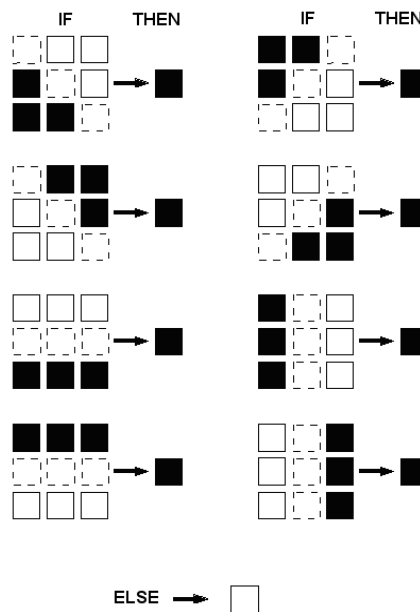
 For J←2to N-1
 If A (I-1, J)>A (I-1, J+1)
 Then If A (I-1, J-1)>A (I, J)
 Then If A (I, J-1)>A (I+1, J-1)
 Then
 B (I-1, J+1) ←0
 B (I, J) ←0
 B (I+1, J-1) ←0

 End For
 End For
 For I←2 to M-1
 For J←2to N-1
 If B(I-1,J)=255& B(I,J)=0& B(I+1,J)=255& B(I,J-1)=255
 Then B (I, J) is minimum and highlighted as edge initially.
 End For
 End For

In the above algorithm [9] Lower construction is used but not after fuzzification as after fuzzification the membership values would become fractions that can't be stored in unsigned char. Hence the same technique of min construction is used but on true picture and taking into consideration 8-nbd of a pixel (i,j). We can observe in the above algorithm written for a particular fuzzy condition that the nesting of statements is done in a manner that only the edge associated pixels are granted black pixel values and initially min valued edge pixels are given white value. These pixels are initially marked as edge.

Phase 2. Input: An image B (256 color true bmp image) of size MxN
 Output: Edge image of size MxN

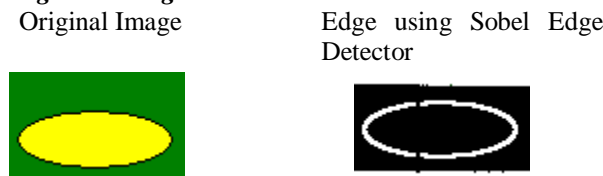
We now use Laplacian of Gaussian (LOG) operator [13] on the intermediate image to get the edge image. In this way whatever image is being constructed is compared with edges found on same image by other existing techniques. The phase 1 actually performs a check in the given manner and black pixels are having min values.



X. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, the edges of several images have shown in the Fig. 8-12 after implementation of Sobel, Prewitt, Robert, Canny and the proposed method Lower Construction-LOG edge detectors on the original images.

A. Edge Detection of Images using Sobel Edge Detector





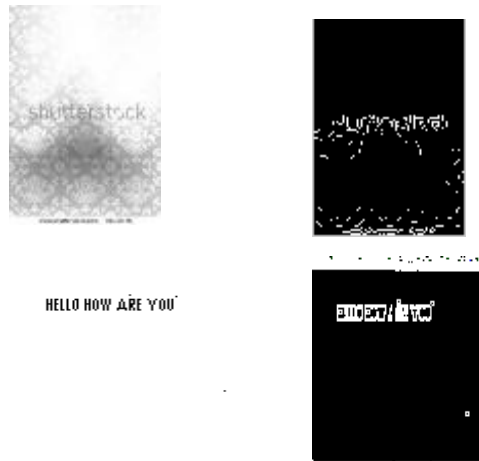
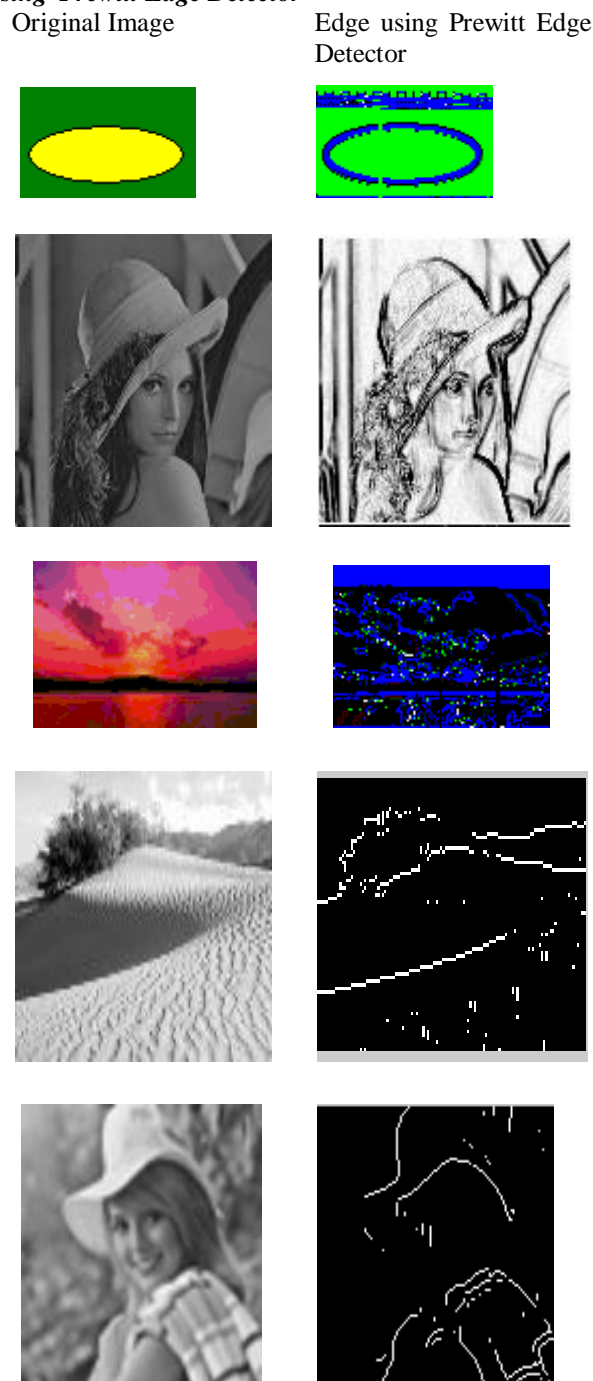


Fig. 8 Results Using Sobel Edge Detector ([23]-Original Images)

B. Edge Detection of Images using Prewitt Edge Detector



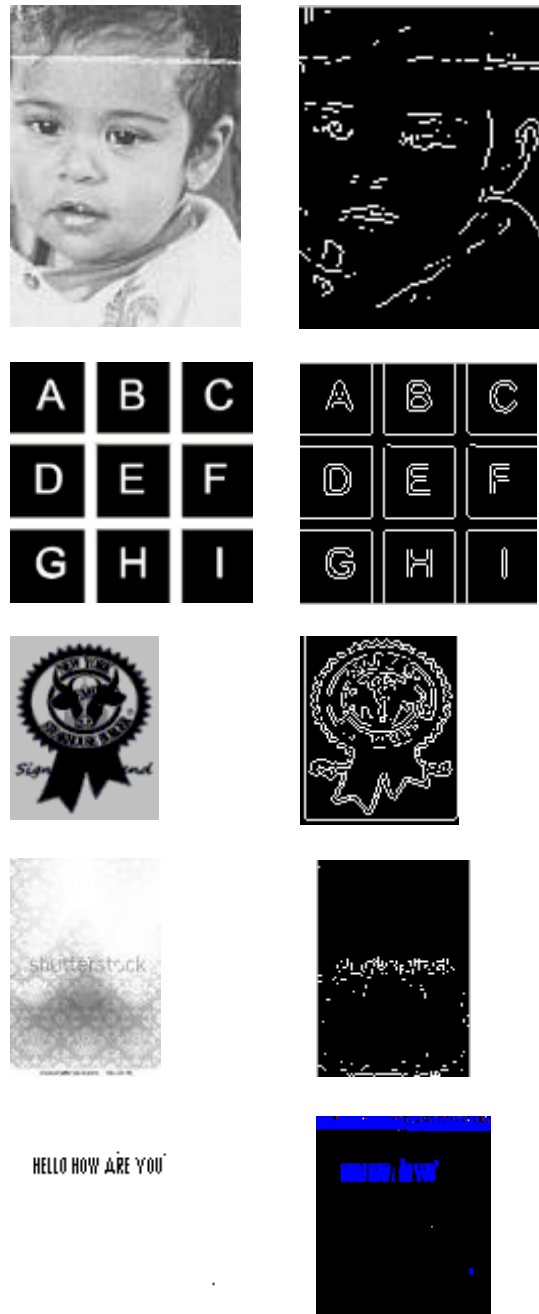


Fig. 9 Results Using Prewitt Edge Detector ([23]-Original Images)

C. Edge Detection of Images using Canny Edge Detector

Original Image Edge using Canny Edge Detector



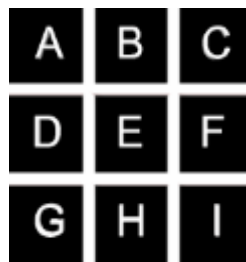




Fig. 10 Results Using Canny Edge Detector ([23]-Original Images)

D. Edge Detection of Images using Robert Edge Detector

Original Image

Edge using Robert Edge Detector

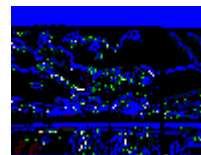
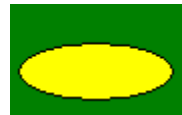




Fig. 11 Results Using Robet Edge Detector ([23]-Original Images)

E. Edge Detection of Images using Lower Construction-LOG Edge Detector

Original Image Edge using Lower Construction-LOG Edge Detector(Proposed Method)

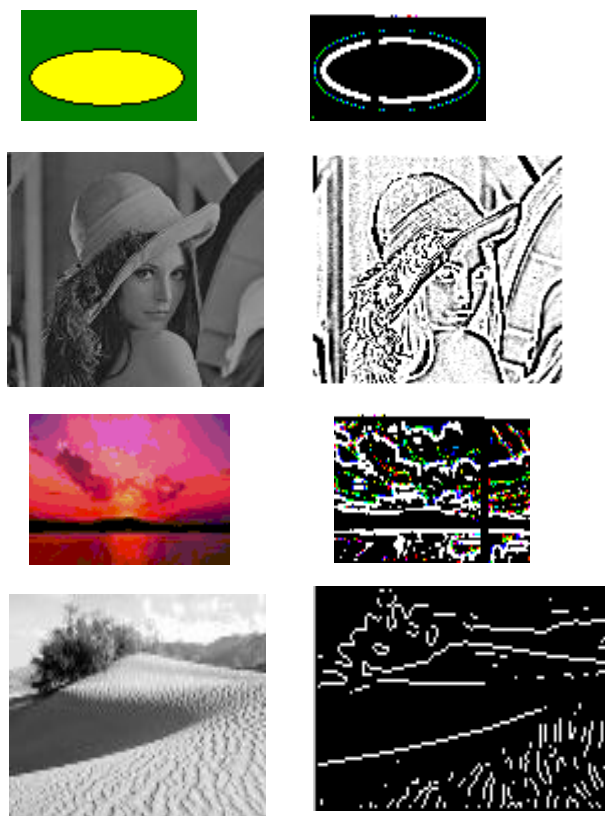




Fig. 12 Results Using Lower Construction-LOG Edge Detector ([23]-Original Images)

After analysis of edges of several images from Fig. 8-12, it is observed that for darker or low contrast images the proposed method Lower Construction-LOG edge detector provides better results as compared to other edge detectors. Several edge detectors with their performance results (percentage of accuracy on edge detection) are mentioned in Fig. 13, by considering several darker images (Fig. 8-12).

Fig. 13 Results Analysis Table

EDGE DETECTOR	PERCENTAGE OF ACCURACY
SOBEL	30
PREWITT	30
CANNY	50
ROBERT	25

LOWER CONSTRUCTION-LOG (PROPOSED METHOD)	90
------------------------------------------------	----

XI. CONCLUSION

In this paper, the algorithm to find the edges associated with several images had been implemented. Comparisons were made amongst the various edge detection algorithms. Sobel operator works better for brighter images. It can be affected by noise and it is slower than Prewitt operator. Prewitt operator also works better for brighter images. It is faster than Sobel operator and it is also affected by noise. Canny operator works better for brighter Gray images. It requires better localization of images and it cannot provide better result when the width of the image increases. Robert operator works better for brighter and small images. These four operators (Sobel, Prewitt, Robert and Canny) provide better result for Brighter or high contrast images and these operators can be affected by noise. But these operators cannot provide better result for darker or low contrast images. The proposed method lower construction-LOG operator provides better result for darker images compared to the existing operators and this operator is also less affected by noise. The proposed method can be used in several real time applications such as face recognition, medical imaging, computer guided surgery diagnosis, locate object in satellite images, finger print recognition, study of anatomical structure and automatic traffic controlling systems etc. Edge detection methodology can also be applied for analysis and processing of several tumor images in future work.

REFERENCES

- [1] Saban Öztürk and Bayram Akdemir, "Comparison of Edge Detection Algorithms for Texture Analysis on Glass Production," *ELSEVIER World Conference on Technology, Innovation and Entrepreneurship*, pp. 2675 – 2682, 2015.
- [2] Girish Chaple and R. D. Daruwala, "Design of Sobel Operator based Image Edge Detection Algorithm on FPGA," *IEEE International Conference on Communication and Signal Processing, India*, 2014.
- [3] Bo Lia, Aleksandar Jevticb, Ulrik Söderströma, Shafiq Ur Réhmana and Haibo Lic , "Fast Edge Detection by Center of Mass," *Proceedings of the 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing*, 2013.
- [4] Geng Xin, Chen Ke and Hu Xiaoguang, "An improved Canny edge detection algorithm for color image," *IEEE*, 2012.
- [5] M.Sridevi and C.Mala, "A Survey on Monochrome Image Segmentation Methods," *ELSEVIER 2nd International Conference on Communication, Computing & Security(ICCCS)*, pp. 548 – 555 ,2012.
- [6] Edurne Barrenechea, Humberto Bustince, Bernard De Baets and Carlos Lopez-Molina, "Construction of interval-valued fuzzy relations with application to the generation of fuzzy edge images," *IEEE transactions on fuzzy systems*, vol. 19, no. 5, pp. 819-830,2011.
- [7] Amiya Halder, Nilabha Chatterjee, Arindam Kar, Swastik Pal and Soumajit Pramanik, "Edge Detection: A Statistical approach," *IEEE 3rd International Conference on Electronics Computer Technology (ICECT)*, 2011.
- [8] H. Bustince, E.Barrenechea, M.Pagola and J.Fernandez, "Interval-valued fuzzy sets constructed from matrices: Application to edge detection," *ELSEVIER Fuzzy Sets and Systems*, pp. 1819–1840,2009.
- [9] Shashank Mathur and Anil Ahlawat, "Application Of Fuzzy Logic On Image Edge Detection," *ACM Intelligent Information and Engineering Systems , INFOS 2008, Varna, Bulgaria*, June-July 2008.
- [10] Jinbo Wu, Zhouping Yin and Youlun Xiong, "The Fast Multilevel Fuzzy Edge Detection of Blurry Images," *IEEE Signal Processing Letters*, vol. 14, no. 5, 2007.
- [11] Zhao Yu-qian, Gui Wei-hua, Chen Zhen-cheng , Tang Jing-tian and Li Ling-yun , "Medical Images Edge Detection Based on Mathematical Morphology," *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference Shanghai, China*, 2005.
- [12] Stamatia Giannarou and Tania Stathaki, "Edge Detection Using Quantitative Combination of Multiple Operators," *IEEE SIPS*, 2005.
- [13] Mitra Basu , "Gaussian-Based Edge-Detection Methods—A Survey," *IEEE Transactions On Systems, Man, And Cybernetics—Part C: Applications And Reviews*, vol. 32, no. 3, 2002.
- [14] Z.J. Hou and G.W. Wei, "A new approach to edge detection," *ELSEVIER Pattern Recognition*, pp.1559–1570, 2002.
- [15] J. Scharcanski And A. N. Venetsanopoulos, "Edge Detection Of Color Images Using Directional Operators," *IEEE Transactions On Circuits And Systems For Video Technology*, vol. 7, no. 2, 1997.
- [16] K. K. Jena, "Result Analysis of Different Image Edges By Applying Existing And New Techniques", *International Journal of Computer Science and Information Technology & Security*, ISSN: 2249-9555, vol. 5, no. 1, 2015.
- [17] K. K. Jena, "A New Programming and Mathematical Approach to Improve the Edge Detection of Brighter Images", *International Journal of Advanced Research in Computer Science and Software Engineering*, ISSN: 2277-128X , vol. 5, no. 1, 2015.
- [18] K. K. Jena , S. Mishra and S.N. Mishra , "Edge Detection of Satellite Images: A Comparative Study", *International Journal of Innovative Science, Engineering & Technology*, ISSN:2348 – 7968, vol. 2, no. 3, 2015.

- [19] K. K. Jena, S. Routray and A. Rahman, "Improving Edge Detection of Dark Images Using Fuzzy Method", *International Journal of Advanced Technology in Engineering and Science*, ISSN: 2348 – 7550, vol. 3, no. 1, 2015.
- [20] S. Mishra, K. K. Jena and S.N. Mishra , "Result Analysis of Edge Detection of Brighter Satellite Images: A Comparative Study of New and Existing Techniques", *International Journal of Advanced Research in Computer Engineering & Technology*, ISSN: 2278 – 1323 , vol. 4, no. 2, 2015.
- [21] K. K. Jena, S. Mishra and S. N. Mishra , "LMS:A Novel Approach for Edge detection of Images", *International Journal of Application or Innovation in Engineering & Management*, ISSN: 2319 – 4847, vol. 5, no. 6, 2016.
- [22] Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing", 2nd Edition, 2008.
- [23] www.ece.rice.edu (for standard test images)

AUTHOR PROFILE



Mr. Kalyan Kumar Jena
Asst. Prof.
Dept. of CSE
PMEC, Berhampur-761003
BPUT, Odisha, India
[Email-kalyankumarjena@gmail.com](mailto:kalyankumarjena@gmail.com)

He received B. Tech. and M.Tech. degrees in Computer Science & Engineering from BPUT, Odisha. He is continuing his Ph. D. in UU (IGIT, Sarang-Nodal Center), Odisha. He is working as an Assistant Professor in the dept. of CSE, PMEC, Berhampur. His publications comprise more than 29 papers in national and international journals. His research interests are image processing and parallel algorithms. He has more than 3 years of research as well as teaching experience. He acts as an editorial board member of various national and international journals and different professional bodies. He has acted as supervisor of B.Tech. and M.Tech. students. He received several government awards in M.Tech., B.Tech., 10th, 7th, 5th and 3rd standards.



Prof. Dr. (Mrs.) Sasmita Mishra
Associate Prof.
Dept. of CSEA
IGIT, Sarang-759146
[Email-sasmita.mishra.csea@gmail.com](mailto:sasmita.mishra.csea@gmail.com)

She received MCA and ME degree in Computer Science& Engineering from REC, Rourkela. She received her Ph. D. degree from UU, Odisha. She is working as an Associate Professor in the dept. of CSEA, IGIT, Sarang. Her publications comprise more than 40 papers in national and international journals and conferences. Her research interests are RDBMS, Analysis & Design of algorithms and Image Processing. She has more than 23 years of research as well as teaching experience. She acts as an editorial board member of various national & international journals and member of several national & international professional bodies. She has acted as supervisor of 6 Ph.D. students. She received several Central as well as State government awards.



Prof. (Dr.) Sarojananda Mishra
Prof. & HOD
Dept. Of CSEA
IGIT, Sarang-759146
[Email-sarose.mishra@gmail.com](mailto:sarose.mishra@gmail.com)

He received MCA from S.U. and M. Tech. degree in Computer Science& Engineering from IIT, Delhi. He received his Ph. D. degree from UU, Odisha. He is working as an Professor & HOD in the dept. of CSEA, IGIT, Sarang. His publications comprise more than 50 papers in national and international journals and conferences. His research interests are Image Processing, Fractals and Graphics, System Dynamics, MIS, Operation Research, Networking, Computer Programming. He has more than 25 years of research as well as teaching experience. He acts as an editorial board member of various national & international journals and member of several national & international professional bodies. He has acted as supervisor of 12 Ph.D. students. He received several Central as well as State government awards.