DIRECT GRAYSCALE VERSUS BINARISATION FOR TOEPRINT MINUTIAE DETECTION AND PERSONAL RECOGNITION: A COMPARATIVE STUDY

Samuel Enemakwu Obaje
Department of Computer Engineering, Federal Polytechnic, Offa, Nigeria. obajese@yahoo.com; 08035859639, 08078564332)

Abstract

It is observed that a large percentage of victims of accident and leprosy who have lost their fingers cannot participate conveniently in the contemporary voting system in the world, which is characterized by the use of fingerprint. The objective of this study was to verify the validity of Toeprint for identification purposes. It was also to verify the faster, more robust and suitable means of minutiae detection between Binarisation and Direct Grayscale methods. From the experiment, it was observed that the average computation time for the Binarisation method is 28.86sec as against the 3.27sec for the Direct Grayscale method. Also the average error percentage for the Binarisation method is 71.18% and 21.68% for the Direct Grayscale method. The Quality Index values obtained during the experiment reveals the fact that Direct Grayscale is faster, more accurate and robust than the Binarisation. The conclusion is that the Toeprint when used for biometric recognition and forensic application purposes can compete significantly with the usual fingerprint biometric system.
Keywords:*Binarisation, Direct Grayscale, Morphology, Toeprint, Fingerprint, Quality Index, Identification, Authentication*

Introduction

Toe printing is the process of obtaining an impression of the papillary ridges of the toes for the purpose of identification. Although dactylics have not really gone into study of the Toeprint, it is evident that the toes, just like the fingers, do have ridges that could be studied. The First extensive collection of fingerprint records was probably Francis Galton's collection which was made in the late 19th century. Sir Francis Galton is an English scientist who specialized in the study of heredity. After an extended investigation, Galton established conclusively, the two basic facts on which fingerprint identification rests. These facts are one, individuality, i.e. the fingerprint is unique to an individual, and two, persistence, i.e. the basic characteristics of the fingerprint do not change with time. These facts according to Jain, Hong, and Bolle, (1997) could be traced to other similar findings before now, but no one could come out with a very solid conclusion as Galton's. The toes by observation also possess similar characteristics to the fingerprint, so they could be used for personal identification as well. The Toeprint classification is as shown in Fig. 1.

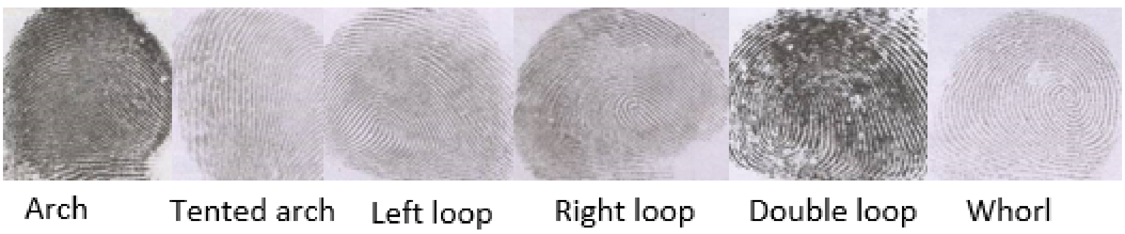

Fig. 1: A Classification of Toeprint,(Obaja, 2010).

The five toes can be identified by the positional name. For example, the thickest toe which is more prominent than the other four is called the big toe, followed by the index, then middle, next is fourth toes and finally little toe, as shown in Fig. 2.

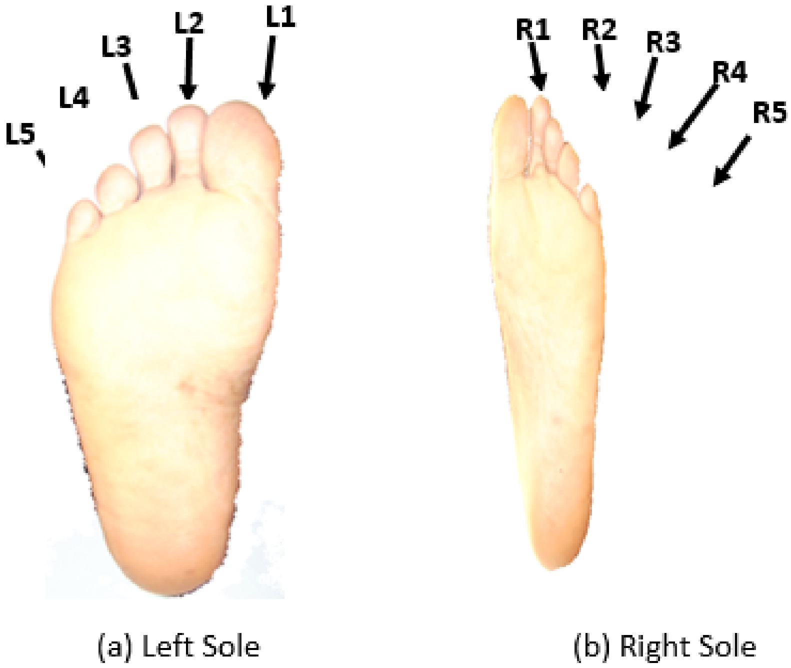

Fig 2: Toeprint by name. Obaje (2010

Key to Fig. 2

(a) Left Sole (L1 = Big Toe, L2 = Index Toe, L3 = Middle Toe, L4 = Fourth Toe, L5 = Little Toe)

(b) Right Sole (R1 = Big Toe, R2 = Index Toe, R3 = Middle Toe, R4 = Fourth Toe, R5 = Little Toe)

The primary objective of the present study is to determine the possibilities of employing Binarisation and Direct Grayscale as methods of identification of subjects of Purdah, victims of leprosy and victims of accident; especially during election. Purdah, a religious practice, makes it difficult to make use of the fingerprint for identification of some women in some parts of Nigeria. Also, victims of leprosy, accident, and natural causes, who have lost all or most of their fingers, cannot be identified through their fingerprints. Other problems this work stands to solve are the problems relating to the creation of resourceful Toeprint database management and file handling system. It is observed that an extensive study on toeprint has not been carried out in the past. Therefore there is no existing toeprint database or filling system for the database.

BINARIZATION METHOD

This method of minutiae extraction consists mainly of five stages namely; Image Enhancement, Ridge Extraction, Binarisation, Thinning, and Minutiae Extraction.

Image Enhancement

This idea simply means, removing of noise and sharpening the ridges using a Gabor filter, an even symmetry Gabor filter has the following general form in the spatial domain.

$$g(x, y, T, \varphi) = exp\left(-\frac{1}{2}\left[\frac{x_\varphi^2}{\sigma_x^2} + \frac{y_\varphi^2}{\sigma_y^2}\right]\right) cos\left(\frac{2\pi x_\varphi}{T}\right) \quad (4)$$

$$x_\varphi = x \cos\varphi + y\, sin\, \varphi \quad (5)$$

$$y_\varphi = -x\, sin\, \varphi + y\, cos\, \varphi \quad (6)$$

Source: Hong, Wan, and Jain (1998)

Where φ is the orientation of the derived Gabor filter, and T is the period of the sinusoidal plane wave.

Ridge Extraction

The objective of the Ridge Detection Algorithm is to separate ridges from valleys in a given toeprint image. A more reliable property of the ridges in a toeprint image is that the gray level values on ridges attain local minimal along a direction normal to the local ridge orientation.

To achieve this, a normalization process is desired. Assuming a grey-level input image I is defined as an N x N matrix, where I (i,j) represents the intensity of the pixel at ith row and jth column. The mean and variance of a grey level image is calculated according to Hong et al (1998) using the following equations 1

$$M(I) = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i, j)$$

$$VAR(I) = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(I(i, j) - M(I))^2 \quad (1)$$

Respectively

Hong et al (1998) also specified that the normalized image should then be generated by computing G (i,j) as the normalized grey-level value at pixel (i,j). Using equations 2

$$G(i, j) = \begin{cases} M_0 + \sqrt{\frac{VAR_0\,(I(i,j) - M)^2}{VAR}} & \text{, if } I \text{ (i,j) > M} \\ | \\ | \\ | \\ M_0 - \sqrt{\frac{VAR_0\,(I(i,j) - M)^2}{VAR}} & \text{, otherwise} \end{cases} \quad (2)$$

Where M0 and VAR0 are the desired mean and variance respectively

The local ridge orientation at point (x,y), defined as the angle θxy that the image ridges, crossing through an arbitrary small neighborhood centered at (x,y), form with the horizontal axis; the estimation of local ridge orientation is as follows:

The gradient g(xi,yj) at point (xi,yj) of the image I is computed as a two dimensional vector [gx(xi,yj), gy(xi,yj)] using the Sobel convolution masks. Here gx and gy components are the derivative of I at (xi,yi) with respect to y and x axis respectively.

Applying the method as proposed by Ratha, Karu, Chen, and Jain, (1996) the dominant ridge orientation in a local neighbourhood is computed using equations 3

$$Vx(i,j) = \sum_{u=i-8}^{i+8} \sum_{v=j-8}^{j+8} 2gx(u,v)gy(u,v),$$

$$Vy(i,j) = \sum_{u=i-8}^{i+8} \sum_{v=j-8}^{j+8} gx^2(u,v) - gy^2(u,v),$$

$$\theta(i,j) = \frac{1}{2} archtan\left(\frac{Vy(i,j)}{Vx(i,j)}\right) \qquad (3)$$

The implementation of the equation requires that the derivatives of the image should be taken at different points. Unfortunately differentiating an image is a noise- sensitive operation.

Binarisation

Peihao, Chia-Yung, and Chaur-Chin, (2007) proposed a binarisation method in which the image classifies each pixel into either "ridge" (darker) or "valley" (lighter). From an enhanced image {B(i, j)}, binarisation operation is achieved according to the following rules:

(a) Assign "ridge' to pixel (i,j) if B(i,j) ≤ Pk, where Pk is the Kth percentile of histogram of {B(i,j)}, e.g. k = 25 in this case.

(b) Assign "valley" to pixel (i,j) if B(i,j). ≥ P50,

(c) The remaining pixel {(i,j)} are classified as "valley" if B(i,j) ≥ T5x5, and 'ridge" otherwise; where T5x5 is the 30th percentile of 5x5 pixel values surrounding the pixel (i,j).

Thinning

The objective of the ridge step is to obtain a thinned image using morphological filters on binary images where all the ridges are to become only 1- pixel thick. This is simply achieved by conducting erosion until no further changes occur in the image, (Zhang & Fu, 1994). One major approach is to iteratively delete edge points from the region until just the skeleton remains. For a point to be deleted, the following conditions must hold:

- The point is not an endpoint
- the removal does not break the connectedness of the skeleton; and
- the removal does not cause excessive erosion of the region, (Maltoni, Mario, Jain & Prabhakar, 2003).

A contour point p0 is defined to have at least one pixel in its 8-pixel neighborhood that belongs to the background. A contour point is marked for deletion in the first pass if all of the following conditions hold for its 8-pixel neighbourhood:

- The number of neighbours that belong to the region must be between 2 and 6
- All neighbours that belong to the background must be connected
- p2, p4 or p6 must belong to the background

- p4, p6 or p8 must also belong to the background.

If all conditions are met the point is marked for deletion. However, the point is not deleted until all region points are processed to ensure that the data structure is not changed during the execution of the step. For a point to be marked for deletion in the second pass of the algorithm the first two conditions still must hold, but third and forth condition are changed as follows: p2, p4 or p8 must belong to the background and p2, p6 or p8 must also belong to the background.

P8	P1	P2
P7	P0	P3
P6	P5	P4

Fig. 3: Maltoni et al's (2003) 8-pixel Neighborhood Used in Thinning Algorithm.

"Fig.3" locations are considered as follows;

P0 = (x, y). p1= (x, y-1). P2 = (x+1, y-1). P3 = (x+1, y). p4 = (x+1, y+1). P5 = (x, y+1). P6 = (x-1, y+1). P7 = (x-1, y). p8 = (x-1, y-1)

Fig. 4 (a)

Fig. 4 (b)

"Fig. 4(a)" above shows the thinned ridge before spike removal, while "Fig. 4 (b)" shows the configuration of the thinned ridge after spike removal and gaps filling (the white dots in "Fig. 4 (b)" indicate the filled potions). Source: Maltoni et al (2003)

Minutiae Detection

Maltoni et al (2003) claim that once the thinned ridge map is available, the ridge pixels with three ridge pixel neighbours are identified as Ridge bifurcations and those with one ridge pixel neighbour are identified as Ridge endings. From a thinned image, each ridge pixel is classified into one of the following categories according to its 8-connected neighbour. A ridge pixel is called:

(a) An isolated point, if it does not have any 8 connected neighbour.

(b) An ending, if it has exactly one 8 connected neighbour.

(c) An edge point, if it has two 8- connected neighbours.

(d) A bifurcation, if it has three 8- connected neighbours; and

(e) A crossing, if it has four 8- connected neighbours.

The crossing number approach for Minutiae extraction is one of the major ways of achieving this. Mathematically in equation (7), crossing number of pixel 'p' is defined as half the sum of the differences between pairs of adjacent pixels defining the 8-neighborhoods of 'p', i.e.

$$cn(p) = \frac{1}{2}\sum_{i=1,...,8} |val(p_{i \bmod 8}) - val(p_{i-1})| \qquad (7)$$

Where p0 to p7 are the pixels belonging to an ordered sequence of pixels defining the 8-neighborhood of p and val (p) is the pixel value.

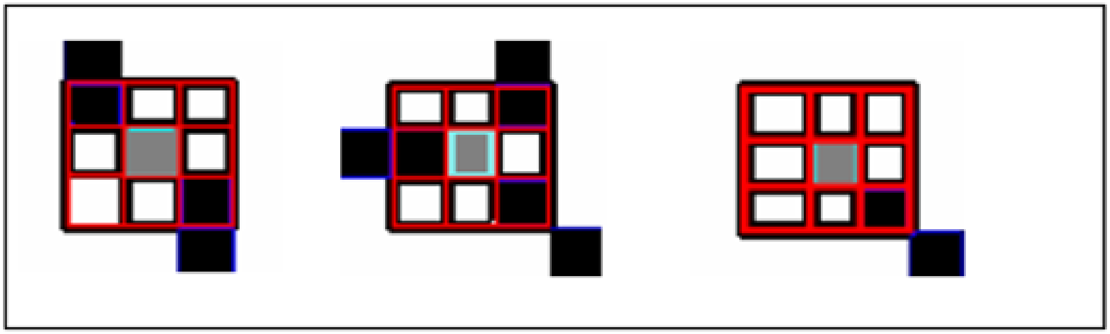

Fig.5: cn (p)=2,cn (p)=3 and cn (p)=1 representing a non-minutiae region, a bifurcation and a ridge ending. Source: Maltoni et al (2003).

Crossing numbers 1 and 3 correspond to ridge endings and ridge bifurcations respectively. An intermediate ridge point has a crossing number of 2. The minutiae obtained from this algorithm must be filtered to preserve only the true minutiae. The different types of false minutiae introduced during minutiae extraction include spike, bridge, hole, break, spur, ladder, and Misclassified Border areas.

The number of minutiae in a given area is also limited. Therefore, the minutiae density must also be kept in check. In order to filter out these false minutiae, a 3 level filtering process is applied:

Level 1: Removes the false ridge endings created as a result of the application of minutiae extraction algorithm at the ends of the thinned image.

Level 2: Removes the first five types of minutiae mentioned above using the rule based morphological minutiae filtering approach.

Level 3: This stage limits the maximum number of minutiae present in the thinned image to a pre-specified threshold.

DIRECT GRAYSCALE MINUTIAE DETECTION

To detect minutiae using the direct grayscale method, Mario and Maltoni (1997) suggests the following algorithm:

The algorithm starts by computing a point (ic, jc) belonging to the ridge line nearest to the starting point (is, js). This operation can be carried out as follows:

a. Compute the tangent direction φs in (is, js).

b. Section S defined as direction (is, js), along direction φs + □□/2 and with length 2σ + 1.

c. Regularize the section determined in step b and compute all the local maxima.

d. Choose, from the local maxima determined in step c, the local maximum (ic, jc) nearest to (is, js).

e. Sectioning can be achieved by intersecting S with a cutting plane parallel to the z direction

f. The section set Ω((it, jt), Φ, σ) centered in (it, jt), with direction Φ = φc + π/2, and length 2σ + 1 pixels, is defined as, Ω = {(i,j) | (i,j) Є I, (i,j) Є segment ((istart, istart),(iend,jend))} meaning, Ω = {(i,j) is orthogonal of which (i,j) is a subset of the image, I(i,j), is also a subset of segment (starting point), to segment (end point).

Where,

$$(i_{start}, j_{start}) = (\text{round}\ (i_t - \sigma.\cos\phi), \text{round}\ (j_t - \sigma.\sin\phi))$$
$$(i_{end}, j_{end}) = (\text{round}\ (i_t + \sigma.\cos\phi), \text{round}\ (j_t + \sigma.\sin\phi))$$

$$round(X) = \begin{cases} [X + 0.5]\ if\ X \geq 0 \\ [\ X - 0.5]\ otherwise \end{cases}$$

Fig. 6: below, shows an example of the ridge line starting point:

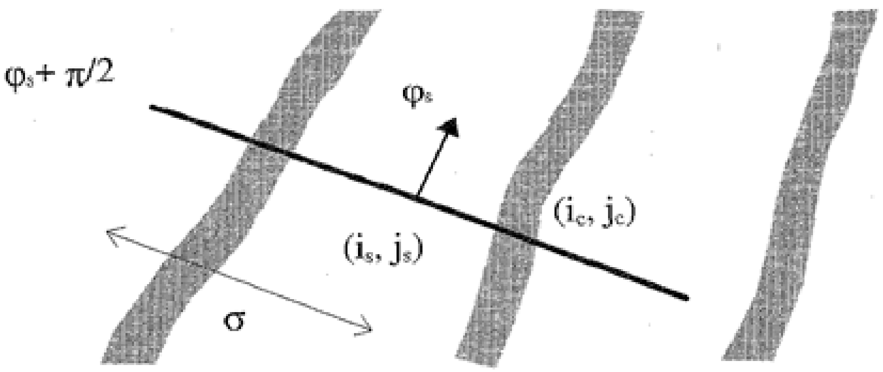

Fig. 6: Mario et al's (1997) Points (ic, jc) belonging to the ridge line nearest to the starting point (is, js).

METHODOLOGY

45,680 Toeprint were acquired for the work, the process employed was the ink dabbed method, as is shown in Fig. 7.

Fig. 7: *A Toeprint Capture process*

To develop minutiae detection and coding system, a personal computer with Pentium IV Intel Microprocessor support motherboard, dual cored, with 250 Gigabytes of Hard disk memory space,1014 Megabytes of Random Access Memory, with a processor speed of 1.60GigaHetz, and 32-bit Windows 7-Operating System was utilised. Furthermore, the system had on it a Microsoft Visual Studio where the C++ compiler, a computer language was found. A programming language is used for the development and testing of the software routine. The computer monitor is 17" wide, and the images scanned with a flatbed Hewlett Packard scanner, model G2710, were stored using the bitmap file format, which was made available by windows 'paint-brush' software that is available on the computer. The flowchart as developed by the researcher for the minutiae extraction system is shown in Fig. 8.

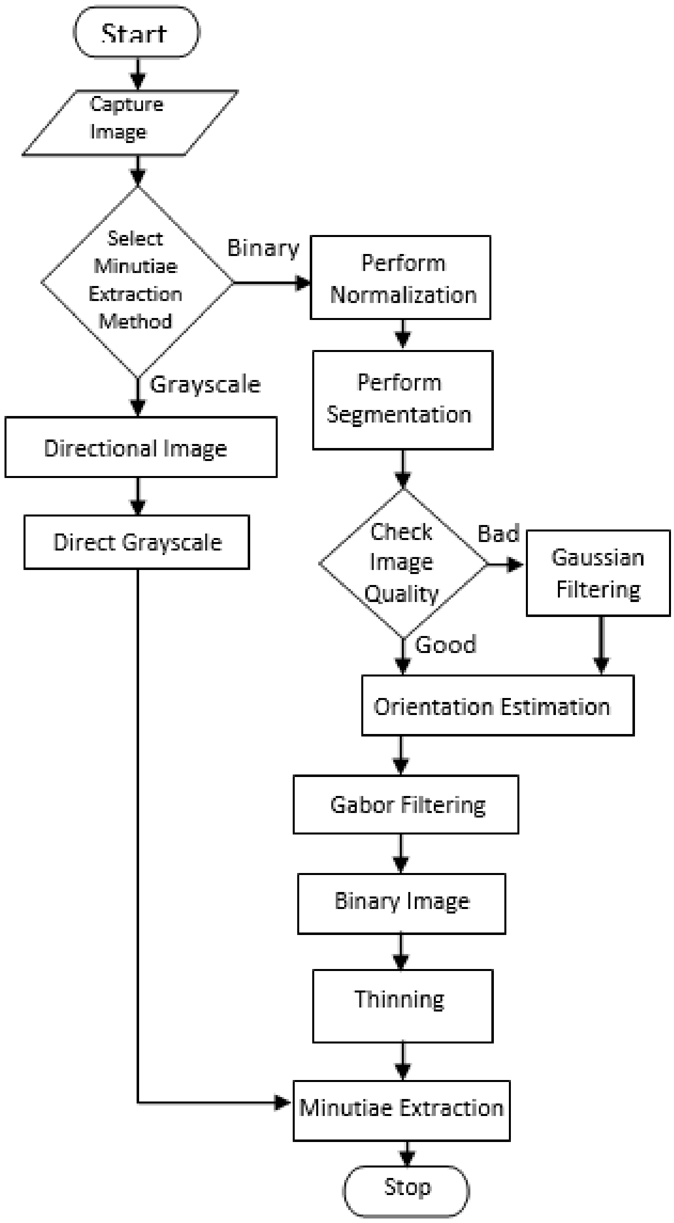

Fig. 8: A Flowchart for Toeprint Minutiae detection System.

The user interface of the system is shown in Fig. 9. The user interface allows the user to load a print image, perform pre-processing and minutiae detection, with a choice of either using the direct grayscale or binary method. The system performs the following tasks in three stages; and in the order listed below:

Stage 1
a. Load a print image from the database and find its relevant parts.
b. Enhance the image to ease the minutiae extraction.

This is the first stage and the common part. The other two stages that follow are made use of, depending on the choice that the researcher decides to make.

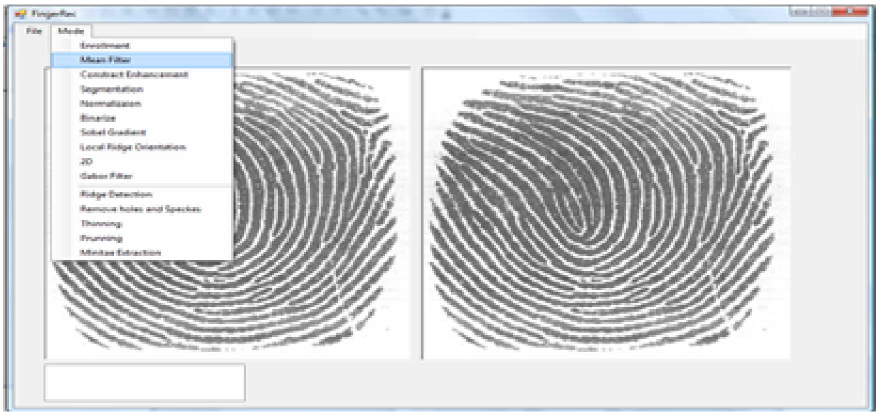

Fig. 9: *A System User Interface*

Stage 2

a. Transform the grayscale image to black and white.

b. Thin the ridgelines of the black and white image to one pixel width to ease the minutiae extraction.

c. Extract minutiae information from the thinned binary image, and store.

Stage 3

a. Follow the ridgeline in the grayscale image until they end or bifurcate.

b. Mark the ridgeline ending and bifurcation as minutiae points.

Performance Analysis

The minutiae extraction system that was developed was tested on a database sample set consisting of Toeprint images drawn from the created database. The sample set was discovered to exhibit different degree of image quality. The sample set has the following composition: A (100 x 8 images) and set B (10 x 8 images) for evaluating and training, respectively.

Five Toeprint (numbered 1 to 5) was carefully selected. To ensure that it involves the five classes; left loop, right loop, tented arch, arch, and whorl were used as the sample set. Table 1 shows the average computational time for the process.

Time (Seconds)		
	Binarization	**Direct Grayscale**
Segmentation	0.51	-
Normalization	0.14	-
Directional image	0.55	0.55
Smoothing	0.58	-
Binarization	0.16	-
Thinning	21.82	-
Minutiae detection and filtering.	0.096	2.72
Total time	23.86	3.27

Table 1: *A Toeprint Average Computational Time.*

Table 2 reports the results in terms of Undetected minutiae U, and False minutiae (does not exist but marked) F. Table 3 was drawn to report the average error percentage for the entire sample set. The plot, Fig. 10 compares the average error percentage with the average computational time obtained in the two approaches.

Toeprint serial number	Minutiae through Manual inspection	Direct Grayscale (A)		Binarization (B)	
		U	F	U	F
1	25	0	2	5	31
2	36	1	4	4	21
3	22	2	5	8	58
4	17	0	7	6	26
5	14	3	5	3	102

Table 2: *Toeprint Automatic Minutiae Detection*

"U" and "F" in Table 2 denote the number of undetected minutiae and, false minutiae, respectively.

Table 3: Average Error Percentage

	A	B
undetected minutiae (U)	5.67%	13.30%
False minutiae (F)	16.01%	57.88%
Total error	21.68%	71.18%

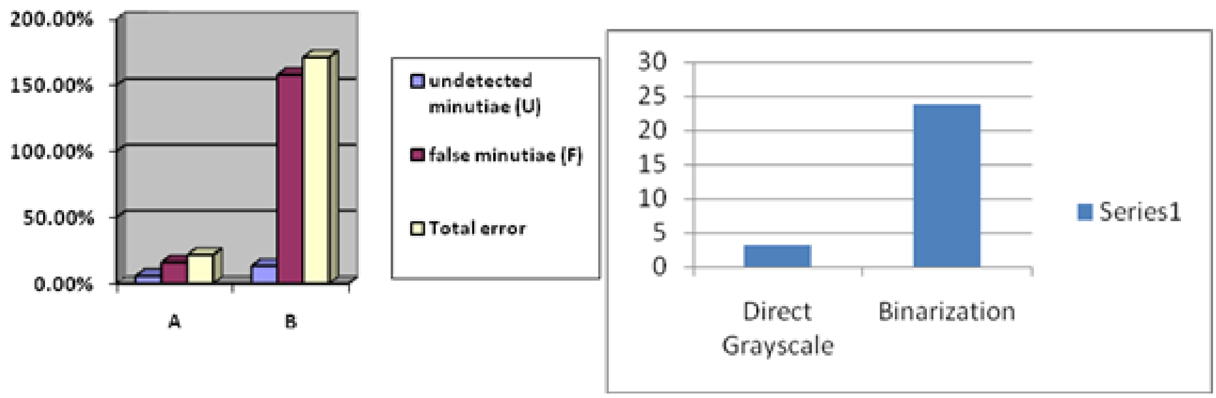

Fig. 10: *A Comparison between Average Error Percentage and Average Computational Time for both approaches in Toeprint.*

Image #	Grayscale	Binary	Image #	Grayscale	Binary	Image #	Grayscale	Binary
1	0.93	0.41	5	0.64	0.12	8	0.92	0.64
2	0.88	0.59	6	0.79	0.32	9	0.86	0.32
3	0.76	0.25	7	0.82	0.47	10	0.79	0.46
4	0.71	0.35						
Direct Grayscale								
Mean = 0.81			**Variance = 0.0009**			**Standard Deviation = 0.03**		
Binarization								
Mean = 0.39			**Variance = 0.0025**			**Standard Deviation = 0.05**		

Table 4: *Toeprint images. quality index values, mean, variance and standard deviation*

Findings and Conclusion
Findings:

The following findings were drawn from the experiment.

- The average error percentage (undetected minutiae), were compared for both approaches A and B, and it was discovered that "approach B" is on the higher side.
- The average computational time of "approach B" is also discovered to be higher than that in "approach A".
- There is a larger false minutia noticed in "approach B", especially in bad images than in "approach A". This is because more efforts will be required to enhance image quality before false minutiae can be reduced in such a case.
- The quality index of approach A is also discovered to be higher than that of "approach B".

Conclusion

It is therefore a conclusion of this study that the Toeprint can be used for automatic recognition just like the fingerprint. And the direct grayscale method is by far faster than the binarisation method of minutiae detection. The study also concluded that the toeprint authentication strategy as proposed in this study be adopted for identification purposes during election in Nigeria, especially for victims of accident or leprosy.

Recommendations

- The Toeprint was discovered to have characteristics similar to that of the fingerprint; it is therefore recommended that it can be used for voting, by people that have no fingers, for women in Purdah and for forensic applications.
- Direct grayscale method is also recommended for greater speed and performance on minutiae detection algorithm rather than methods that involve binarisation and thinning because those methods are considered to involve time consuming procedures.
- The database we created contains about 45,680 Toeprints scanned with image size of 256 X 256, a resolution of 500 dpi and stored in bitmap format. Interested researchers can therefore make use of the database for further research work in the field. However, the data may prove difficult to use but since they all have same characteristics, adequate tuning of a sample on your algorithm will make it accept the entire database for usage.

References

1. Hong, L. (1998). Automatic Personal Identification Using Fingerprints. Unpublished PhD Thesis. Michigan State University, USA.
2. Hong, L., Wan, Y., and Jain, A. K. (1998). Fingerprint Enhancement Algorithms and Performance Evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, no. 8, pp. 777-789.
3. Jain, A. K., Hong, L., and Bolle, R. (1997). On-Line Fingerprint Verification. IEEE Transaction on Pattern Analysis and Machine Intelligence. Vol. 19, no 4, pp. 302-314.
4. Mario, D., and Maltoni, D. (1997). Direct Gray-Scale Minutiae Detection in Fingerprints. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol.19, no.1,pp. 27-39.
5. Maltoni, D., Mario, D., Jain, A. K., and Prabhakar, S. (2003). Hand Book of Fingerprint Recognition. New York: Springer.
6. Obaje, S. E. (2010). Automatic Fingerprint and Toeprint Recognition for Personal Identification and Forensic Application. Unpublished PhD Thesis. Department of Electrical, University of Ilorin.
7. Peihao, H., Chia-Yung, C., and Chaur-Chin, C. (2007). Implementation of an Automatic Fingerprint Identification System. IEEE EIT Proceedings pp.461.
8. Ratha, N. K., Karu, K., Chen, S., and Jain, A. K. (1996). A Real time Matching System for Large Fingerprint Databases. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 18, no. 8, pp. 799 813.
9. Zhang, S., and Fu, K. S. (1994). A Thinning Algorithm for Discrete Binary Image. In Proceedings of International Conference of Computers and Applications. Beijing, China. pp. 879 886.