ISSN 2229-5518

Comparative Analysis of Iris Edge Detection Algorithms

Thompson, A.F., Olotuah, R.T. Oluwadare, S.A.,

Abstract— Biometric identification is a technology that is adopted in the recognition system which guards against fraud and multiple identities. In biometric application system, a high-quality image processing algorithm is desirable as images often used have sharp discontinuities called edges, which are abrupt changes in pixel intensity that characterize boundaries of the objects. These edges require detection in order to remove irrelevant information in the image and preserve the relevant ones for better image processing. In image processing, edge detection is a fundamental tool most especially in feature detection and extraction. Various operators are often adopted in edge detection and the design and quality of the technique determine its efficacy. Edge detection reduces the complexity of image processing algorithm by reducing the amount of data to be processed through the removal of irrelevant information and the preservation of the relevant ones. The focus of this study was to determine the efficacy and limitations of Canny, Sobel, Robert, Prewitt, Laplacian of Gaussian (LoG), Petrou-Kittler and Multiscale Directional Filter Bank (DFB) edge detection algorithms on 1,200 iris images. The iris images were acquired from CASIA iris Data Base Version 4 and the edges detected using the seven algorithms. Edge values and Edge maps were obtained on respective technique and these were compared using performance evaluation metrics which included, processing time, accuracy, specificity, sensitivity and correlation parameters using MATLAB 2017b. Consequently, the edge maps obtained from the MATLAB were also visually compared. On comparing the algorithms using the metrics, results showed that Prewitt was able detect edges on the selected iris images in fewer seconds of 106.8196 compared to the other algorithms. The Laplacian of Gaussian having obtained the accuracy value of 96.7% had proved to be the most accurate of all the algorithms, although, Canny, Sobel, Robert and Prewitt also displayed similar and considerably high level of accuracy. The most sensitive technique is the Multiscale Directional filter bank with sensitivity value of 0.9887. Consequently, Canny, Sobel, Robert, Prewitt and Laplacian of Gaussian all displayed very high level of specificity while Laplacian of Gaussian excels with 0.9805. All the seven algorithms performed below average except Petrou-Kittler with a correlation parameter value of 0.5607. This study gave an insight to the attributes of the algorithms and the features required in the selection of the best approach to edge detection to achieve optimal result. This further enhanced improved and reliable method of biometric identification.

Index Terms— Biometrics, Edge detection, Iris, Gaussian smoothing, Image Processing, Edge operators, Image Acquisition.

1 Introduction

The biometric identification system is one of the technologies used in the recognition system. Unlike the use of other forms of authentication, such as passwords or tokens, biometric recognition provides a strong link between an individual and a claimed identity. It equally provides substantial help in guarding against attempts to establish fraudulent multiple identities or prevent identity fraud. [1] reported that Biometric Identification systems perform a one-to-many comparison against a biometric database in an attempt to establish the identity of an unknown individual. So, in electronic transactions, biometric authentication is very popular and secure. In biometric application systems, users of digital images desire to improve the native resolution offered by imaging hardware. Image interpolation aims to reconstruct a higher resolution (HR) image from the associated low-resolution (LR) capture.

- *Thompson A.F.(PhD) is a Senior Lecturer in Computer Sciences Department, Federal University of Technology, Akure, Nigeria. E-mail: bmkubuoye@futa.edu.ng*
- *Olotuah R.T. is currently pursuing Master degree program Computer Sciences Department, Federal University of Technology, Akure, Nigeria. E-mail: roseline.olotuah@aaua.edu.ng*
- *Oluwadare S.A. (PhD) is a Senior Lecturer in Computer Sciences Department, Federal University of Technology, Akure, Nigeria. E-mail: saoluwadare@futa.edu.ng*

This is also applicable in medical imaging, remote sensing and digital photographs [2]. With the prevalence of inexpensive and relatively LR digital imaging devices and the ever increasing computing power, interests in and demands for high-quality image interpolation algorithms have also increased.

Images often have sharp discontinuities called edges, which are abrupt changes in pixel intensity that characterize boundaries of the objects. Consequently, most semantic and shape information from the image can be encoded in the edge being a place of rapid change in the image intensity. This edge requires detection in order to remove irrelevant information in the image and preserve the relevant ones for better image processing [3].

Therefore, edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly in the areas of feature detection and feature extraction. It is the name for a set of mathematical methods which aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. These points are typically organized into a set of curved line segments termed edges. The same problem of finding discontinuities in 1-D signals is known as step detection and the problem of finding signal discontinuities over time is known as change detection.

The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. It can be shown that under rather general assumptions for an image formation model, discontinuities in image brightness are likely to correspond to discontinuities in depth, discontinuities in surface orientation, changes in material properties and variations in scene illumination [4]. In the ideal case, the result of applying an edge detector to an image may lead to a set of connected curves that indicate the boundaries of objects, the boundaries of surface markings as well as curves that correspond to discontinuities in surface orientation. Thus, applying an edge detection algorithm to an image may significantly reduce the amount of data to be processed and may therefore filter out information that may be regarded as less relevant, while preserving the important structural properties of an image. If the edge detection step is successful, the subsequent task of interpreting the information contents in the original image may therefore be substantially simplified. However, it is not always possible to obtain such ideal edges from real life images of moderate complexity.

Edges extracted from non-trivial images are often hampered by fragmentation, meaning that the edge curves are not connected, missing edge segments as well as false edges not corresponding to interesting phenomena in the image – thus complicating the subsequent task of interpreting the image data [4].

There are several edge detection operators and each is designed to be sensitive to certain types of edges. Also, the variables often considered in the selection of 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 which in turn can be optimized to look for horizontal, vertical, or diagonal edges. One of the factors that beset edge detection is noisy images which contain high frequency content like the edges and there are several operators used for noise reduction to discount localized noisy pixels. Consequently, an attempt to reduce the noise may result in the formation of blurred and distorted images. Therefore, noise and edge imperfection should be given due consideration, so that edge filters could be constructed traditionally, so as to improve the suppression of unwanted disturbances by appropriate low-pass filtering such that there will be good compromise between noise reduction and edge localization [5].

Several methods could be adopted in edge detection but these could be either Gradient or Laplacian based.

This study was thus focused on the comparative efficacy of some detectors from both the gradient and the Laplacian in iris image processing. The operators include: Canny, Sobel, Robert, Prewitt, Laplacian of Gaussian, Petrou-Kittler and Multiscale Directional Filter Bank Edge Detectors.

2 METHODOLOGY

The methodology for this study was in three phases namely: image acquisition phase, edge detection phase and Performance comparative stage.

In this study, the following edge detection algorithms: Canny, Sobel, Robert, Prewitt, Laplacian of Gaussian (LoG), Petrou-Kittler and Multiscale Directional Filter Bank (MDFB) edge detector were identified and used on selected Iris images.

The work was aimed at using these detectors to identify points on the iris images where the brightness changes abruptly and then compare the performance of the detectors with some performance metrics earlier stated.

2.1 Image Acquisition

This phase was the importation of Iris images from the CASIA database using MATLAB functions and was followed by the implementation of the various algorithms on the downloaded Iris images.

2.2 Detection of Edges

This phase was characterized by detection of edges using the seven detection algorithms. This phase was also in stages based on the implementation procedures.

2.2.1 Detection of Edges using Canny

The canny edge detector which is considered to be a standard and optimal algorithm was implemented in phases. The first phase was the smoothing of the image with Gaussian function to reduce noise in the image, which was done by approximating two 1-Dimensional Gaussian in the x and y directions respectively:

$$\left(G_\sigma(x,y) = \frac{1}{2\pi\sigma} exp^{-\frac{x^2}{2\sigma^2}}\right)\left(G_\sigma(x,y) = \frac{1}{2\pi\sigma} exp^{-\frac{y^2}{2\sigma^2}}\right) \tag{1}$$

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

Equation 1 represents the two 1-D Gausian function,and Equation 2 represents the approximated Gaussian function.

where $G(x, y)$ is the image,

(x, y) are the cartesian coordinate of the image pixels,

$e^{\pi}(x^2+y^2)$ is the eigen function of the fourier transformation,

$1/2\pi\sigma^2$ is the normalizing factor, and

σ is the standard deviation / width of the Gaussian and controls the degree of smoothing.

After the smoothing operation, a set of images with different level of smoothing was obtained as shown in Equation 3:

$$g(x, y, \sigma) = G_{\sigma(x,y)} * g_{(x, y)} \tag{3}$$

The next stage in the Canny process was to find the derivative 2D, which is the gradient vector of the derivative to x and y. This shows changes in intensity, which indicates the presence of edges. This actually gave two results, the gradient in the x direction and the gradient in the y direction as shown in equation 4 and the angle of orientation as shown in equation 5.

$$|\nabla f| = \sqrt{f_x^2 + f_y^2}$$

$$= \sqrt{\left(\frac{\partial y}{\partial x}\right)^2 + \left(\frac{\partial y}{\partial x}\right)^2} \approx |f_x| + |f_y| \tag{4}$$

$$\theta = \tan^{-1}\left(f_y | f_x\right) \tag{5}$$

Since edges occur at points where the gradient is at a maximum, all points not at a maximum were suppressed. This was done after the magnitude and direction of the gradient has been computed at each pixel. Then, for each pixel check, where the magnitude of the gradient was greater at one pixel away in either the positive or the negative direction perpendicular to the gradient, the pixel was made an edge. Alternatively, where the pixel was not greater than both, it was suppressed and this is referred to as Non- maxima suppression.

The next stage of the second phase actually carried out the identification of edges. This stage is called thresholding by "hysteresis". It makes use of both a high threshold and a low threshold. where a pixel has a value above the high threshold, it was set as an edge pixel and when a pixel has a value above the low threshold and also the neighbor of an edge pixel, it was set as an edge pixel as well. Where a pixel has a value above the low threshold but not the neighbor of an edge pixel, it was never set as an edge pixel, also where a pixel has a value below the low threshold, it was never set as an edge pixel.

2.2.2 Detection of Edges using Sobel

This performed a 2-D spatial gradient measurement on the Iris images and so emphasis regions of high spatial frequency that correspond to edges by finding the approximate gradient magnitude at each point as shown in equation 6 in the input grayscale image, these was then combined to find the absolute magnitude of the gradient as shown in equation 7 and the angle of orientation as in equation 8 respectively.

$$|G| = \sqrt{Gx^2 + Gy^2} \tag{6}$$

$$|G| = |Gx| + |Gy| \tag{7}$$

$$\theta = arctan(Gx/Gy) \tag{8}$$

2.2.3 Detection of Edges using Robert

Pixel values at each point in the output in Robert represented the estimated absolute magnitude of the spatial gradient of the input image at that point while the operator consists of a pair of 2x2 convolution masks. This is similar to Sobel.

2.2.4 Detection of Edges using Prewitt

The Prewitt operator uses the same equation as the Sobel operator, just that the constant does not place any emphasis on pixels that are closer to the center of the masks. It is used for detecting vertical and horizontal edges in images [6]. Prewitt operator suppresses noise by averaging image information, but the average of image information is equivalent to a low-pass filter. Therefore, the boundary localization by Prewitt operator is not as good as Robert's operator.

2.2.5 Detection of Edges using Laplacian of Gaussian

The Laplacian of Gaussian algorithm is the combination of the Gaussian filter and the Laplace filter. The Laplacian is a 2-D isotropic measure of the 2nd spatial derivative of an image. The Laplace operator is sensitive to noise, thus, the image had to be smoothed with Gaussian filter as shown in equation 9.

LoG is an orientation-independent filter (i.e. no information about the orientation) that breaks down at corners, curves, and at locations where image intensity function varies in a nonlinear manner along an edge. As a result, it cannot detect edges at such positions. The smoothing and differentiation operations can be implemented by a single operator consisting on the convolution of the image with the Laplacian of the Gaussian function and the final form of the filters, known as LoG with scale σ which was convolved with the image is as defined in equation 11:

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

The Laplacian ***L(x, y)*** of an image with pixel intensity values ***f(x, y)*** is given by equation 10:

$$f(x,y) = \nabla^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} \tag{10}$$

$$LoG(x,y) = \frac{1}{\pi\sigma^4}\left|\frac{x^2+y^2}{2\sigma^2}\right| e^{-\frac{x^2+y^2}{2\sigma^2}}$$

(11)

2.2.6 Detection of Edges using Directional Filter Bank

The directional filter bank is composed of an analysis filter bank and a synthesis bank. The analysis filter bank splits the original image into eight directionally passed sub-band images while the synthesis bank combined the sub-band image into one image. It uses a fan filter banks followed by checkerboard filter to obtain its wedge shaped responses, which produced the edge information. This provided directional specific information and it was subjected to various edge labeling algorithms to obtain the edge map. The checkerboard is given as in equation 12 while the fan filter is given as in equation 13.

The steerable and scaled DFB has been presented to obtain directional along with scaled information. From the DFB based image decomposition, the scaled information is combined by scale multiplication. The whole process of the DFB can be represented with the algorithms:

Input: Iris Image
Step 1: DFB decomposition
Step 2: Combining Responses
Step 3: Applying Non maximal Suppression + Hysteresis Threshold (or)
Step 4: Gray threshold (or) K – means clustering
Output: Detected Edges:

$$C(Z_1, Z_2) = \frac{1}{2}\left[P_{0(-Z_1{}^2)} P_{0(-Z_2{}^2)Z_0 Z_1 P_{1(-Z_1{}^2)} P_{1(-Z_1{}^2)}}\right] \quad (12)$$

$$F_v(e^{jw1}, e^{jw2}) = e^{j(w1+w2)}/2, e^{j(w1+w2)}/2 \quad (13)$$

2.2.7 Detection of Edges using Petrou-Kittler

The Petrou-Kittler [7] is a Directional Filter Based method used to build a 2-D edge filter that collect information from the different filtering direction around a central pixel, the 2-D was specified by three parameters, the number of filtering direction n, the radial span ɩ, and the angular overlap δ. The Petrou-Kittler was primarily constructed as 1-D filters and then extended appropriately into two dimension and the 2-D filtering components are applied in orthogonal directions to estimate the magnitude (and direction) of 2-D edges. 1-D filter to two dimensions is typically performed by applying the edge filter in the direction perpendicular to the edge and a projection function along the edges. The projection function used was the Gaussian function applied in the angular direction, the edge strength maps generated using convolution of images with the tri-directional filter:

$h(u) = ɲ.s(r) \,.\, p(\theta)$ Where,

ɲ is the normalization constant, and the pixel vector u is specified by its radial and angular components, r and θ. The radial component of a 2-D filter, s(r), and $p(\theta) = \delta$.

2.3 Performance Evaluation

Finally, the performance of the adopted Canny method, Sobel edge detector, Robert edge detector, Prewitt edge detector Laplacian of Gaussian (LoG), the Petrou-Kittler edge detector and Multiscale Directional Filter bank (MDFB), were evaluated using the following parameters:

Sensitivity (***Se***): It is the ratio of correctly classified instances among all instances and is given by:

$$Se = \frac{TP}{TP + FN} \quad (15)$$

Specificity (*Sp*): The specificity is a function of FP and TN, and is given by:

$$Sp = 1 - \frac{FP}{FP + TN} \quad (16)$$

Accuracy is given as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (17)$$

Where:

TP stands for True Positive and describes when edge pixel in an image is detected correctly as an edge pixel.

FN stands for False Negative and describes when edge pixel detected wrongly as non-edge pixel

TN stands for True Negative and describes when Non-edge pixel detected correctly as non-edge pixel

FP stands for False Positive and describes when Non-edge pixel is detected wrongly as edge pixel

Processing time: is the time taken by the system to run the edge detection algorithm in seconds. Given that i = {1, 2, 3… n} represent set of images, and n is the image upper bound, the set of edge detection algorithms is given as j = {1, 2, 3…m) where m is the 6th edge detection technique. The processing time (*Pt*) for each technique, *j*, is given as follows;

$$Pt = p1 + p + p3 \ldots\ldots p(n-2) + p(n-1) + pn \quad (17)$$

where the Average Processing Time for a technique is given as:

$$APt = \frac{p1 + p + p3 \ldots\ldots p(n-2) + p(n-1) + pn}{n} \quad (18)$$

Cross Correlation Parameter: This is a qualitative measure for edge preservation. To evaluate the performance of the edge preservation or sharpness, the correlation parameter is defined as follows:

$$CP = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}(\Delta I - \Delta \bar{I})\, x\, (\Delta \hat{\imath} - \Delta \hat{\imath})}{\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}(\Delta I - \Delta \bar{I})^2\, x \sum_{i=1}^{m}\sum_{j=1}^{n}(\Delta I - \Delta \bar{I})^2}} \quad (19)$$

Where ΔI and $\Delta \hat{\imath}$ are high pass filtered versions of original image *I* and filtered image $\hat{\imath}$ obtained via a 3x3 pixel standard approximation of the Laplacian operator. The $\Delta \bar{I}$ and $\Delta \hat{\imath}$ are the mean values *I* of and $\hat{\imath}$, respectively. The correlation parameter should be closer to unity for an optimal effect of edge preservation

The work was implemented using MATLAB (R2017a) version 9.2.5 on window operating system platform. MATLAB was chosen because it allows matrix manipulations, plotting of functions, implementation of algorithm, and creation of user interfaces most of which are required in this work.

3 Results

The comparison of the efficacy of the algorithms was based on the use of 1,200 images. Table 1 shows the average value of the metrics obtained from the 1,200 iris images compared using the seven algorithms. Thus, thus, performance evaluation table clearly reveals the efficacy of each of the algorithms. Consequently, the fastest image processing time was observed in Prewitt operator that ranked 4th and 5th in accuracy and sensitivity respectively. The most sensitive of all the operators was the Multiscale Directional Filter Bank (MDFB) that ranked 4th in the processing time and 7th in both accuracy and specificity. Laplacian of Gaussian (LoG) was observed to be most efficient in terms of specificity, accuracy and also preserves edges most as seen in the correlation parameter, although ranked 3rd and 5th in sensitivity and processing time respectively. The performance analysis highlighted in table 1 shows the average edge values on 1200 iris images.

TABLE 1: MEAN METRIC VALUES OF THE ALGORITHMS USING 1,200 IRIS IMAGES

Edge Tech.	Pro. Time	Acc.	Sens.	Spec.	Cor. Par.
1. Canny	169.8652	95.4228	0.4538	0.9730	0.4007
2.Sobel	109.9094	95.1376	0.3281	0.9754	0.3085
3. Robert	110.6415	94.7903	0.3271	0.9719	0.2914
4. Prewitt	106.8196	95.0769	0.3349	0.9747	0.3072
5.LoG	144.5313	96.7240	0.6407	0.9805	0.5607
6. Pet. Kitt.	218.0004	38.4211	0.9658	0.3622	0.3567
7. MDFB	112.9289	37.1805	0.9887	0.3482	0.3344

3.1 Graphical representation of the comparative analysis of the operators

For further clarity, the graphs of each of the metrics were drawn for all the methods considered based on table 1.

3.1.1. Determination of processing time for all the seven algorithms

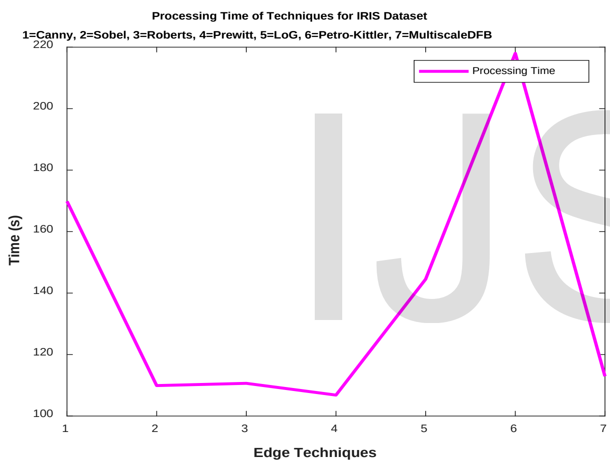

Fig. 1. Graph showing the comparative processing time of the operators

From the performance analysis in table 1, it was discovered that the Prewitt operator detected edges in fewer seconds and took less execution time when compared to the other six algorithms. This is shown in figure 1.

3.1.2. Determination of Accuracy for all the seven algorithms

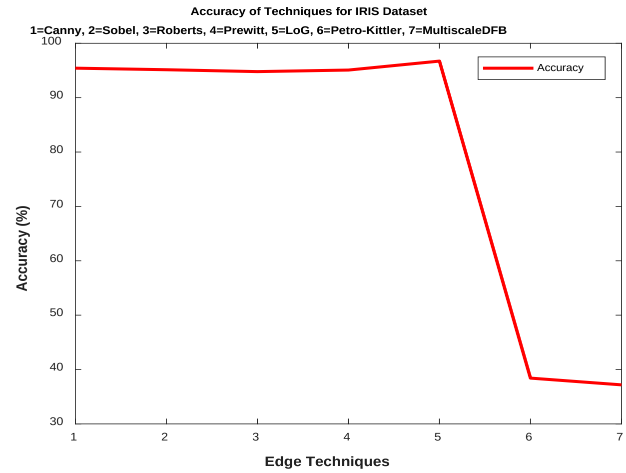

Fig. 2. Graph showing the comparative accuracy of the operators.

From the analysis of the accuracy measures of the different edge detection methods considered, it could be observed that the Laplacian of Gaussian (LoG) method attained the highest percentage of accuracy, followed by Canny edge detector. Though, Sobel, Robert and Prewitt operators similarly showed considerable high level of accuracy as shown in Figure 2.

This highest percentage of accuracy displayed by LoG and Canny detector is never unconnected with the benefit of the application of Gaussian function for noise reduction during the edge detection process.

3.1.3. Determination of Sensitivity for all the seven algorithms

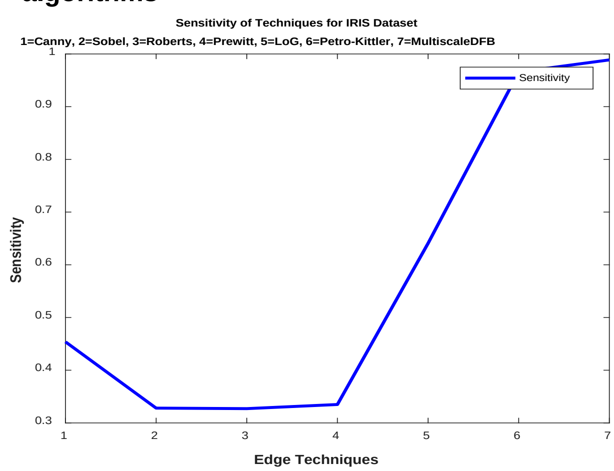

Fig. 3. Graph showing the comparative sensitivity of the operators.

From the analysis measure of sensitivity of each of the detector as shown in Figure 3, it was discovered that the Multiscale Directional Filter bank is the most sensitive closely followed by the Petrou-Kittler detector and the Laplacian of Gaussian operators. The Sobel and the Robert algorithms are less sensitive, closely followed by Canny.

3.1.4. Determination of Specificity for all the seven algorithms

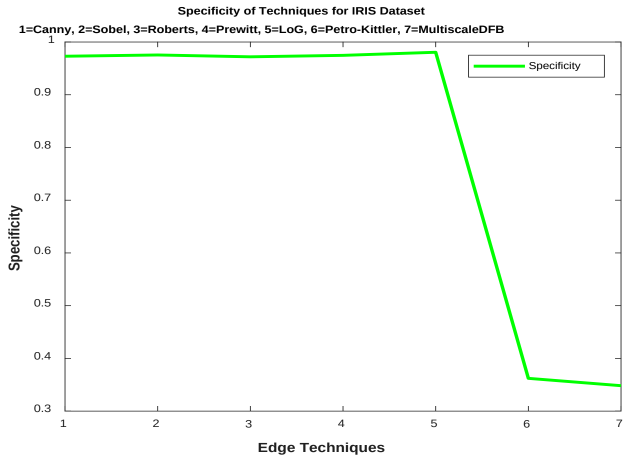

Fig. 4. Graph showing the comparative specificity of the operators.

From the analysis of specificity level, the LoG displayed the highest specificity level closely followed by Sobel, Robert, Prewitt and the Canny detectors as shown in Figure 4.

Table 2 shows the values of performance rating of respective operators with respect to the 5 metrics on a Likert scale of 1-7 except in the processing time where values were assigned to each technique in reverse order as the highest figure was assigned the lowest value. Consequently, the final average column in Table 2 gives the cumulative performance of the operators based on the rating.

TABLE 2. SCALE 1-7 PERFORMANCE RATING OF THE OPERATORS ON TABLE 1.

Based on the performance of each of the algorithms with respect to the metrics used, a final average of the performance was drawn as indicated in the last Column of table 2. This is to evaluate the most effective technique based on the parameters considered as shown in Figure 6.

3.1.5. Determination of Correlation Parameter for all the seven Algorithms.

Edge Tech.	Pro. Time	Acc.	Sens.	Spec.	Cor. Par.	Final Ave.
Canny	2	6	4	4	6	4.4
Sobel	6	5	2	6	3	4.4
Robert	5	3	1	3	1	2.6
Prewitt	7	4	3	5	2	4.2
LoG	4	7	5	7	7	6
Pet. Kitt.	1	2	6	2	5	3.2
MDFB	3	1	7	1	4	3.2

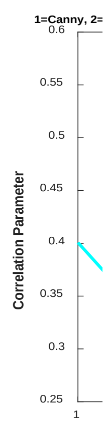

Fig. 5. Graph showing the comparative correlation parameter of the operators

It was discovered that the Laplacian of Gaussian (LoG) had the highest performance level in the measurement of Correlation Parameter. This is a qualitative measure for edge preservation as shown in Figure 6.

3.2. Graphical representation on Scale 1-7 Performance rating of the Operators

The graphical representation of the average value in Table 2 is represented in Figure 6.

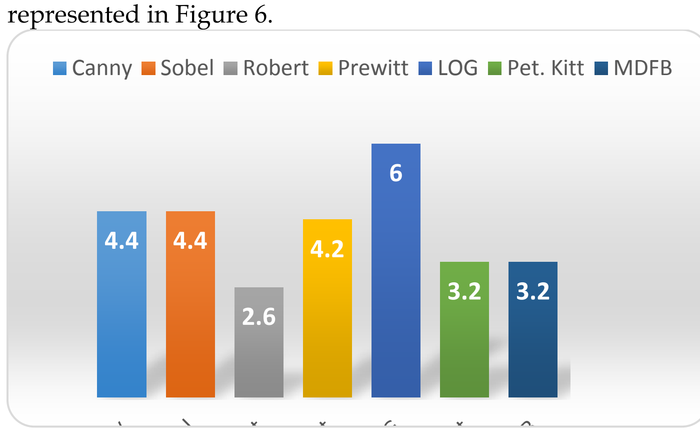

Fig. 6. Graph showing the Cumulative Performance of the Operators Based on average value of the detection on 1200 images.

3.3 Edges of Iris Images

In this study, 1,200 images were considered, some samples of the edge maps obtained from the edge detection process images are displayed in Figure 7.

ISSN 2229-5518

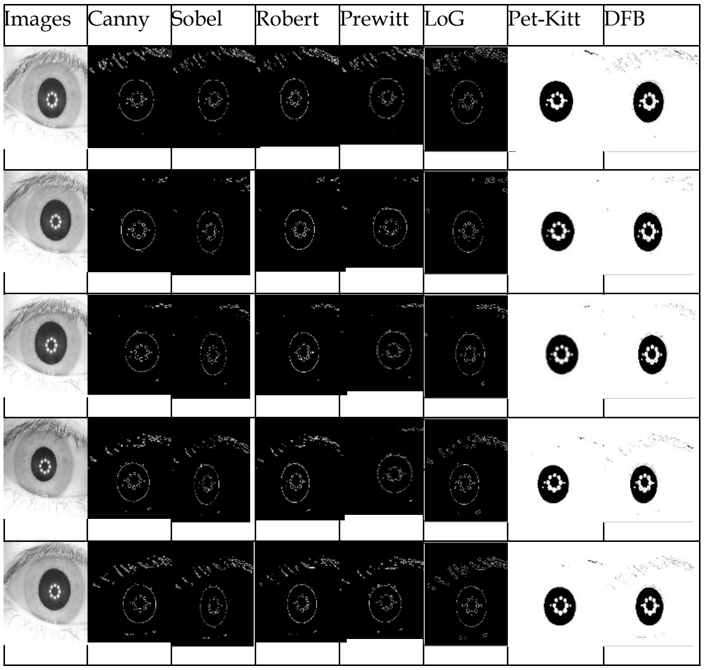

Fig. 7. Samples of Iris Edge Maps

4. Discussion of Results

Edge detection which is the basic framework of this study was observed on 1,200 iris images using seven operators. The collection of facial images corroborates the method adopted by [8], in the comparative analysis of Sobel and Canny detectors. In the comparative study of the seven operators, performance indicators which included, processing time, sensitivity, accuracy, specificity and correlation parameter were used in the study as validating parameters to verify the quality of segmented images. The degree of performance observed varied amongst the operators as reflected in the graphical representations from Figure 1 to 5. Table 1 shows the comparative analysis of the detectors. The values on the table indicate that no operator is completely outstanding with respect to the metrics. This concept is also shared by [8] while comparing the Sobel and Canny edge detector methods using the MRI images. It was discovered that the Canny method produced good edge with the smooth continuous pixels and thin edge, while the Sobel edge detector could not produce smooth and thin edge compared to the Canny method.

In the comparative analysis table1, it was observed that the Canny edge detection technique gave a processing time of 169.8652 seconds, which was the second largest out of all the edge algorithms. With respect to processing time, the edge detection technique was definitely seen to be computationally expensive. It gave a sensitivity of 0.4538 and a specificity of 0.973. This suggests that the Canny edge system tends to misclassify some of the foreground pixels as background, that is, some edge pixels were missed out. A specificity of 0.973 suggests that the Canny edge detection correctly classified most of the background pixels and did not misclassify them as edge pixels. However, the Canny edge system gives a high accuracy of 95.42%, suggesting that it classified the edge pixels and background pixels correctly with a high rate. This character exhibited is not unconnected with Canny's good compromise between noise suppression and edge detection. Similarly, this feature had been reported by [9] in the study on some edge detectors for iris recognition system. Canny gave a correlation parameter of 0.4 which specified that it produced a considerable level of sharp images. Ideally, the parameter should be closer to unity.

The Sobel edge detection technique gave a processing time of 109.9094 seconds, which is the second lowest of all algorithms, indicating that it is less time expensive compared to Canny. It gave a sensitivity and specificity of 0.32 and 0.97 respectively. This suggests a misclassification of some of the edge pixels as background. However, most of the background pixels were classified correctly. The high level of specificity displayed by this operator may however be due to its simplicity and for the fact that it is easy to program. It gave an accuracy of 95% and a correlation parameter of 0.308. Sobel edge detection is comparable to Roberts in terms of sensitivity and specificity. However, it does take the lead in terms of better computational time and greater accuracy.

The Roberts edge detection technique gave a processing time of 110.6415 seconds, which was one of the lowest. It gave a sensitivity and specificity of 0.32 and 0.97 respectively. This suggests a misclassification of some of the edge pixels as background. However, most of the background pixels were classified correctly. It gave an accuracy of 94% which is still considerable and a correlation parameter of 0.291 indicating an almost blurred image.

The Prewitt edge detector can be rated as having the best processing time performance of 106.8196 seconds. This attribute makes it to be computationally less expensive in terms of processing time. It gave sensitivity of 0.3349 and specificity of 0.974 respectively. Consequently, it displayed a high level of accuracy of 95% based on its simplicity and easy programming factor and correlation parameters of 0.307 indicating a very low ability to preserve the edge maps.

The LoG detection technique gave a processing time of 144.5313 seconds, which was the longest processing time. This suggests that it is time expensive. It gave a sensitivity and specificity of 0.64 and 0.98 respectively. This suggests that it correctly classified almost most of the edge pixels and almost all of the background pixels correctly. It gave a higher sensitivity too, signifying a superior performance compared to the other algorithms. It gave an accuracy of 96%, which was the highest out of all the algorithms and a correlation parameter of 0.56, which was as well the best correlation result. Therefore, apart from the processing time and sensitivity, LoG gave the best performance. Consequently, the outstanding performance of LoG in some of the metrics is indicative of the presence of the smoothing stage which tremendously reduces noise in the images.

The Petrou-Kittler detection technique gave a processing time of 23.3 seconds, which was second lowest. It gave a sensitivity and specificity of 0.965 and 0.362 respectively. This suggests that this edge system tends to correctly classify most of the edge pixels as edge. The specificity measure suggests that the Petrou-Kittler edge detection misclassified many of the background pixels as edges. It also had a lower accuracy of 38%. The correlation parameter was 0.36. This edge technique performed very well on the sensitivity measure.

The Multiscale Directional Filter Bank technique gave a processing time of 112.9289 seconds, which was one of the lowest, because it was subjected to various edge labeling algorithms to obtain effective edge map. It gave a sensitivity and specificity of 0.988 and 0.348 respectively. This suggests that this edge system tends to correctly classify most of the edge pixels as edge. A specificity of 0.348 suggests that the Multiscale Directional Filter Bank edge detection misclassified many of the background pixels as edges. It also had a lower accuracy of 37%, which was the lowest of all. The correlation parameter was 0.33. This edge technique gave the best sensitivity out of all. This was also demonstrated over other operators in [10] that described an edge detection using Multiscale Directional Filter Bank (DFB).

In the comparative analysis of the operators, the overall assessment was considered based on the cumulative values of all the metrics as shown in the last column of Table 2. This assessment showed the outstanding performance of LoG, having the highest performance in accuracy, Specificity and Correlation Parameter while Petrou-Kittler had the highest in processing time and DFB ranked first in sensitivity as indicated in Table 2. Canny and Sobel operators had similar overall performances and closely followed by the Prewitt operator. The Laplacian of Gaussian and the Petrou-Kittler detectors also had close performances.

It is worthwhile to state at this level that the operator with the least processing time value was actually ranked the best for having run the system within the least possible time. This has led to the revaluation of the operators and the performance metrics values as shown in Table 2. A performance index was ranged between 1 and 7 suggesting the degree of responsiveness of the seven operators with respect to the evaluating parameters. The most effective operator to an evaluating parameter was assigned 7 while the least was assigned 1. On this basis, the most outstanding operator was determined.

The edge maps in Figure 7 represent samples from 1,200 iris images. The quality of these samples characterizes the efficacy of each of the operators in the detection of edges in iris images. The outstanding performance of LoG by producing better quality and most visible images compared to other operators could be attributed to its overall performance.

5. CONCLUSION

Several works had been reviewed on the use of edge detector for detection of edges in different types of images, but in this study, Canny, Sobel, Robert, Prewitt, Laplacian of Gaussian, Petrou-Kittler and Multiscale Directional Filter Bank detectors were used on samples of iris images and subsequently followed by the Comparative analysis results of the different algorithms. Some of these methods showed considerable level of performance in the detection of edges, as the major goal of this study was to determine the most effective method using iris image. Since edge detection is the basis for image processing, it is therefore highly imperative to have an understanding of the characteristics, capabilities, strength and weaknesses of operators to employ in image processing when required.

Asides providing first-hand information on the operators, the study has also classified the operators on the basis of their strength and capabilities. It has unequivocally provided a basis for selecting the best operators most especially in handling applications like pattern recognition, optical character recognition, computer-aided medical diagnosis and other related image processing procedures, in order to achieve maximum results.

The performance metrics adopted in the study had also been able to unveil the specific characters of the operators. Convincingly, the introduction of some other performance indicators may further classify the operators.

Sequel to the adoption of the seven edge detector methods on selected Iris images and their consequent performances to detect edge accurately which was further evaluated using the adopted metrics, the reliable and unbiased assessments had proven Laplacian of Gaussian edge detector the best method. This performance could be hinged on its better detection ability especially in noisy condition among other factors. This also consolidates the importance of smoothing stage in edge detection processes.

From this study, it is recommended that Laplacian of Gaussian edge detector could be adopted as a preferred operator during edge detection. Also, in order to establish the identity of an unknown individual during Biometric Identification, the most effective operator for image processing should be adopted.

Finally, from the comparative analysis, it was observed that each operator performed considerably well under various condition.

REFERENCES

[1] D. Saini and S. R. Vijay. (2012): “Biometrics Identity Authentication in Secure Electronic Transactions”, IJCSMS *International Journal of Computer Science & Management Studies*, Special Issue of Vol. 12, June 2012 ISSN (Online): 2231 –5268.

[2] M. Unser. (2002): “Splines: A perfect fit for signal and image processing, ”*IEEE Signal Process. Mag.*, No. 11, pp. 22–38.

[3] Jain, K. Anil.; Ross, Arun (2008): "Introduction to Biometrics". In Jain, AK; Flynn; Ross, A. *Handbook of Biometrics*. Springer. pp. 1–22. ISBN 978-0-387-71040-2. *Journal of Applied Computer Science & Mathematics,* no. 12 (6) /2012.

[4] T Linberg. (1998): “Edge Detection and Ridge Detection with Automatic Scale Selection”, *International Journal of Computer Vision*, 30(5), pp. 117-154R. Nicole, "The Last Word on Decision Theory," *J. Computer Vision,* submitted for publication. (Pending publication).

[5] D. Marr and E. Hildreth. (1980): “Theory of Edge Detection”. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, Vol. 207, No. 1167. (29 February 1980), pp. 187-217.

[6] Raman Maini and. A. Himanshu. (2012): Study and Comparison of Various Image Edge Detection Techniques. *International Journal of Image Processing (IJIP)*, Volume (3) (1) pp. 8-12.

[7] M. Petrou. and J. Kittler. (1991): “Optimal edge detectors for ramp edges,” IEEE Trans. *Pattern Anal. Machine Intell.*, vol. 13, pp. 483–491,.

[8] Z. Othman, M. Rafiq, and Kadri (2009): "A Comparison of Canny and Sobel Edge Detection in MRI Images". Computer Science Biomechanics and Tissue Engineering Group and Information System. pp. 133-136.

[9] Z. Xianglong., Z. Jingbo and W. Kuoing. (2014): "Edge Detection method of Iris Image", *International Conference on E-education, E-business and Information Management.* pp. 65-68.

[10] S. Anand, T. Thivya. and Jeeva (2012): "Edge detection using multi scale Directional Filter Bank (DFB)". *International Journal of Applied Information Systems*, vol.4 (IJAIS) – ISSN: 2249-0868. pp. 21-27.