Indian Journal of Science and Technology, Vol 8(29), DOI: 10.17485/ijst/2015/v8i29/85285, November 2015
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Feature Extraction with Thinning Algorithms for Precise Cretoscopy

J. Grace Hannah* and D. Gladis

Department of Computer Science, Presidency College, Chennai – 600005, Tamil Nadu, India;
hannahgrace161289@gmail.com

Abstract

The surge of technology and information has led to the need to escalate the protection of data. The security codes and pins are in jeopardy due to the profuse increase of hacking. A diacatholicon for authenticity is provided by verifying and identifying each person. Fingerprints have proved to be impressive and flawless with a gargantuan number of users. The existing work has dealt with the traditional methods of processing and matching fingerprints. The proposed work deals with enhancement and algorithms for thinning and minutiae extraction. The texture features such as energy, correlation coefficient and entropy have been implemented to accredit the thinning process. Three fingerprint images are taken for processing and comparison. Thinning algorithms such as Zhang-Suen's and Hall's have been used. The minutiae are unsheathed by the Crossing number and Summing up algorithms. The simulation results of the algorithms have been compared, and the precision rates of fingerprints have been procured.

Keywords: Fingerprint, Minutiae, Minutiae Extraction, Precision Rate, Texture Features, Thinning

1. Introduction

The birth of an individual vouchsafes ineradicable traits and idiosyncratic characteristics which befits a biometric system. Due to the colossal advantages, large number of modalities has been introduced. Fingerprints, for example, have been used for over one hundred years and, therefore, are generally well accepted as an extraordinary recognition technology. Other modalities such as Facial recognition, palm prints, Iris scans, voice recognition, DNA, signature, hand geometry and gait recognition[1] are also employed in many fields. The taxonomy of Galton's system classifies that the curvy lines which tend to form a loop, a whorl, a delta and an arch are called as ridges[3] as shown in Figure 1 in section 4. An arch is a pattern when ridges enter from one side, surges up in the center and exits on the other side of the finger. Loops are ridge patterns which enter from one side, make a rise and egresses on the same side from where it entered. A circular formation of the ridges is called as a Whorl. Apart from the basic ridge patterns, there are other subdivided patterns which occur. The basic arch pattern can also have a sharp uphill in the center, and is termed as a tented arch. The loops are further divided into ulnar, radial, double and center pocket loops. Concentric, spiral, imploding and composite whorls are a few sub patterns which are formed from the basic whorl pattern.

The extraordinary feature of a fingerprint is determined by a pivotal component called the minutiae[7,8]. Each minutiae may occur at different positions which vary from finger to finger. These minutiae points play a significant role in the post processing stages like the fingerprint matching. The Figure 2 in section 4, shows the various classifications of minutiae such as dots, bridge, enclosures, and trifurcations[9]. Fingerprint applications ranges from forensic departments for criminal investigations[2] to attendance systems and various other security systems. All fingerprint images go through the preprocessing steps for better enhancement and clarity[4,20]. A fingerprint with good quality is that which is clear and free from noise, a medium quality and a poor quality fingerprint would consist of relative amount of noise, scars and dirt[10]. But

*Author for correspondence

however good the quality of the fingerprint is, it should go through the process of preprocessing for accurate plotting of minutiae. Preprocessing involves three stages such as binarization, direction and thinning. Before a fingerprint image enters this preprocessing stage, it is checked for its type, whether it is a RGB or a grayscale image. A RGB image is that which defines the red, green and blue components of a single pixel in the matrix. The RGB image is otherwise called as a true color image[11]. A gray scale image consists of 256 values. It has values ranging from 1 to 255. If it is a RGB image, it is first converted to its grayscale image and then sent for the preprocessing channel. Binarization of the grayscale image involves converting an image with 256 values to a corresponding image having only 0's and 1's, where 0 represents the ridges and 1 is for furrows or valleys. Binary images are easy to store and manipulate, as each pixel is converted to a single valued pixel. A binary image is obtained by comparing each pixel value of a grayscale image with a threshold value. If the gray value of a particular pixel is lower than the threshold value, the value becomes 0; else the value is assigned as 1.

The next step in preprocessing is the direction map. The direction map helps in obtaining the respective region of interest and the flow of frequency in a fingerprint image. The direction map calculates the local flow orientation in each local window with size (block size x block size)[5,19]. The direction map differs from one fingerprint to the other.

Figure 3 (a) and (b) in section 4, shows the original grayscale image and the converted binary image respectively. The next step in preprocessing which is vital is the thinning process. In a binarized image, the ridges of the fingerprint are usually thick which makes it difficult for plotting of minutiae. Thinning is the process of reducing the width of each ridge to a one pixel value[6, 20]. A minutiae point is better extracted in a perfectly thinned image rather than in a binarized image. Skeletonization is a form of thinning where the pixels on the boundary of the image are removed but the continuity is preserved.

1.1 Thinning Algorithms

Zhang-Suen's algorithm[6,18] is an iterative process executed with many passes over each pixel of the fingerprint. This process is executed till there is no more change on the ridge width of the fingerprints. Zhang-Suen's algorithm is applied on a block of pixels in a chosen matrix, with a center value 1 and having 8 neighbors. The summation of all the values of the chosen matrix is given by B(p). A(p) is given by 8-neighborhood of the matrix p, containing exactly one 4-connected component of 1's. On the other hand the Hall's algorithm[6,18] is a parallel thinning algorithm that thins the ridges of the fingerprint and at the same time preserves the connectivity of the image. This algorithm is partially serialized by breaking it into two distinct sub-iterations. Unlike the Zhang-Suen's algorithm, the Hall's algorithm checks for the deletability of the pixel at the initial stage, and then the ridges are thinned pixel by pixel. B(p) and A(p) are the sum of all values and the number of 0 to 1 transitions respectively.

It has been observed that Zhang-Suen's algorithm gave a thinned image, whereas the Hall's algorithm produced a skeletonized image. The minutiae extracted for the two algorithms gave a result where the minutiae points were reduced profusely, thereby helping in fast processing of fingerprints. The Zhang-Suen's and Hall's algorithm are given in Tables1 and 2 in section 4.

2. Texture Features Analysis

Texture features of a fingerprint help in adding relevant and adequate information for a given fingerprint image. Three texture features like the Entropy, Correlation coefficient and Energy has been computed to show the effect of thinning on fingerprints. Entropy is classified as a statistical measure of randomness that can be used to characterize the texture of an input image[14]. Entropy is defined as:[11]

$$-\text{sum}\ (p.*\log^2(p)) \qquad (1)$$

In the above equation, p refers to the value obtained after the normalization of the image.

The 2-D Correlation Coefficient is computed in two steps: The first is to remove the noise in an image by applying a 2-D median filter[14]. The second step is to apply the correlation coefficient function[11] to the filtered image. The correlation coefficient for an image is given by

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2 \sum_m \sum_n (B_{mn} - \bar{B})^2\right)}} \qquad (2)$$

Where A and B are the mean values of A and B.

The retained energy in a given image is computed by using a wavelet toolbox. The input image is subjected to level 5 Daubechies wavelet decomposition[14,20]. The Shannon entropy is used with a threshold method called Balance Sparsity-norm to compress the image and thereby find the energy retained in the image after decomposition.

The values of each texture feature are obtained for images before and after thinning. It has been deduced that the values obtained after the thinning process is better than the values acquired before thinning. When the noise is reduced and the quality of the image is good and consistent throughout, the value of the texture features such as entropy and correlation coefficient is said to reduce, which is inversely proportional to the increase in energy retained in an image. This signifies that thinning is a mandatory process that revolutionizes a fingerprint quality. The results of the texture features are shown in Tables 3 and 4 in section 4.

3. Minutiae Extraction

The crucial feature found in every fingerprint which helps to match two fingerprints precisely based on their pixel position is termed as minutiae[12]. Minutiae extraction is the process of unsheathing these small points from the fingerprint[13]. In this paper, the terminations and bifurcations are the two types of minutiae which are extracted.

The thinning process plays a key role in the accurate plotting of minutiae. The thinned fingerprint image is split into N X N matrix. The centroid for each chosen matrix is found by using the sliding neighborhood operation[17]. The position of the centroid pixel varies for each matrix, depending on whether it is an odd matrix, an even matrix or a customized N X N matrix, chosen by the user. Once the centroid is found, its value is checked. If the value of the centroid is one, then the summation of all the pixels in the chosen matrix is done and then subtracted by 1[17]. But if the value of the centroid pixel is zero, then the matrix is deleted. This process is done for all the pixels in each chosen N X N matrix. Once the values of the respective pixels are found, they are compared to the values in the crossing number technique to check if the pixel is a ridge ending or if the ridge is branched. The simulations are done using a 3 X 3 matrix and the minutiae are extracted by comparing the pixel values with the crossing number method.

3.1 Crossing Number Technique

The crossing number method[15,16] is implemented to find the minutiae points in a fingerprint. The algorithm works for all types by having a number for each minutiae. For example, the number 0 is given for an isolated point or a dot, number 4 is given for a crossing ridge and so on. Similarly for Terminations the crossing number has a value 1 and for bifurcations the value is 3. Crossing Number is defined as half of the sum of differences between intensity values of two adjacent pixels[15]. It is given by

$$CN=1/2\Sigma^{8}_{i=1}[|P_i - P_{i+1}|] \qquad (3)$$

For any chosen pixel in the matrix, the 8 neighbors of the pixel are scanned in the anticlockwise direction. Figure 4 shows the centroid pixel with the value 1 and having one neighborhood which is termed as Termination. Bifurcation is given when the center pixel has the value 1 with three neighborhoods.

3.2 Summing up Method

A gray scale image goes through the pre-processing steps such as binarization and thinning. Once the fingerprint image is thinned, its ridges are in a one valued pixel. This helps in obtaining the accurate minutiae points. The summing up method involves the sliding neighborhood operation which helps to choose a 3 X 3 matrix. The centroid (center pixel) is chosen from the block matrix. The next step is to add all the values in the chosen matrix. The summation is performed for each pixel of the thinned matrix to obtain a new matrix consisting of minutiae points. The sum computed is then subtracted by 1, which gives the value of the pixel[17]. The neighborhood analysis of 1 neighbor, 3 neighbors for the center pixel 1, is considered as terminations and bifurcations respectively. For any pixel P of the chosen matrix, the summing up method is done in a clock-wise direction. The simulations done with this method are presented in the next section.

4. Results

Fingerprint images are taken from biometrics.idealtest.org and the simulations are carried out using MATLAB. The implementation of the thinning process is emphasized by computing the texture features of the fingerprint. The minutiae extraction is done in a graphical user interface and the results have been obtained successfully. The variations in texture features have been shown in the below two tables. Three fingerprint images were taken for observation to elucidate that texture values obtained after thinning proved to be better.

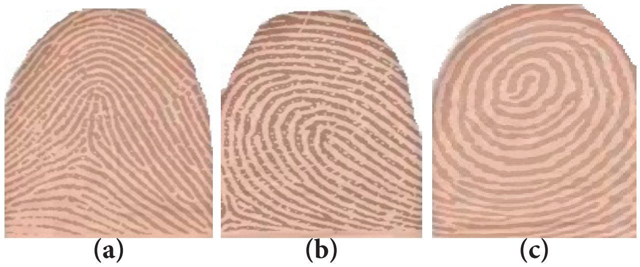

Figure 1. Basic ridge patterns. (a) Arch. (b) Loop. (c) Whorl.

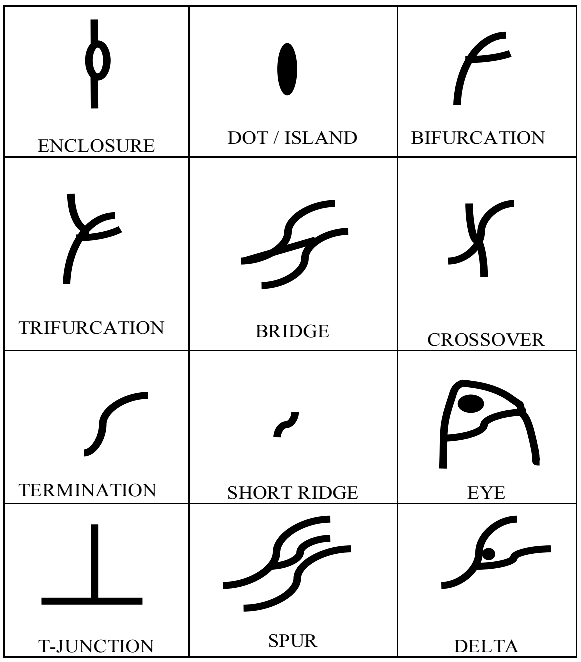

Figure 2. Minutiae.

Figure 3. **(a)** Grayscale image. **(b)** Binarized image.

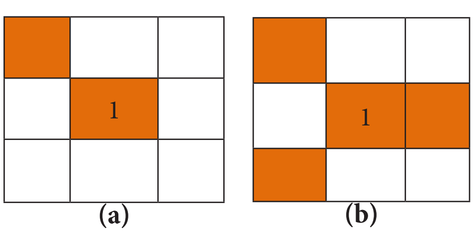

Figure 4. Minutiae extraction using crossing number. **(a)** Termination. **(b)** Bifurcation.

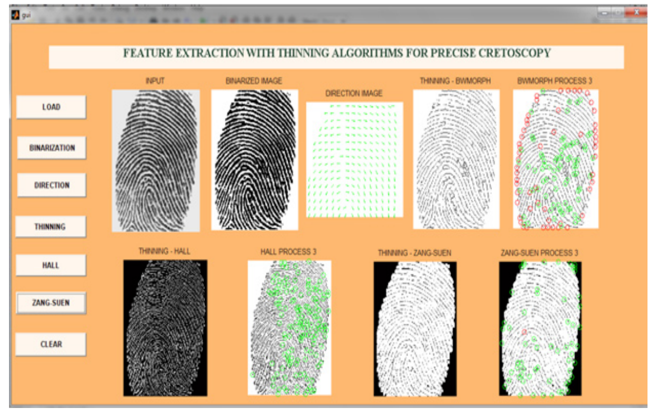

Figure 5. Thinning and minutiae extraction using crossing number algorithm.

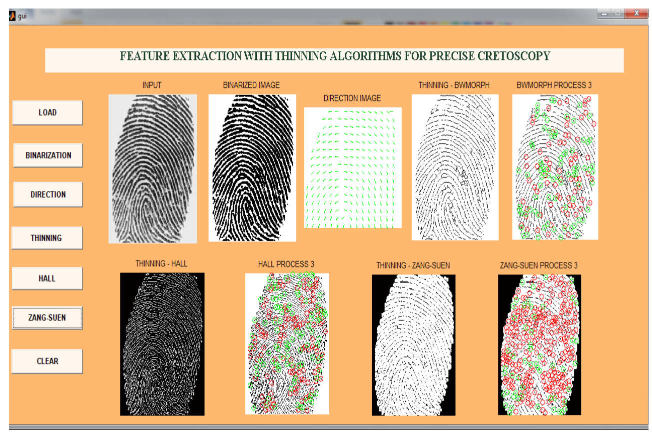

Figure 6. Thinning and minutiae extraction using summing up algorithm.

Table 1. Algorithm: Thinning by Zhang-Suen's algorithm

```
While points are deleted do
for all pixels p(i, j) do
if (2<= B(p) <=6) & (A(p) == 1)
            Apply one of the following
              a)   (i-1)*(i,j+1)*(i+1,j) = 0 in odd iterations
              b)   (i-1, j)*(i, j+1)*(i, j-1) = 0 in even
iterations
            Apply one of the following
              a)   (i,j+1)*(i+1,j)*(i,j-1) = 0 in odd iterations
              b)   (i-1,j)*(i+1,j)*(i,j-1) = 0 in even iterations
      then
         Delete p(i, j)
   end if
 end for
end while
end
```

Table 2. Algorithm: Thinning by Hall's algorithm

```
While points are deleted do
  for all pixels p(i,j) do
      Determine deletability of pixel
        if (1< B(p) < 7) & (A(p) = = 1)
             then p(i,j) = deletable
        end if
  end for
    for all pixels p(i,j) do
      if ((i-1,j) = (i+1,j) =1 and (i,j+1) is deletable)
& ((i,j+1) = (i,j-1) = 1 and (i+1,j) is deletable) &
((i,j+1),(i+1,j+1),(i+1,j) are deletable)
        then Do not Delete p(i,j)
      end if
    end for
end while
end
```

Table 3. Texture feature values before thinning

Images	Before Thinning En CC Er
Fp1	4.92 0.98 96.12%
Fp2	4.24 0.98 97.20%
Fp3	4.67 0.99 96.60%

Table 4. Texture feature values after thinning

Images	After Thinning Zhang-Suen's Hall's En CC Er En CC Er
Fp1	3.99 0.88 99.98% 3.99 0.62 99.97%
Fp2	3.73 0.86 99.99% 3.69 0.60 99.96%
Fp3	3.80 0.88 99.98% 3.68 0.61 99.95%

Table 3 and 4 shows the variations in the values which are observed before and after the thinning process. Fp1, Fp2 and Fp3 are the raw fingerprint images taken without the thinning process implemented on them.

The terms En, CC and Er, stands for Entropy, Correlation coefficient and Energy respectively. Table 4 shows the two thinning methods Zhang-Suen's and Hall's methods applied on the three fingerprint images. The values of the texture features such as Entropy and correlation coefficient decreases with the increase in the quality of the image, whereas the Energy retained in the image increases with the increase in the quality of the image. The results obtained after thinning significantly stipulate that it is an imperative process for a good quality image.

Figure 5 shows the step wise approach involved from loading an image to the minutiae extraction process in a GUI environment. The first step in the processing of a fingerprint is to load an image by applying ink over the finger and scanning it directly on the computer or by using a griaule's fingerprint scanner to obtain the fingerprint image. This raw image is binarized so that it is easily processed. The direction map is generated for the binarized fingerprint image. The arrow marks in the direction map indicates the direction of flow. The next process is to apply the morphological operation 'thin' to the binarized fingerprint image. This operation of thinning is done infinitely till there occurs no change in the fingerprint image. The minutiae points are extracted by using the crossing number algorithm and then the spurious minutiae are removed by the procedure of distance computation. The neighborhood operation is carried out on the binarized image to effectuate the Hall's thinning algorithm.

The similar procedure is carried out but with a difference that instead of using crossing number algorithm, the minutiae are extracted from the thinned fingerprints by applying the summing up algorithm. Figure 6 shows the minutiae points extracted by using the summing up method.

It can be deduced that in summing up algorithm, a relatively large number of minutiae appear when compared to the crossing number algorithm. When comparing the Zhang-Suen's and Hall's algorithm, the former was seen to be faster and also reduced a large number of false minutiae. The bifurcation points in Zhang-Suen's and Hall's algorithm are reduced. The variation in the minutiae points can be seen clearly in the results produced.

The precision rate[20] is defined to evince the accuracy of computation in each of the thinning algorithms used. And the precision rate is calculated by taking into account the TP and FN for the process. TP and FN stands for True Positive and False Negative. True Positive takes the value of the number of minutiae points extracted properly from the fingerprint. False Negative is the number of minutiae which are missed being detected from the fingerprint. The precision rate is given by

$$\text{Precision} = \frac{\text{TP X 100}}{\text{TP + FN}} \qquad (4)$$

The following table shows the precision rate for the thinning algorithms used. The computation has been done and the percentage for each fingerprint with respect to the thinning algorithm has been given in Table 5 and Table 6.

Table 5. Precision rate using crossing number algorithm

	Thinning Algorithms	
	Zang- Suen's	Hall's
Fp1	85.91%	84.11%
Fp2	80.64%	79.21%
Fp3	88.73%	86.23%

Table 6. Precision rate using summing up algorithm

	Thinning Algorithms	
	Zang- Suen's	Hall's
Fp1	75.39%	73.80%
Fp2	73.54%	72.84%
Fp3	75.00%	74.11%

5. Conclusion

The simulations are performed successfully by binarizing, thinning and after which the minutiae are extracted from the fingerprint image. The success of any fingerprint recognition strongly relies on the precision obtained after the thinning and the minutiae extraction phases of a fingerprint. Experimental results showed that the crossing number algorithm not only helped eliminate the spurious minutiae but also gave a higher precision rate with respect to the thinning algorithms. But the same cannot be said of the summing up algorithm as it failed in reducing the spurious minutiae as well as gave a much lesser percentage for the three fingerprints which are taken. In the thinning process, it can be inferred that Zhang-Suen's Algorithm has a higher precision rate and was much more efficient when compared to the Hall's algorithm. In order to obtain more accurate results, an improvement in the image binarization step and minutiae matching algorithm could be introduced with fingerprints with more noise such as scars and cuts. The texture features could be more extensively studied as a future work.

5.1 Sliding Neighborhood Operation

The centroid and the matrix to be chosen on the thinned image to classify the minutiae points are done with the help of the sliding neighborhood operation. A sliding neighborhood is an operation that is performed on one pixel at a time. The neighborhood is a rectangular block and is moved over one value of the pixel to the next in an image matrix. The sliding of the block is done from the left to the right of the matrix. The center pixel always has to have the value 1.

5.1.1 Even Matrix

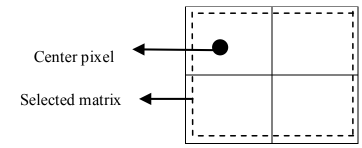

Figure 7. Representation of even matrix.

The left most pixel forms the center point in an even matrix.

5.1.2 Odd Matrix

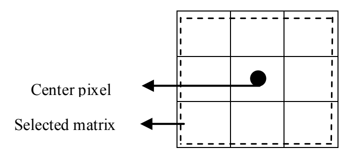

Figure 8. Representation of odd matrix.

The center pixel forms the center point of an odd matrix.

5.1.3 Matrix of N X N

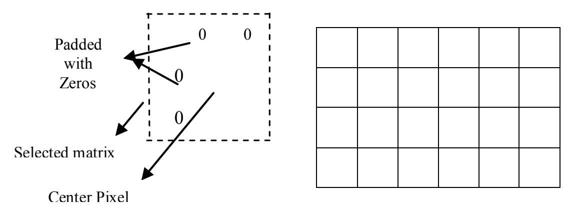

Figure 9. Representation of N X N matrix.

In a matrix of N X N, the pixel next to the left top pixel in the matrix forms the center pixel. If there exists no values for the first center pixel to be selected, then the neighboring pixels are padded with zero. Now the matrix is chosen and the corresponding values of the minutiae points are obtained by using the crossing number algorithm.

6. References

1. Ross AA, Nandakumar K, Jain AK. Handbook of multibiometrics. Springer.
2. FBI, Fingerprints. 2014 Jan. Available from: http://www.fbi.gov/about-us/cjis/fingerprints_biometrics/biometric-center-of-excellence/modalities/fingerprint
3. Forensic Science. 2014 Jan. Available from: www.all-about-forensic-science

4. Mario D, Maltoni D. Direct gray-scale minutiae detection in fingerprints. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997; 19(1):27–39.
5. Huang P, Chang CY, Chen CC. Implementation of an automatic fingerprint identification system; 1998.
6. Khanyile NP, Tapamo JR, Dube E. A comparative study of fingerprint thinning algorithms; 2011.
7. Patil N, Kulkarni G and Patil D. Fingerprint recognition for Library management. International journal of computation Engineering and Management. 2013 Jan; 16(1).
8. Maltoni D, Maio D, Jain AK, Prabhakar S. Handbook of fingerprint recognition. Springer; 2009.
9. Mali K, Bhattacharya S. Various aspects of minutiae as a fingerprint feature. International journal of Computer Science and Engineering Technology (IJCSET). 2002.
10. Maltoni D, Maio D. Fingerprint analysis and representation. IEEE transactions on pattern analysis and machine intelligence. 1997; 19(1):27–39.
11. Types of Images. 2014 Jan. Available from: www.mathworks.in
12. Prabhakar S, Jain AK, Pankathi S. Learning fingerprint minutiae location and type. Journal of Pattern Recognition. 2003; 36(8):1847–57.
13. Maddala S, Tangellapally SR. Implementation and evaluation of NIST biometric image software for fingerprint recognition [Thesis]; 2010.
14. Saha M, Chaki J, Parekh R. Fingerprint recognition using texture features. International Journal of Science and Research (IJSR). 2013; 2(12).
15. Raymond T. Fingerprint Image enhancement and minutiae extraction [PhD thesis]. School of Computer Science and software engineering, University of western Australia; 2003. p. 21–56.
16. Stoney DA. Distribution of epidermal ridge minutiae. American Journal of Physical Anthropology. 2005; 77(33):367 – 76.
17. Hannah GJ, Gladis D. Dactyloscopy and comparison of algorithms for efficacious minutiae extraction. International Conference on Advance Research in Engineering and Technology (ICARET); 2014. p. 52–7.
18. Hannah GJ, Gladis D. A juxtapose of algorithms for effective thinning integrated with minutiae extraction in fingerprints. International Conference on Communication and Computing (ICC); Elsevier Pvt Ltd; 2014. p. 8–15.
19. Hannah GJ, Gladis D. thinning algorithms with texture features for effective dermatoglyphic analysis. International Conference on Innovations in Contemporary IT Research (ICITR); 2015. p. 58.
20. Hannah GJ, Gladis D. Lophoscopy with thinning algorithms for inimitable fingerprint processing. International Conference on Information and Convergence Technology for Smart Society (ICICTS). 2015; 1(1):254–60.