

Optimum Fuzzy based Image Edge Detection Algorithm

Ajenaghughrure Ighoyota Ben

Vels University, Department of Computer Science, old pallavarm, Chennai-600117, Tamil Nadu India
Email: ighoyotaben@gmail.com

Ogini Nicholas.O., and Onyekweli Charles O.

Delta State University, Department of Computer Science, P.M.B 1, Abraka, Delta State, Nigeria
Email: oginiabrakaaa@gmail.com, ogokweli@yahoo.com

Abstract—Edge detection is important in image processing to aid operations such as object classification and identification amongst others. This is solely to improve interpretability of the image. Common edge detection techniques such as Sobel, Prewitt, Canny, Laplacian of Gaussian (LOG), Robertss and Zero-Crossing has attracted the attention of researchers to perform a comparative analysis on these techniques excepts fuzzy, using different type of images. Fuzzy logic based edge detection algorithms development and comparison with existing algorithm became important due to the fact that the pixels' boundaries identifying image deqs are crystal clear as expected, hence other edge detection algorithms using crisp values will be omitting some vital information pixels, this impairs the quality of the image edge detected and further application through proper interpretation. This research further extends the investigation of edge detection techniques optimality, through comparing Sobel, Prewitt, Canny, Laplacian of Gaussian (LOG), and Robertss edge detection algorithms with our proposed fuzzy based edge detection algorithm designed using MATLAB. The result indicated that the novel fuzzy based edge detection algorithm developed in this research outperforms the Canny, Sobel, Prewitt, Robertss and LOG edge detection algorithms in three different experiments with different images

Index Terms—Edge detection, algorithm, fuzzy, algorithms, Canny, Prewitt, Sobel, LOG,

morphological method makes use of mathematical models [3]. Despite these variations, edge detection further introduces challenges such as forged edge detection, omitted exact edges, creating skinny or deep lines and challenges resulting from noisy images [2], hence motivating several researchers into investigating effectiveness of various edge detection algorithms amongst which includes [1],[2],[3],[4],[5],[6],[7] and [8].

This research paper involves the design of a fuzzy based image edge detection system. We develop a complete fuzzy inference system for accurate edge detection in comparison with most commonly used Gradient and Laplacian based Edge Detection techniques, using MATLAB 2010 Ra. This is achieved through an analytical comparative study between the fuzzy based edge detection system output from the same image with the outputs of other edge detection algorithm (Prewitt, Robertss, Sobel, Canny, and LOG)

This research paper is divided into four sections. The first section is a brief introduction; the second section is an overview of related research literature while the third section presents analysis of various gradient and Laplacian edged detection algorithms operational principles, which this research considers in addition to the proposed fuzzy based edge detection algorithm, to establish the effectiveness of the fuzzy based edge detection algorithm over others. The fourth section is a comparative analysis between the proposed fuzzy edge detection algorithm and the other edge detection algorithms considered in this research. The final section is the conclusion based on finding from the fourth section.

I. INTRODUCTION

Edge detection primarily deals with object extraction for easy identification [3] therefore leading to novel application such as civil structural analysis, medical diagnosis image analysis, security analysis of images, amongst other. Preliminary study reveals that edged detection algorithms can be classified into three main groups which are the gradient, template and morphological edge detection techniques [3]. The gradient methodology uses derivatives for tracing intensity changes, while the template is a method used in matching scheme that performs the comparison. Also, the

II. RELATED LITERATURE

Several researchers have investigated image edge detection algorithms, either comparing existing algorithms or developing new algorithms. in [1], a research was conducted to compare the effectiveness between Canny, Roberts, Sobel, and Laplacian of Gaussian edge detection algorithms, applying them on TeraSAR-X stripmap tracing paleo-shorelines of ancient lake terraces East of Lake Manyara in Tanzania. The result indicated that Canny edge detection algorithm is more efficient however, the stronger the speckle noise

and filter, the better the other edge detection algorithm outputs. Furthermore, [2] made a comparative analysis on various edge detection algorithms such as Canny, Laplacian of Gaussian Roberts, Prewitt and Sobel were investigated and the result clearly indicated that Canny edge detection algorithm is more efficient in performance than other edge detection algorithms. In addition, [3] conducted a comparative analysis between Canny, Prewitt, Roberts, Laplacian of Gaussian, and Sobel edge detection algorithms on images from the Indian Remote Sensing Satellite (IRS) sensors LISS-III, LISS-IV & Cartosat-I as well as Google Earth where the results further supported Canny as the most efficient edge detection algorithm while Sobel edge detection algorithm was efficient with respect to lesser time and space complexity. Also, [4] investigated the most efficient edge detection algorithm for flaming fire images. This study included Sobel, Canny, Roberts, Prewitt, Laplacian of Gaussian and local binary pattern. The result indicated that local binary pattern edge detection algorithm is more efficient in comparison with Canny edge detection algorithm which is also proven to be efficient but consumes more time storage space. In addition, [5] conducted a comparative analysis on various edge detection algorithms, amongst which include Boolean edge detector, Canny edge detector, color Canny edge detector, Euclidean distance/vector angle edge detector, and the multi-flash edge detector. During the analysis, Canny and Boolean edge detector performed better in grayscale image images, but Canny is still preferred since it produces single pixel, while the Boolean edge detectors are spottier. Also, the Canny color image edge detector performs better than the Canny grayscale edge detector. Furthermore, the Euclidean distance vector edge detector also performed well but lacked well refined details. Finally, the multi-flash edge detection algorithm although proved promising but requires further research. [6] developed and compared a fuzzy based edge detection algorithm with other first and second order edge detection algorithms. The proposed fuzzy algorithm is derived from first order derivative edge detection algorithms leading to fuzzy-Canny, Roberts, Prewitt, LOG, and Sobel. The result clearly showed that only the fuzzy based edge detection algorithm was more efficient in comparison to other edge detection algorithms such as Prewitt, Canny, Sobel, Roberts, Zero-Cross. In [7], a comparative analytical study was conducted on the various edge detection algorithms considering color and gray scale images. During the study, Sobel, Prewitt, Roberts, LOG, Zero-Cross and Canny edge detection algorithms were examined. The result indicated that Canny edge detection algorithm performed excellently better than all other edge detection algorithms in both gray scale and color image edge detection operations. Furthermore, [8] investigated the effectiveness of two first order derivative edge detection algorithms, which are the Canny and Sobel edge detection algorithms. The result of this study revealed that Canny edge detection algorithm performs better than Sobel edge detection algorithm with its only drawback on more time

consumption. Also, [9] investigated novel cost effective solution for bridge inspection and crack detection to replace manual bridge crack inspection methodology. This study involved comparing various edge detection algorithms such as Fast Haar Transform (FHT), Fast Fourier Transform (FFT), Sobel, and Canny for effectiveness on 50 concrete bridge images (25 without cracks and 25 with cracks). The result indicated that FHT image detection algorithm performed better than other edge detection algorithms.

Despite the numerous comparative analyses of edge detection algorithms, only Canny edge detection algorithm seem to be commonly recommended as the most effective edge detection algorithm in addition to Fast Fourier Transform which in comparison to Canny edge detection algorithm are not too different. A few researches have considered the development of fuzzy edge detection algorithm and its effectiveness in comparison with other edge detection algorithms. For example, [6] developed a fuzzy based edge detection algorithm using relative pixel, based on statistical values of SNR, PSN, and RM which are statistically computed. The rule in here is that if the value of RMSE is low and the values of SNR and PSNR are larger than the enhancement approach is better[6].

III. METHODOLOGY

In this section, we examine the existing edge detection algorithms operational principles and present the proposed fuzzy edge detection algorithms as well.

A. Roberts Edge Detection Algorithm

This is an edge detection technique that finds the differences between two adjacent pixels. This is achieved by explicitly using +1 to -1 to calculate the adjacent pixels. This is also referred to as forward difference in mathematical science. Roberts edged detection technique is practically too inefficient to determine the image edges, especially where the images are mixed with noise. One advantage of the Roberts edge detection technique is that it the simplest means of implementing the first order derivative edge detection

$$\frac{\delta f}{\delta x} = f(i, j) - f(i + 1, j + 1) \quad (1)$$

$$\frac{\delta f}{\delta x} = f(i + 1, j) - f(i, j + 1) \quad (2)$$

The partial derivatives given above can be implemented by approximating them to two 2*2 mask. The Roberts operator masks are given by

$$G_x = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } G_y = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (3)$$

These filters have the shortest support, thus the position of the edges is more accurate, but the problem with the short support of the filters is its vulnerability to noise

B. Prewitt Edge Detection Algorithm

The Prewitt edge detection technique developed by Judy Prewitt is based on the idea of central difference. The operators of Prewitt edge detection techniques are more efficient than those of Roberts edge detection technique.

$$\begin{bmatrix} a_0 & a_1 & a_2 \\ a_7 & [i,j] & a_3 \\ a_6 & a_5 & a_4 \end{bmatrix} \quad (4)$$

The partial derivatives of the Prewitt operator are calculated as follows:

$$G_x = (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6) \quad (5)$$

and

$$G_y = (a_6 + ca_5 + a_4) - (a_0 + ca_1 + a_2) \quad (6)$$

The constant c in the above expression implies the emphasis given to pixels closer to the center of the mask. G_x and G_y are the approximations at (i,j) .

Setting $c=1$, the Prewitt operator mask is obtained as:

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

and

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

The Prewitt mask has longer support. The Prewitt mask differentiates in one direction and averages in another direction, therefore making it less vulnerable to noise presence in images.

C. Sobel Edge Detection Algorithm

The Sobel edge detection algorithm was developed by Irwin Sobel. Sobel edge detection algorithm is built around the central difference, but with more emphasis on central pixels when averaging. Also, the Sobel edge detection algorithm can be thought of as a 3×3 approximations to first derivative of Gaussian edge detector. The partial derivatives of Sobel edge detector are calculated as follows:

$$G_x = (a_2 + 2a_3 + a_4) - (a_0 + 2a_7 + a_6) \quad (7)$$

$$G_y = (a_6 + 2a_5 + a_4) - (a_0 + 2a_1 + a_2) \quad (8)$$

The Sobel mask in matrix form are given below as:

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

and

$$G_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

The Sobel edge detector is more noise-proof when applied to image with noise, from a simple comparison of the mask matrix of the Sobel edge detector in comparison to the Prewitt operator mask

D. Laplacian Of Gaussian Edge Detection Algorithm

The Laplacian of Gaussian (LOG) edge detection is a second order derivative edge detection technique that uses its operator to smoothen the image through convolution with Gaussian shaped edge followed by applying the Laplacian operator. The only pitfall of the Laplacian edge detection algorithm is the impact of noise in images which can be overcome by preprocessing the image to remove the noise before applying the LOG algorithm. The LOG edge detection algorithm follows the steps below during execution

Step1: Smoothing of the input image given as $f(m,n)$ is smoothing using Gaussian mask $h(m,n)$ to get the resultant smooth image $g(m,n)$:

$$g(m,n) = f(m,n) \otimes h(m,n) \quad (9)$$

Step2: The Laplacian operator is applied to the smoothened image obtained from step1. This is represented by:

$$g'(m,n) = \nabla^2(g(m,n)) \quad (10)$$

Substituting eq(10) into eq(11) we obtain

$$g'(m,n) = \nabla^2(g(m,n)) = \nabla^2(f(m,n)) \times h(m,n) \quad (11)$$

Where $f(m,n)$ represents the original input image and $h(m,n)$ represent the Gaussian mask. The Gaussian mask is represented by

$$h(m,n) = e^{-\frac{r^2}{2\sigma^2}} \quad (12)$$

Here $r^2 = m^2 + n^2$ and σ is the width of the Gaussian.

Disadvantages of the Laplacian of Gaussian edge detection algorithm suffers from significant level of noise. In addition it usually generates contours that are not real.

E. Canny Edge Detection Algorithm

Canny edge detection algorithm developed although developed by John Canny 33 years ago, is still considered a more effective edge detection algorithm used by academics and industry today, due to its efficiency in edge search optimization problem solution. The Canny edge detector defines edges as Zero-Crossing of second order derivative in the direction of the greatest first derivative. This is achieved through the Canny operator which employs a multistage processing. The first stage is

the smoothing of the inputted image using Gaussian convolution. After which, a two dimension first derivative operator is applied to the smoothed image to highlight regions with high spatial derivatives in the image. Edges lead to ridges in the gradient magnitude image. The Canny algorithm then tracks along the top of these ridges and sets to zero all pixels that are not on the top of the ridge to create a thin line in the output. This process is referred to as the non-maximal suppression. The edge tracking process exhibits hysteresis regulated by two variables T_1 and T_2 , with $T_1 > T_2$. This implies that tracking only begins from a point greater than T_1 , and continues in both directions from the start point until the height falls below T_2 . This hysteresis prevents the possibility of noise breaking down into multiple edge fragments. The strength of Canny edge detector is determined by the following parameter

- Width of the Gaussian edge detector: The Gaussian width increase leads to reduction in the sensitivity to noise of the Canny edge detection algorithm. Also, the localization error increases in the edge detected using Canny edge detector when the Gaussian edge is increased which is primarily controls the amount of details appearing in the image edge detected and to suppress noise in the image to be processed.
- Upper threshold used: this is usually set quiet high to obtain good result in the edge detected.
- Lower threshold used by the tracker: this is usually set quiet low to obtain optimum result of the edge detected.

A typical Canny edge detection algorithm consist of the following steps

- Remove the noise from the image using a two dimensional Gaussian approximation technique on both x and y axis, due to the excessive resource utilization in two dimensional Gaussian computation.
- Obtain the changes in the x and y dimensions (gradient), because every changes in both dimension indicates an edge presence.
- Exclude all lower gradient values, to identify the real edges in the image. This is due to the fact that edge occurs in areas where there high gradient value.
- Mark out edge pixels through a process known as thresholding (high and low threshold values) using the hysteresis technique, sub-division as follows: When the value of a pixel is more than the threshold then the pixel is selected as a If a pixel has a value above the high threshold, it is set as an edge pixel. If a pixel has a value above the low threshold and is the neighbor of an edge pixel, it is set as an edge pixel as well. If a pixel has a value above the low threshold but is not the neighbor of an edge pixel, it is not set as an edge pixel. If a pixel has a value below the low threshold, it is never set as an edge pixel.

F. Proposed Fuzzy Based Edge Detection Algorithm

The use fuzzy logic method in image processing enables the adoption of membership functions to relate the extent a pixel belongs to an edge or common area. This is because, the boundary between two uniform areas could not always be an edge, and sometimes it could be a shading effect. This situation is what makes edge a not crispy boundary and fuzzy technique more suitable for edge detection problems. The proposed fuzzy based edge detection system architecture is given in fig1 below.

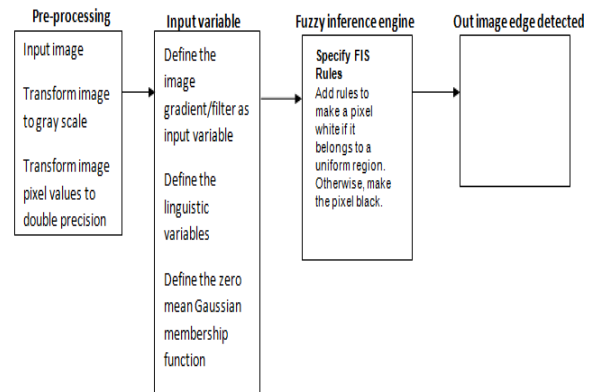


Fig.1. Fuzzy edge detection system architecture

- *Input variable*: the input variables of the fuzzy based edge detection system are the gradient of the image with respect to X and Y axis respectively. The gradient of the image is the convolution of the image pixels with respect to axis orientation within specified range of -1 to 1. A convolution of the image with gradientY creates a matrix containing only Y-axis gradient. Similarly, a convolution of the image with gradientX creates a matrix containing only X-axis gradient.
- *Fuzzy linguistics*: the linguistic variable of the fuzzy based system is basically black and white since the image to be processed by the fuzzy system has undergone some pre-processing converting it from any other form to gray scale format suitable for the fuzzy system processing. Therefore the fuzzy system linguistics is defined as

$$\text{ImageEdgeDetector-Lingusitics} = \text{black or/ white}$$

- *Input membership function*: Having specified the input variables for the fuzzy based edge detection algorithm, the next step is the defining the degree to which each input variable belongs to either uniform area(white) or non-uniform area (balck) in order to accurately detect image edges, the membership function is defined as follows

$$\text{Trimf}(\text{ImageEdgeDetector}, \text{white}, \text{black}) = \begin{pmatrix} \text{White if } w1 = 0.1, w2 = 1 \text{ and } w3 = 1 & \forall 0.1 \leq 1 \\ \text{Black if } b1 = 0, b2 = 0 \text{ and } b3 = 0.7 & \forall 0 \leq 0.7 \end{pmatrix}$$

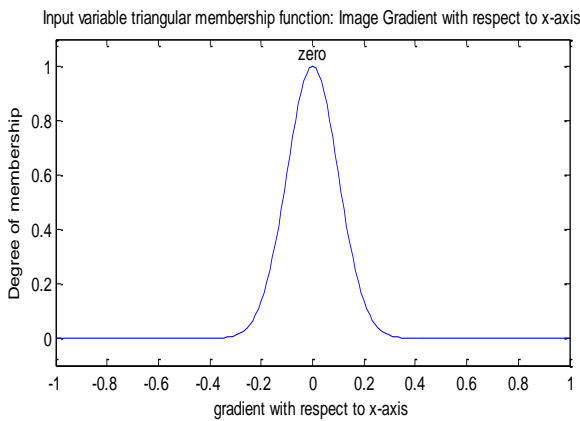


Fig.2a. Fuzzy based edge detection algorithm input variables (White) member functions

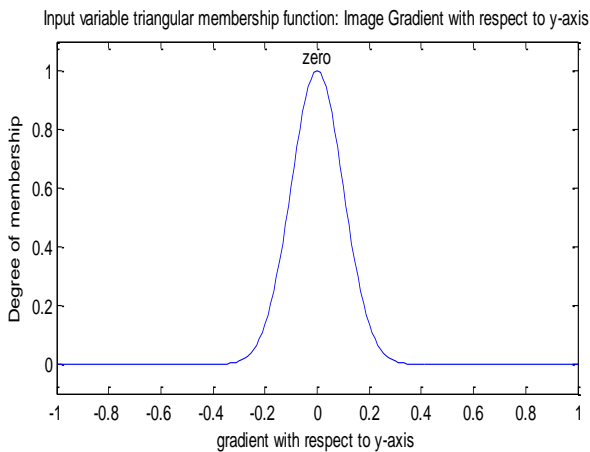


Fig.2b. Fuzzy based edge detection algorithm input variables (Black) member functions

The actual degree to which an input variable membership is evaluated is defined in the zero mean Gaussian membership function, which determines to what extent an input variable belonging to black is in white and white in black respectively. This is normally defined with respect to axis dimension for example $x=0.1$ and $y=0.1$ is defined as x is a member of white (0) but belong to black with point one degree.

- *Fuzzy inference engine:* This is often considered the knowledge based of the fuzzy system, consisting of premise and predicate. The rule base is next to be specified, since it decides which pixel is an edge or not. The fuzzy rule for this system is defined if a pixel is white then it belongs to a uniform region. Otherwise, make the pixel black.

$r1 =$ 'If ImageGradientX is zero and ImageGradientY is zero then ImageEdgeDetected is white';

$r2 =$ 'If ImageGradientX is not zero or ImageGradientY is not zero then ImageEdgeDetected is black';

- *Fuzzy output membership function:* the fuzzy based image edge detection system output membership functions are binary values for both black and white output variables.

$$F_{out} = \begin{cases} \text{white if } 1 \\ \text{black if } 0 \end{cases}$$

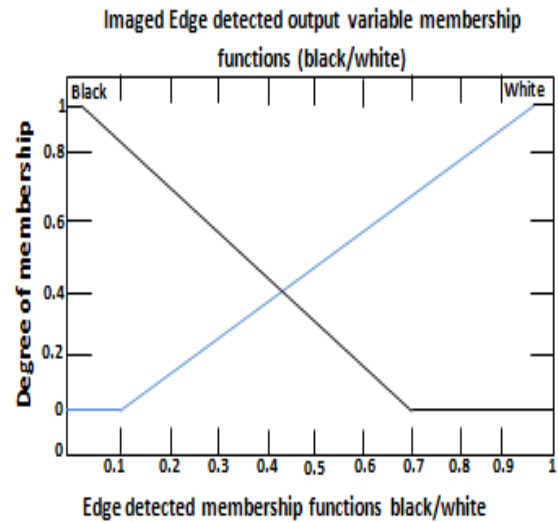


Fig.3. Fuzzy based edge detection algorithm output variables membership functions

IV. EXPERIMENTS AND RESULTS DISCUSSION

In order to validate the thesis that our proposed fuzzy based edge detection algorithm outperforms Sobel, Canny, Prewitt, Roberts and LOG edge detection algorithm, we performed three experiments with three different images as detailed below.

A. Experiment 1:

This experiment was conducted with image in fig4g below, applying Roberts, Prewitt, Sobel, Canny, LOG and the proposed fuzzy algorithm on Fig 4g below. Fig4a, b, c, d, e and f were obtained. Clearly we can observe from best to worst, that the output from the proposed fuzzy algorithm outweighs the performance of Canny, Sobel, Prewitt, Roberts and LOG edge detection algorithms, as shown in Fig4f in comparison with Fig5a, b, c, d, e. Next in performance is the Canny and log edge detection algorithm output in fig4d and e having almost similar performance. Although the Canny edge detection algorithm performs better in detecting edges than the LOG algorithm which misses out a lot of information that are covered by the Canny edge detection algorithm as observed in visually comparing Fig4d and Fig4e. These, information covered by the Canny edge detection algorithm make its output quiet difficult to interpret. Furthermore, the Prewitt and Sobel edge detection algorithm is next in performance with almost similar output as shown in Fig4b and c. While the Roberts edge detection algorithm is least performing algorithm giving

its output contains less information and lots of discontinuity in the image edge detected as shown in Fig5a.

Based on the analysis of output images Fig4a, b, c, d, e, f corresponding the application of the various edged detection algorithms considered in this study (Prewitt, Sobel, Roberts, Canny, LOG and fuzzy), we can infer that the proposed fuzzy based edge detection algorithm is more effective than all others. Since Fig4f (fuzzy based edge detection algorithm output) provides details of the image as vegetables, describable visually, such as onion observation. These details are scarcely observed in other algorithm output images (Fig4a, b, c, d, and e).



Fig.4a. Roberts edge detection algorithm output

The Roberts edge detection algorithm performance output in fig4a above poorly represents the object, due to its low sensitivity to image edges. Visually inspecting the graphical performance output in fig4a above, no object can be visibly recognized for both vision spatial cognitively disabled and able people as well.



Fig.4b. Prewitt edge detection algorithm output

The output of the Prewitt edge detection algorithm in Fig4b above significantly improved visual spatial cognitive analysis, as some of the object can be identified as fruits/vegetables generally. But edge detection sensitivity although higher than Roberts edge detection algorithm, requires significant improvement.



Fig.4c. Sobel edge detection algorithm output

The output from the Sobel edge detection algorithm in Fig4c above is similar to the performance output of the Prewitt edge detection algorithm and also suffers the same drawbacks.



Fig.4d. Canny edge detection algorithm output

The Canny edge detection algorithm performance output in fig4d above is more easily interpretable with vision spatial cognitive capability, due to its higher sensitivity to image edge detection. Visually analyzing the graphical output of the Canny edge detection algorithm in Fig4d above, although more effective than the Prewitt, Roberts and Sobel edge detection algorithm, its high sensitivity to image edge detection makes it more difficult to interpret as a result of its inclusion of noise (relevant and irrelevant edges).



Fig.4e. LOG edge detection algorithm output

Visually analysinh the graphical output of the LOG edge detection algorihtm performance output in Fig4e above is a bit closer to the performance output of the Canny edge detection algorithm in Fig4d, but higher in performance than the Sobel,Prewitt and Roberts edge detectionalgorithm.

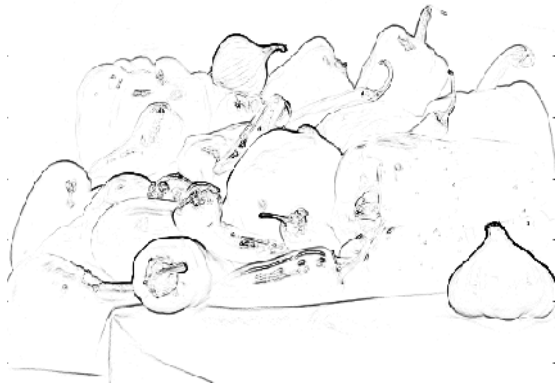


Fig.4f. Fuzzy edge detection algorithm output

The fuzzy based edge detection algorithm performance output in Fig4f above clearly depicts the object as collection fruits/vegetables. This is due to both higher sensitivity to edge detection and accuracy of edge membership to present inclusion of noise as part the detected edges. Hence the fuzzy based edge detection algorithm is considered more effective than all the other edge detection algorithms under study (Canny, LOG, Prewitt, Sobel and Roberts)

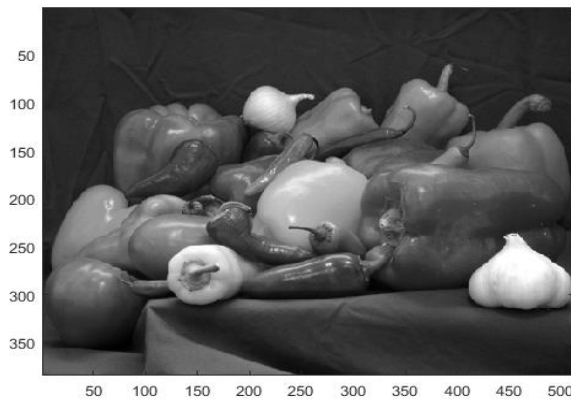


Fig.4g. original input image

B. Experiment 2:

This experiment was conducted with image in Fig5g below, applying Roberts, Pewitt, Sobel, Canny, LOG and the proposed fuzzy algorithm on fig5g below. Fig5a, b, c, d, e and f were obtained. Clearly we can observe from best to worst, that the output from the proposed fuzzy algorithm outweighs the performance of Canny, Sobel, Prewitt, Roberts and LOG edge detection algorithms, as shown in Fig5a in comparison with Fig5b, c, d, e, f. Next in performance is the Canny and log edge detection algorithm output in Fig5e and f having almost similar performance. Although the Canny edge detection

algorithm performs better in detecting edges than the LOG algorithm which misses out a lot of information that is covered by the Canny edge detection algorithm. These, information covered by the Canny edge detection algorithm make its output more difficult to interpret. Furthermore, the Prewitt and Sobel edge detection algorithm is next in performance with almost similar output as shown in Fig5b and d. While the Roberts edge detection algorithm is least performing algorithm giving its output contains less information and lots of discontinuity in the image edge detected as shown in Fig5c.

Hence, based on the analysis of output images Fig5a, b, c, d, e, f, g corresponding the application of the various edged detection algorithms considered in this study, we can infer that the proposed fuzzy based edge detection algorithm is more effective than all others. Since Fig4a (fuzzy based edge detection algorithm output) is provides details of the image as a bird with eye, beaker and claws on both legs. These details are scarcely observed in other algorithm output images (Fig5b, c, d, e and f).

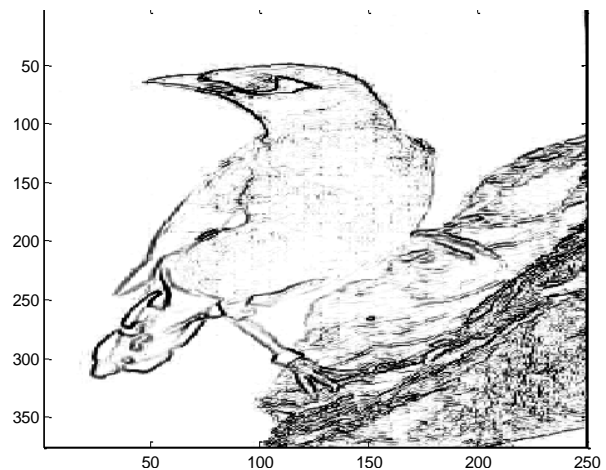


Fig.5a. Fuzzy edge detection algorithm output

The output of the fuzzy based image edge detection algorithm depicted in fig5a above depicts the object clearly as a bird, with eye, long beak, two legs with claws,

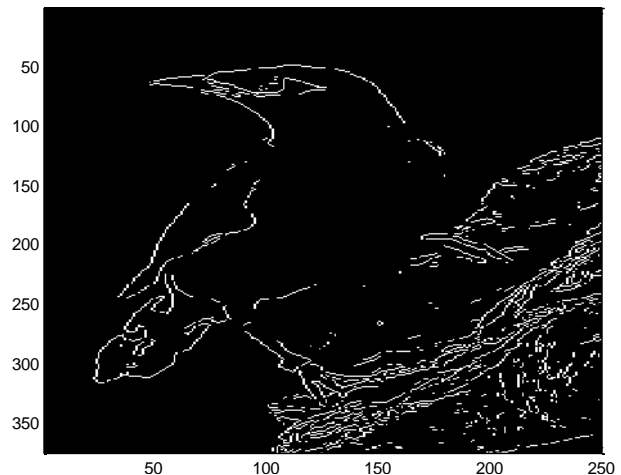


Fig.5b. Sobel edge detection algorithm output

perched on a rough surface suspended in the open sky as clearly the background of the image is purely white. Furthermore, the bird skin texture also appears to be rough, as expected of a bird with feathers.

The Sobel edge detection algorithm output showed in Fig 5b above poorly indicate the object as a bird and its background. Visually inspecting the output in Fig5b above, clearly the object is poorly identified as a bird, with only one leg, with long beak, with eye, without skin texture as either rough or smooth and lots more missing and poorly represented features of the object (bird)

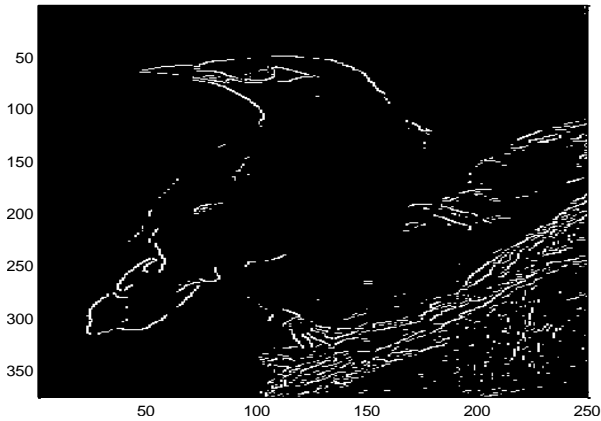


Fig.5c. Roberts edge detection algorithm output

Furthermore, the Roberts image edge detection algorithm output depicted in Fig5c above, extremely poorly represented the object beyond easily recognition to everyone. This will be mostly difficult for those with Visio spatial cognitive disability, because the object analysis presented in Fig5c above although barely represents a bird, but without legs, eye, feather, skin texture, status such as perched or flying.

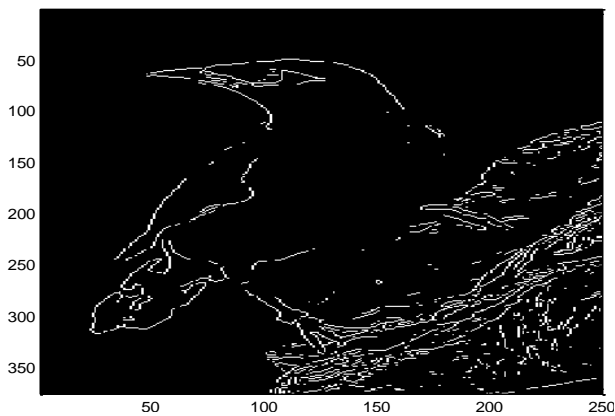


Fig.5d. Prewitt edge detection algorithm output

The Prewitt edged detection algorithm performance output in fig5d is almost similar to the performance output of the Sobel edge detection algorithm in fig5b above. Though, the Prewitt edge detection algorithm poorly identified the object as a bird with similar features as those identified by the Sobel edge detection algorithm, it's still fall short of accuracy.

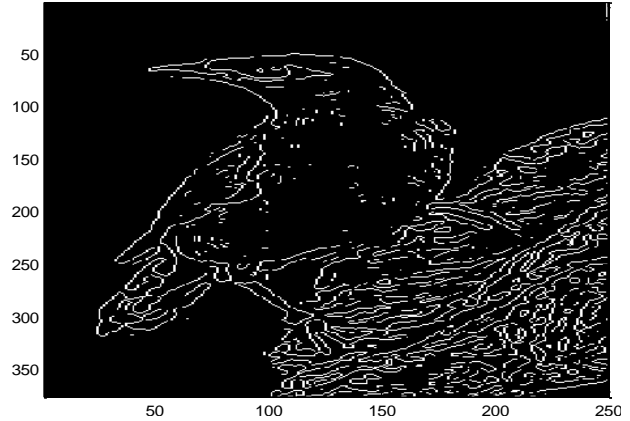


Fig.5e. LOG edge detection algorithm output

The LOG edge detection algorithm output in Fig5e above is quiet more efficient that the Prewitt, Robertss and Sobel edge detection algorithm, due to its higher sensitivity to detecting image (Fig5g) egdes. This obviously presents the processed image features better, but over representation also makes the output (Fig5e) from the LOG edge detection algorithm quiet difficult to interprete. For eacmple the output in fig5e above can be seen as a bird, that peched on a platform (unkonw but with rough surface), with only one leg barely visible, long beak, unrecornigised eye, poorly representedd skil surface texture, and lots more.

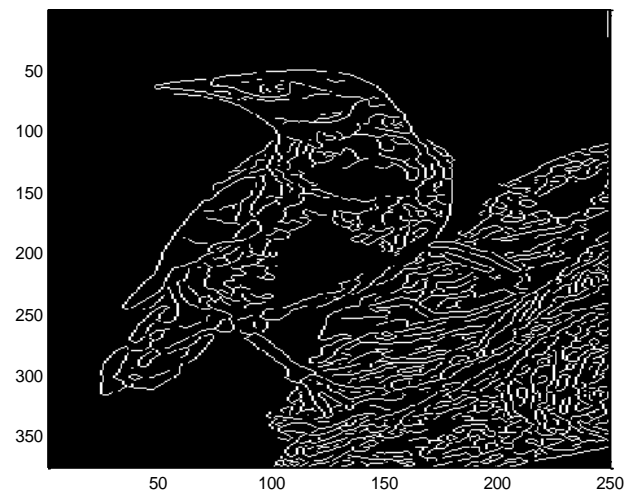


Fig.5f. Canny edge detection algorithm output

The output from the Canny edge detection algorithm in fig5f above is almost similar to the performance ouptu of the LOG edge detection algorithm in fig5e above. Therefore making the analysed object recognizable easily Due to to its higher sensitive to detecting image edges than the LOG, Prewitt, Robertss, and Sobel edge detection algorithms. This higher sensitivity further imapires easy visio spartial cognition of the object due to difficulty in interpretation of object analysed.



Fig.5g. Original image used for the experiment.

C. Experiment 3:

This experiment was conducted with image in Fig6g below, applying Roberts, Prewitt, Sobel, Canny, Log and the proposed fuzzy algorithm on fig6g below. Fig6 a, b, c, d, e and f were obtained. Clearly we can observed from best to worst, that the output from the proposed fuzzy algorithm outweighs the performance of Canny, Sobel, Prewitt, Roberts and LOG edge detection algorithms, as shown in Fig6a in comparison with Fig6b, c, d, e, f. Next, in performance is the Canny and LOG edge detection algorithm output in Fig5d and e having almost similar performance. Although the Canny edge detection algorithm performs better in detecting edges than the LOG algorithm which misses out a lot of information that is covered by the Canny edge detection algorithm. These, information covered by the Canny edge detection algorithm make its output quiet more difficult to interpret as show in fig6e below, while these unattended information by the LOG algorithm makes it. Furthermore, the Prewitt and Sobel edge detection algorithm is next in performance with almost similar output as shown in fig6b and c. While the Roberts edge detection algorithm is least performing algorithm giving its output contains lesser information and lots of discontinuity in the image edge detected as shown in fig6f.

Hence, based on the analysis of output images Fig6a, b, c, d, e, f corresponding to the application of the various edged detection algorithms considered in this study, we can infer that the proposed fuzzy based edge detection algorithm is more effective than all others. Since Fig6a (fuzzy based edge detection algorithm output) is provides details of the image as a deer with tree-like horns, surrounded by grasses and tree leaves, and in motion with four legs etc. These details are scarcely observed in other algorithm output images (Fig6b, c, d, e and f).

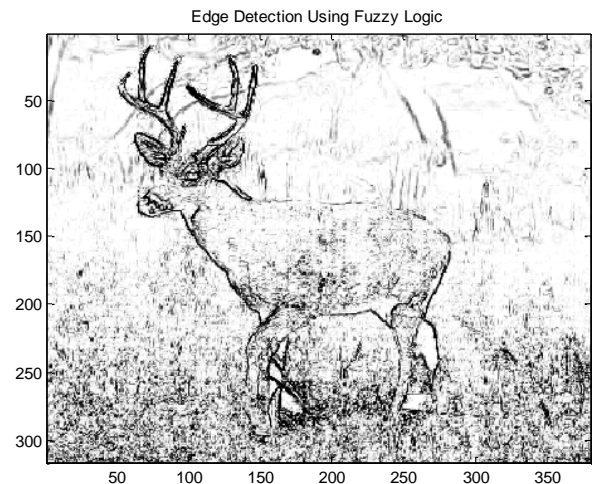


Fig.6a. Fuzzy edge detection algorithm output

The fuzzy based edge detection algorithm output as shown in Fig6a above was derived from the application of the fuzzy based edge detection algorithm on Fig6f. the result in Fig6a clearly identifies a deer in a bush with tree and grasses

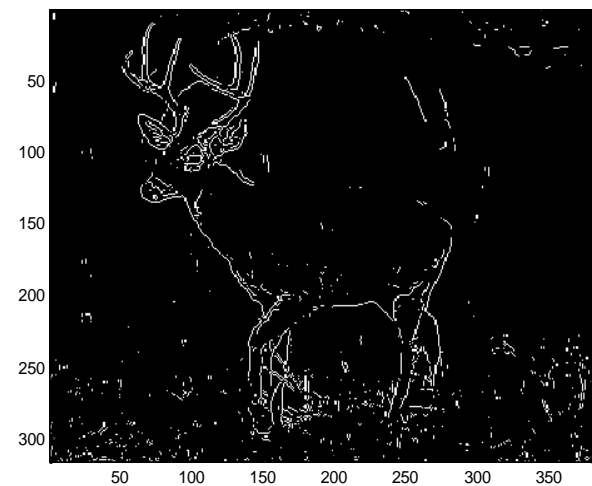


Fig.6b. Sobel edge detection algorithm output

The output of the Sobel edge detection algorithm application on fig6g shown in fig6b above, although identifies a deer-like object through its head, but imperfectly presented, because the back side edge of the deer was not detected, in addition to the background uninterpretability. That whether the deer is is bush or forest etc.

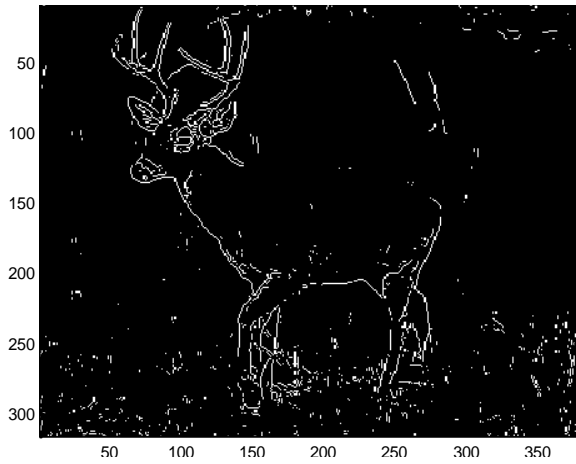


Fig.6c. Prewitt edge detection algorithm output

The Prewitt edge detection algorithm output as shown in fig6c above is also almost similar to the Sobel edge detection algorithm in performance through visual analysis of graphical results. They both suffer the same set back

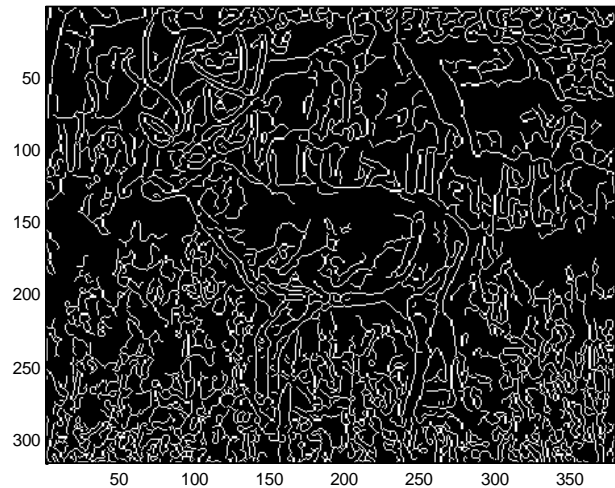


Fig.6e. Canny edge detection algorithm output

The Canny edge detection algorithm output in Fig6e above is much more effective in detecting edges in an image than the LOG and Prewitt and Sobel. Visually analysing the graphical output, the Canny edge detection algorithm is lot more sensitive and detailed in edge detection, but includes noise which makes its output lot more difficult to interpret and analyse.

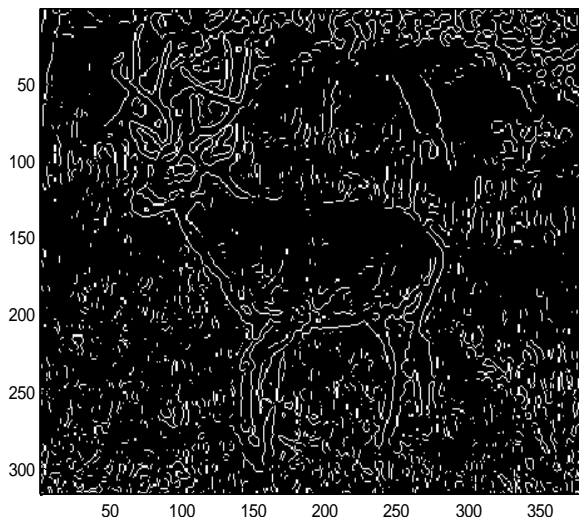


Fig.6d. LOG edge detection algorithm output

The LOG edge detection algorithm output in fig6d above is highly effective than the Sobel and Prewitt edge detection algorithms. Visually comparing their graphical output, the LOG edge detection algorithm detects more edges identifying the object as a deer than others. Though, its background is highly noisy and uninterruptable.

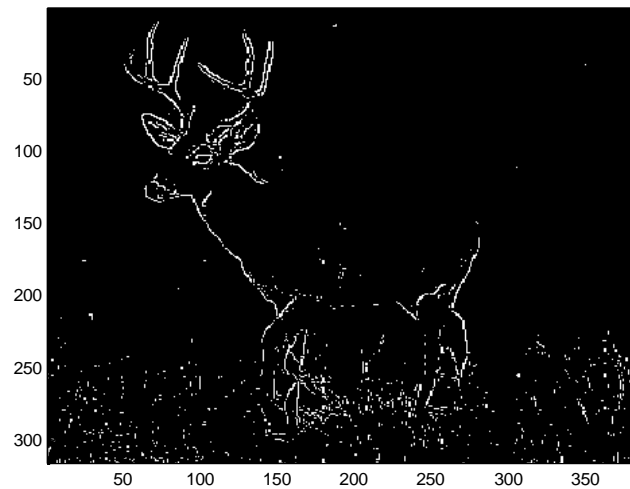


Fig.6f. Roberts edge detection algorithm output

The least efficient edge detection algorithm of all is the Roberts edge detection algorithm as shown in Fig6f above. Visually comparing its output with others, it rarely depicts the deer image accurately, as the legs, head, back and background are poorly represented.



Fig.6g. Original input image

V. CONCLUSION

In conclusion, this research paper delved into the design and experimentation of a fuzzy based edge detection algorithm with three different sample input images with complex background to identify the feasibility and effectiveness of a fuzzy based edge detection algorithm for image processing. In addition, a comparison between the designed fuzzy based edge detection algorithm and Sobel, Canny, Roberts, Prewitt and Laplacian of Gaussian (LOG) edge detection algorithms was conducted. The result clearly indicates that the design fuzzy based edge detection algorithm outperforms the Canny, Roberts, Prewitt, LOG and Sobel edge detection algorithms in the three-test experiment conducted. These are obvious visibly proven through direct observation of the comparison of Fig4f with Fig4a, b, c, d and e in the first experiment, Fig5a with Fig5b,c,d,e, and f in the second experiment and Fig6a with Fig6b, c, d, e, and f in the third experiment. Hence the fuzzy based edge detection algorithm is recommended as part of further research in image edge detection medical imaging, civil/engineering imaging, object recognition, shape recognition amongst others. In addition, further research into its comparison with other edge detection algorithm will be conducted.

REFERENCE

- [1] Felix Bachofer, Geraldine Quénervé, Thimm Zwiener, Michael Maerker and Volker Hochschild, Comparative analysis of Edge Detection techniques for SAR images European Journal of Remote Sensing - Vol49: pp 205-224, 2016.
- [2] Raman Maini & Himanshu Aggarwal, Study and Comparison of Various Image Edge Detection Techniques International Journal of Image Processing (IJIP), Volume (3) : Issue (1), pp1-12, February 2009.
- [3] S.K. Katiyar P.V. Arun, Comparative analysis of common edge detection techniques in context of object extraction, IEEE Transactions on Geosciences and Remote Sensing, Vol. 50, NO. 11b, pp 68-79, November 2012
- [4] Zakir Hussain, Diwakar Agarwal, A comparative analysis of edge detection techniques used in flame image

processing, International Journal of Advance Research In Science And Engineering IJARSE, Vol. No.4, Special Issue (01), pp1335-1343, March 2015

- [5] Parvinder Singh Sandhu, Mamta Juneja and Ekta Walia, Comparative Analysis of Edge Detection Techniques for extracting Refined Boundaries 2009 International Conference on Machine Learning and Computing IPCSIT vol.3 (2011) © (2011) IACSIT Press, Singapore PPI-10
- [6] T. Dharani I. Laurence Aroquiaraj V. Mageshwari, Comparative Analysis of Edge Detection Algorithms Based on Content Based Image Retrieval With Heterogeneous Images, International Journal of Computational Intelligence and Informatics, Vol. 5: No. 4, pp374-38, March 2016
- [7] Shaveta Malik, Tapas Kumar, Comparative Analysis of Edge Detection between Gray Scale and Color Image Communications on Applied Electronics (CAE) – ISSN : 2394-4714 Foundation of Computer Science FCS, New York, USA Volume 5– No. 2, 38 -43, May 2016
- [8] Vineet Saini, Rajnish Garg, A Comparative Analysis on Edge Detection Techniques Used in Image Processing, IOSR Journal of Electronics and Communication Engineering (IOSRJECE) ,Volume 1, Issue 2, PP 56-59, May-June 2012,
- [9] Canny, John, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 6, pp. 679-698. 1986
- [10] Lim, Jae S., *Two-Dimensional Signal and Image Processing*, Englewood Cliffs, NJ, Prentice Hall, pp. 478-488, 1990
- [11] Parker, James R., *Algorithms for Image Processing and Computer Vision*, New York, John Wiley & Sons, Inc., pp. 23-29. 1997. Abdel-Qader, I., Abudayyeh, O., and Kelly, M. "Analysis of Edge-Detection Techniques for Crack Identification in Bridges." journal of computing in civil engineering, Volume 17, Issue 4, pp255-263, October 2003.

Authors' Profiles



Ighoyota B. Ajenaghughrure, recently his masters in computer science from Vels University India, in 2016 may. While he obtained his bachelor's in computer science from Delta State University. His research interest are in artificial intelligence, Human computer Interaction, health and business information systems, Trust and security computing, and computer vision application.

Ogini Nicholas O. Obtained his bachelors of science, masters of science and PhD in computer science in 1993, 1996 and 2013 respective all from University of Benin Nigeria. His research interest are centered around database security and artificial intelligence. Ogini is a senior lecturer with 13yrs teaching experience as a faculty in computer science department of computer science in delta state university. In addition, ogini is currently the director if ICT in delta state university Abraka, Nigeria.



Charles O. Onyekweli in 2012 obtained a Masters degree in Information Technology Management from The Roberts Gordon University (RGU); Aberdeen, Scotland and in 2007, a Bachelors degree in Computer Science from Benson Idahosa University (BIU); Benin City, Nigeria.

He is currently an ASSISTANT LECTURER with the Department of Computer Science at The Delta State University; Abraka, Nigeria. He has previously worked as an IT BUSINESS

ANALYST and IT HELPDESK OFFICER (VOLUNTEER) at RGU, IT SUPERVISOR at United Africa Company of Nigeria (UACN) Plc., and COMPUTER SYSTEM ENGINEER at National Bureau of Statistics among other corporate positions.

He has also over the past couple of years and in collaboration with other authors, published a few articles relevant to some of his areas of interest. These interests bother on Computer process and Security Improvement, Human Computer Interaction, Mobile Device Management, Strategic IT Management, IT Project Management, among others.

How to cite this paper: Ajenaghughrure Ighoyota Ben, Ogini Nicholas.O., Onyekweli Charles O., "Optimum Fuzzy based Image Edge Detection Algorithm", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.9, No.4, pp.44-55, 2017.DOI: 10.5815/ijigsp.2017.04.06