

PROCESSING BIOMEDICAL IMAGES USING LOW LEVEL ALGORITHM

¹Elias O.G, ²Olajide, M.S, ³Olaleke, J.O and ⁴Adetunbi A.O

^{1,2,3} Computer Science Department, Adeyemi College of Education, Ondo

⁴ Federal University of Technology, Akure.

ABSTRACT

Different biomedical images can be captured, stored, and utilized sufficiently to appraise some human features. These images can be critically evaluated using some techniques to excavate tiny, hidden and distinct physical human parts. Some biomedical analysis can be done using morphological operations such as basic statistics, de-noising, edge detection and segmentation to synthesize and analyze in great detail the various parts of the human system. This ultimately assists to draw viable inferences which have the specific biomedical values to undertake numerous decisions. This paper attempts to process biomedical images specifically brain images with low level algorithms with a view to creating a landscape for experts to aid their practice.

Keywords: low level algorithm, biomedical images, edge detection, de-noising, segmentation

1.0 INTRODUCTION

Image processing is a rapid growing aspect of computer science. Its growth has been prompted by technological advances in digital imaging, computer processors and mass storage devices. The primary purpose of image processing is to convert images to information. Image processing is used to improve the quality of images by removing noise from the images, highlighting interested features, and separating the object of interest from the background. Image processing can be defined as the manipulation of an image for the purpose of either extracting information from the image or producing an alternative representation of the image. There are numerous specific

motivations for image processing, but many fall into the following categories: (i) to remove unwanted signal components that are corrupting the image and (ii) to extract information by rendering it in a more obvious or more useful form. Image processing allows the use of much more complex algorithms. These algorithms are categorized into three basic classes which include the low level algorithms, the middle level algorithm and the highest level algorithms (Seamann, 2003).

According to Seamann (2003), the low level algorithms deal directly with the unprocessed and noisy pixel values examples include de-noising and

edge detection. The low level algorithms are also characterized by the fact that both its inputs and outputs are images (Rafael *et al.*, 2004). The middle level algorithms utilize the results of low level algorithms, and it is also characterized by the fact that its inputs generally are images, but its outputs are attributes extracted from those images. Tasks such as segmentation and edge linking are achieved via middle level algorithms, while the highest level are those methods which attempt to extract semantic meaning from the information provided by the lower levels, for example, handwriting recognition. Image processing is applied in several areas of human endeavours. Such applications include remote sensing, cartography, and medicine.

In medicine in particular, images comprise the majority of acquired and processed data, for example a large number of images of various imaging modalities such as Computer Tomography, Magnetic Resonance Imaging, and Ultrasound are produced daily and used to support clinical decision making. A typical radiology department generates about 100,000-10,000,000 images per year (Chandan *et al.*, 2008). Consequently, the medical field relies on images to store, display, and provides information.

A lot of research work has been carried out on processing of biomedical images. However, in spite of the extensive research in biomedical image processing, the field of biomedical image

processing is still at the stage of infancy (Chandan *et al.*, 2008). This is mainly due to the fact that biomedical image data is not as profuse as standard image dataset and usually the biomedical images pose some other inherent challenges such as being noisy (Chandan *et al.*, 2008). This noisy nature of biomedical images poses threats to the decision making in the health sector.

Consequently, in this paper biomedical image specifically the brain image is processed with low level algorithms specifically de-noising and edge detection with a view to helping medical practitioners to aid their practice as well as decision making.

2.0 CONCEPT OF IMAGES AND IMAGE PROCESSING

An image defined in the real world is considered to be a function of two real variables, for example, $a(x, y)$ with a as the amplitude (e.g. brightness) of the image at the real coordinate position (x, y) (Ian *et al.*, 1998). The amplitudes of a given image will be either real numbers or integer numbers. The latter is usually a result of a quantization process that converts a continuous range to a discrete number of levels, the result of this is the production of a digital image. A digital image can be considered as a discrete two dimensional function that has been quantized over its domain and range (Seeman, 2003).

Most images (either real or discrete) exist

on a rectangular grid, consisting of Y rows and X columns (Seamann, 2003), and the amplitude of the image at any pair of coordinates (X,Y) is called the intensity or gray level of the image at that point (Rafael *et al.*, 2004). The resolution of a digital image is defined as $X*Y$, the product of the two coordinates. Each distinct coordinate in an image is called a pixel, (picture element) which is stored as an array. A single pixel represents a value of light, intensity or colour. Associated with each pixel is a number known as Digital Number (DN) or Brightness Value (BV) which depicts the average radiance of a relatively small area within a scene (Kumar, 2000). There are many types of pixels. For instance, thermal images measure bands of the electromagnetic spectrum such as infra-red or radio, or heat while each pixel in the case of three-dimensional volume images are called a voxel (Seamann, 2003). Associated with each pixel is a number known as Digital Number (DN) or Brightness Value (BV) which depicts the average radiance of a relatively small area within a scene (Kumar, 2000).

Image processing can be defined as a process of extracting meaningful and useful information from an image. The applications of image processing include astronomy, ultrasonic imaging, remote sensing, video communications and microscopy, among innumerable others. Image processing offers advantages in cost, speed, and flexibility (Silver, 2000) to the fields. Certain tools

are central to the processing of digital images. These include mathematical tools such as convolution, Fourier analysis, and statistical descriptions, and manipulative tools such as chain codes and run codes (Ian, 2006).

Image processing is applied in areas such as military services, medicine and financial institutions. Its use in medical applications cannot be quantified. This is because medical care and research are literally the most vital part of science for humans, because no man is immune to physical ailments and biological deterioration (Darcy *et al.*, 2008).

3.0 Low Level Algorithms

The low level algorithms are those techniques which deal directly with the raw and noisy pixel values of an image. The low level algorithms employed in this paper are de-noising and edge detection.

3.1.1 De-noising

Images generally are prone to different types of noise. There are several ways that noise can be introduced into an image, depending on how the image is created. For example: If the image is scanned from a photograph made on film, the film grain is a source of noise. Noise can also be the result of damage to the film, or be introduced by a scanner.

The reduction of noise present in images is an important aspect of image processing (Musoko, 2005). De-noising can be defined as a procedure to

recover an image that has been corrupted by noise. There are different types of noise that can affect an image. These include:

a. Additive noise: If $F'(x,y)$ is the noisy digitized version of the ideal image, $F(x,y)$ and

$N(x,y)$ the noise function, then the additive noise can be defined as

$$F'(x,y) = F(x,y) + N(x,y)$$

Additive noise is independent of the pixel values in the original image (Seamann, 2003).

b. Impulse noise: The source of impulse noise is usually the result of an error in transmission or an atmospheric or man-made disturbance.

c. Multiplicative noise, or speckle noise, is a signal dependent form of noise whose magnitude is related to the value of the original pixel.

d. Gaussian Noise: The most common type of noise and the one that is mostly encountered is the Gaussian noise.

e. Speckle or salt and pepper noise: This noise is caused by errors in data transmission.

The goal of de-noising is to produce the best possible estimate $\hat{F}(x,y)$ of the original image

$F(x,y)$. In image de-noising, wavelets analyses are usually employed. Wavelet analysis is simply defined as the process of decomposing a signal into shifted and scaled versions of a mother

(initial) wavelet. It is also the process of reassembling a decomposed signal or image into its original form without loss of information (Musoko, 2005). An example of a simple wavelet function is called the Haar wavelet. Haar wavelet is discontinuous, and it resembles a step function. Other wavelets functions include sym, and db. Thresholding is also a technique used for image de-noising. There are two main approaches of thresholding (Musoko, 2005). These are:

a. Hard thresholding: This deletes all coefficients that are smaller than the threshold and keeps the others unchanged.

b. Soft thresholding: This deletes the coefficients under the threshold, but scales the ones that are left.

The measure of success in de-noising is usually an error measure $E(\hat{F}(x,y), F(x,y))$ between the original $F(x,y)$ and the estimate $\hat{F}(x,y)$. The Mean Square Error (MSE) function is commonly used to determine the success of de-noising because it has a simple mathematical structure that is easy to compute. The Mean Square Error function states that for a discrete image signal $y(m,n)$ and its approximated estimate $\tilde{y}(m,n)$ representing the original image and the de-noised image respectively:

$$M-1 \quad N-1$$

$$MSE = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (y(m,n) - \tilde{y}(m,n))^2$$

$$MN \quad m=0 \quad n=0$$

(Musoko, 2005)

Another way to determine the quality of a de-noised image is the Peak Signal to Noise Ratio (PSNR) which is given by:

$$\text{PSNR} = 10 \log_{10} (\text{Imax} / \text{MSE})$$
 where Imax is the maximum intensity value (Musoko, 2005).

3.1.2 Edge Detection

An edge is a significant change in the grayscale values between adjacent pixels in an image. Edge detection describes the general problem of determining the location, magnitude and orientation of boundaries (segment) in an image. Edge detection takes an intensity image I as its input, and returns a binary image BW of the same size as I , with 1's where the function finds edges in I and 0's elsewhere. There are six techniques for edge detection. This includes:

- a. The Sobel method: This finds edges using the Sobel approximation to the derivative. It returns edges at those points where the gradient of I is maximum.
- b. The Prewitt method: This finds edges using the Prewitt approximation to the derivative. It returns edges at those points where the gradient of I is maximum.
- c. The Laplacian of Gaussian method finds edges by looking for zero crossings after filtering I with a Laplacian of Gaussian filter.
- d. The Roberts method finds edges using the Roberts approximation to the derivative. It returns edges at those points where the

gradient of I is maximum.

- e. The zero-cross method finds edges by looking for zero crossings after filtering the image I with a specified filter.
- f. The Canny method finds edges by looking for local maxima of the gradient of I . The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds, to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges.

4.0 Methodology

We employed Matrix laboratory (Matlab) for the implementation of the low level algorithms: de-noising and edge detection on a brain image. The Haar, sym and db wavelet functions were employed in the de-noising process and the results from the wavelets were compared. We used MSE and PSNR to determine the quality of the de-noised brain images. We employed the Sobel method for the edge detection process.

5.0 Experimental Results for De-noising of the Brain Image

For our test experiments, we added Gaussian noise to our original image by using the syntax "imnoise" in Matlab. The syntax used in adding the noise is shown below:

$$J = \text{imnoise}(I, 'gaussian', m, v)$$
 where I = the

intensity image, m and v the mean and the variance of the image respectively.

The original and noisy images are shown below:

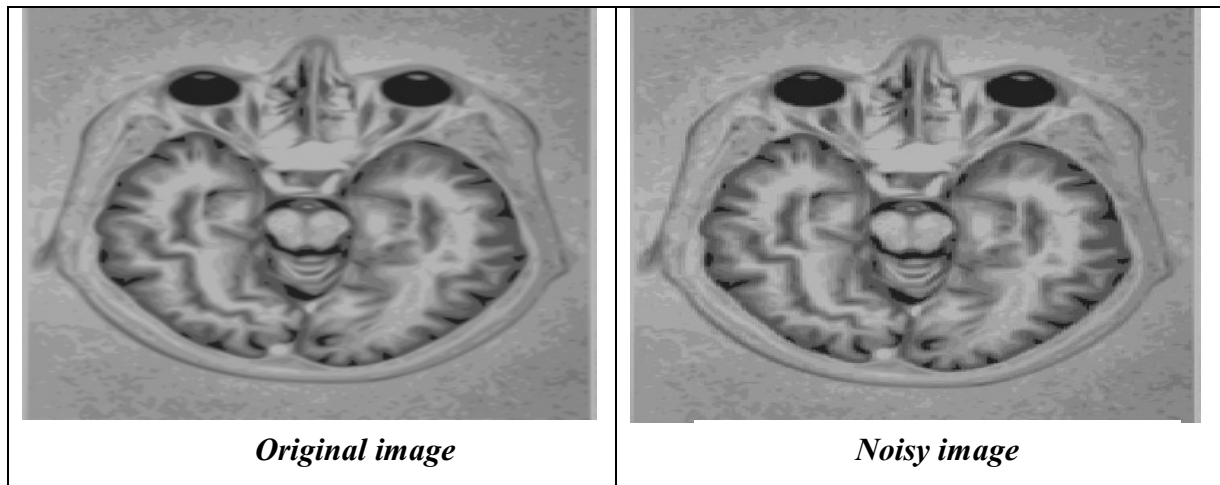


Figure 1

Figure 1

The noisy image was de-noised using three different wavelets, haar, sym and db using hard thresholds. The de-noised images obtained from the different wavelet functions (haar, sym and db)

were compared. MSE and PSNR error criteria were used to assess the performance of the wavelet functions on the images as well as the quality of the de-noised images. Their numerical results are summarized in the tables below:

Wavelet functions	MSE	PSNR
Haar	0.00256	19.56
Syn6	0.00124	30.02
db	0.00143	29.56

Table 1

The MSE and PSNR values from the experimental results showed that db wavelet yields lower Mean Square Error (MSE) and higher PSNR value

compared to other wavelet functions. The simplest, Haar wavelet produces the worst results as evidenced by the higher MSE and lower PSNR

values thus it is not effective in removing noise in biomedical images.

5.1 Experimental Results for Edge Detection of the Brain Image

We employed the Sobel method for the edge detection process. This was implemented in Matlab. Firstly, we converted the original image to an intensity image I and used the following syntax to detect the edges in the image

```
u = imread('C:\Users\olaronke\Desktop\brain.png'); -  
This line reads the image from its source  
>> I = .2989*u(:, :, 1)... This section converts the  
image to a gray scale or intensity image  
+.5870*u(:, :, 2)...  
+.1140*u(:, :, 3);  
>> figure; colormap (gray(256)); image(I)-  
This line displays the intensity image  
>> BW = edge(I,'sobel') This line finds the  
edges in the brain image
```

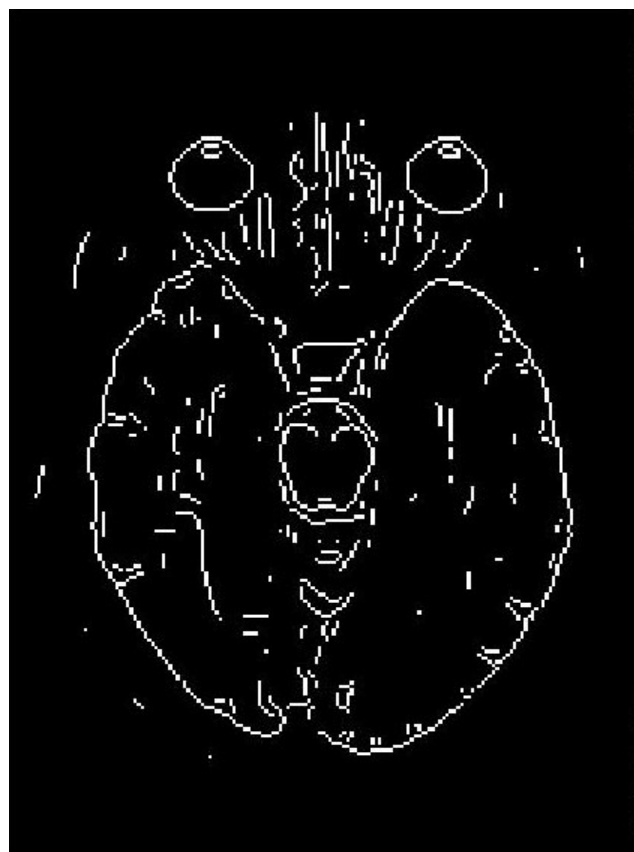


Figure 2-The Result of the Edge Detection Using the Sobel Method

Edge detection is an effective tool for many machine vision applications. The result of edge detection can provide information about the

location of the boundaries of objects and the presence of discontinuities. It can also aid practitioners to detect tumor and cancerous cells

in the image.

Conclusion

This paper is devoted to the processing of a brain image using low level algorithms specifically de-noising and edge detection. These algorithms were implemented respectively in Matlab. The trade-off between noise elimination and detail preservation (de-noising) was analyzed using the MSE and PSNR criteria. Thus comparisons between the performances of various wavelet functions haar, db and sym were deduced using these criteria. Db wavelet function performed better than the other wavelets functions in the de-noising process. Sobel method was employed for the edge detection in the brain image. The result can be used to reveal, diagnose or examine diseases.

References

Chandan K. and Fahima A. (2008). Retrieval and Ranking of Biomedical Images using Boosted Haar Features. *Department of computer Science at Wayne State University, Detroit*

Darcy, A. D., Nitesh, V. C., and Nicholas, B. (2008) Predicting Individual Disease Risk Based on Medical History. *University of Notre Dame. CIKM'08, October 2630, 2008, Napa Valley, California, USA.*

Ian T. Young Jan J. Gerbrands Lucas J. and van Vliet, C. (2006). Fundamentals of Image Processing. *Delft University of Technology, Netherlands*
Indian Institute of Remote Sensing, Dehra Dun.

Kumar, M. (2000). Digital Image Processing. *Photogrammetry and Remote Sensing Division*

Musoko, V. (2005). Biomedical Signal and Image Processing. A PhD Thesis. Chemical and Process Engineering. *Institute Of Chemical Technology, Prague Department of Computing and Control Engineering.*

Rafael, G. And Richard, W (2004). An Introduction to Matrix Manipulation In Matlab: A Support Document For Digital Image Processing. Second Edition. *Prentice Hall Upper Saddle River, Nj 07458.*

Seemann, T. (2003). *Digital Image Processing Using Local Segmentation*. Ph.D Thesis. *School of Computer Science and Software Engineering, Faculty of Information Technology Monash University Australia.*

Silver, B. (2000). *An Introduction to Digital Image Processing*. Cognex Corporation, Modular Vision Systems Division Natick

