A Conceptual Study on Fingerprint Thinning Process based on Edge Prediction

Krishna Prasad K.[1] & P. S. Aithal[2]

[1]Research Scholar, College of Computer and Information Science, Srinivas University, Mangaluru-575001, Karnataka, India

[2] College of Computer and Information Science, Srinivas University, Mangaluru-575001, Karnataka, India

E-Mail: karanikrishna@gmail.com

Type of the Paper: Research Paper.
Type of Review: Peer Reviewed.
Indexed In: OpenAIRE.
DOI: http://dx.doi.org/
Citation : https://scholar.google.com/citations?user=Wb7oZPYAAAAJ&hl=en

How to Cite this Paper:
Krishna Prasad, K. & Aithal, P. S. (2017). A Conceptual Study on Fingerprint Thinning Process based on Edge Prediction. *International Journal of Applied Engineering and Management Letters (IJAEML)*, 1(2), 98-111.
DOI: http://dx.doi.org/

International Journal of Applied Engineering and Management Letters (IJAEML)
A Refereed International Journal

A Conceptual Study on Fingerprint Thinning Process based on Edge Prediction

Krishna Prasad K.[1] & P. S. Aithal[2]

[1]Research Scholar, College of Computer and Information Science, Srinivas University, Mangaluru-575001, Karnataka, India

[2] College of Computer and Information Science, Srinivas University, Mangaluru-575001, Karnataka, India

E-Mail: karanikrishna@gmail.com

ABSTRACT

Biometric recognition encompasses numerous modern strategies. Among them, fingerprint reputation is taken into consideration to be the most effective approach for utmost security authentication. As industrial incentives boom, many new technologies for user identity are being advanced, each with its very own strengths and weaknesses and a potential area of interest marketplace. Fingerprint matching consists of a different process like filtering or preprocessing, binarisation, thinning or skeletonisation, postprocessing, feature extraction, and matching. Out of these fingerprint thinning or skeletonisation is one of the important processes in fingerprint identification or verification systems. Fingerprint thinning or skeletonisation is the manner or technique of lowering the thickness of every line of a fingerprint pattern or ridge pattern to just a single pixel width. After extracting the minutiae from the improved, binarised and thinned image some post-processing is carried out on this final fingerprint image to take away any spurious minutiae. The techniques on this class are of types–crossing number based and morphology-based totally. In this paper even though a new method for thinning is not proposed but a real attempt is made to explain the Edge prediction based thinning process. The Edge Prediction based Skelton formation is totally based on the conditional thinning set of rules, which is used to carry out thinning. The Edge Prediction based thinning process is explained with the help of workflow, algorithm, and flowchart.

Keywords: Thinning, Skeletonisation, Minutiae, Crossing Number, Edge Prediction Based Skelton Formation.

1. INTRODUCTION

The drastic changes in mobile and wireless based technologies and increasing number of applications and users demanded for high security concern, which leads to research on biometrics with a purpose to increase the security aspects and to minimize security threats. Even though biometric systems are not so easily vulnerable to security threats but some intelligent intruder can compromise the system using the information of biometric templates. So it's essential and necessary to develop non invertible, revocable and highly robust biometric templates [1-5].

Fingerprint thinning or skeletonisation is the manner or technique of lowering the thickness of every line of a fingerprint pattern or ridge pattern to just a single pixel width [6-7]. The necessities of a good thinning algorithm with respect to the fingerprint are

- The thinned fingerprint image received must be of single pixel width without discontinuities.
- Each ridge must be thinned to its center pixel.
- Noise and singular pixels have to be removed.

- Elimination of pixels should not cause elimination of true minutiae or after the completion of thinning method none of the pixels eliminated.

There are many techniques to be had in literature for skeletonisation or thinning process. After extracting the minutiae from the improved, binarised and thinned image a post processing is accomplished in this final fingerprint image to filter or remove any spurious minutiae.

Skeletonised technique of minutiae extraction is likewise called Skeletonisation-based minutiae extraction. Here again, pre-processing strategies are carried out to enhance or remove noise, the fingerprint image is segmented and binarised. The binarised image is then thinned using a set of rules that removes pixels from ridges until the ridges are one pixel length [8]). There are many methods available in literature for skeletonisation or thinning process [9-11]. After extracting the minutiae from the improved, binarised and thinned image some post processing is carried out on this final fingerprint image to take away any spurious minutiae. The techniques on this class are of types– crossing number based and morphology based totally.

Crossing number wide variety based, which is the most extensively used technique of minutiae extraction inside the skeletonised binary image class. This method is ideal over different methods due to its computational performance and intrinsic simplicity. In this method, a skeleton image is used in which the ridge run pattern is considered as a window with size of eight-connected. The nearby pixel of every ridge pixel in the image is scanned the usage of a 3×3 window from which the minutiae are extracted as shown in Figure 1. The crossing number may be used to categorise a ridge pixel as a finishing, bifurcation or non-minutiae point. As an example, a ridge pixel with a crossing-number of zero will correspond to an isolated factor and a crossing number of 4 correspond to a crossing factor.

P4	P3	P2
P5	P	P1
P6	P7	P8

Figure 1: 3 × 3 Neighbourhood

Jain et al., (1997) [12] have additionally performed minutiae extraction with the need of the skeleton image. Their approach entails the usage of a 3x3 window to verify the nearby area of each ridge pixel within the image. A pixel is then categorized as a ridge finishing if it has most effective one neighboring ridge pixel within the image, and categorized as a bifurcation if it has 3 neighboring ridge pixels. Therefore, it can be seen that this approach is very much like the crossing number technique.

Table 1: Properties of crossing number of Skeletonisation
(Source: Bansal et al., 2011 [21])

Crossing number	**Property**
0	Isolated point
1	Ridge ending point
2	Continuing ridge point
3	Bifurcation point
4	Crossing point

Fake or false minutiae may be added to the image because of elements including noisy image, and image artifacts created by the thinning or Skeletonisation process. Subsequently, after the minutiae are extracted, it's far essential to do post-processing which will validate the minutiae. Few examples of false minutiae structures include the spur, hollow, triangle and spike systems (Xiao & Raafat, 1991). It could be seen that the spur shape generates false ridge endings; whereas both the hollow and triangle systems generate false bifurcations. The spike structure creates a fake bifurcation and a fake ridge finishing point.

The majority of the proposed work for image post processing-processing after thinning, in the literature [13-15] are based on a series of structural policies used to discard spurious minutiae. For example, a ridge ending factor that is linked to a bifurcation point, and is below a sure or convinced threshold distance is removed. But, alternatively, then using a unique set of heuristics every time to do away with a unique kind of fake minutia, some processes include the validation of different varieties of minutiae right into a single algorithm. They verified the validity of each minutiae point by way of scanning the skeleton image and analyzing the local neighborhood across the minutiae. The algorithm is then able to cancel out fake or false minutiae primarily based on the configuration of the ridge pixels connected to the minutiae point.

Amengual et al., (1997) [16] considered low-level features or minutiae points in order to extract features of the fingerprint image. For the description and retrieval of minutiae, they used already available varieties of minutiae extraction method or techniques by modifying a little bit. Farina et al., (1999) [17] proposed set of algorithms for minutiae extraction from the fingerprint image.

Gnanasivam & Muttan (2010) [18] proposed preprocessing techniques for filtering and noise removal of the image before extracting features from the image. The preprocessing is done to acquire the vertical orientated fingerprint image followed through the center point of fingerprint pattern detection-core point and region of interest choice. Then characteristics extraction is performed in the extracted area of concern image.

Leung et al., (1991) [19] proposed a neural network primarily based approach to minutiae extraction where preprocessing strategies are first carried out to clean or remove the noises and then binary ridge pattern is thinned or skeletonised. Before applying neural network approach skeleton is ready for feature extraction. In a later stage, a multilayer perceptron concept of three layers is trained to extract the minutiae from the skeletonised image of the fingerprint.

The morphology-based minutiae extraction strategies are based on mathematical morphology (Humbe et al., 2007; Bansal et al., 2010) [20-21] in which the image is pre-processed with an intention to reduce the overhead of post-processing filtering. The image is pre-processed with morphological operators to do away with spurs, bridges and so on (Bansal et al., 2010) [21]. After which the authentic minutiae are extracted through the morphological hit or miss rework to extract original minutiae. The morphological operators are forming operators which permit the manipulation of shapes for identity and also the composition of objects and item capabilities. Morphological operators are essentially shaping operators and their composition permits the natural manipulation of shapes for the identity and the composition of objects and object-capabilities [22-26]. The approach develops structuring factors for exceptional forms of minutiae found in a fingerprint image to be utilized by the HMT to extract legitimate minutiae. Ridge endings are those pixels in an image which have only one nearby point in a 3x3 neighbourly located pixels or points. In this paper Edge Prediction based Skeleton formation method is discussed with its workflow, algorithm. The algorithm is analyzed using FVC ongoing 2002 datasets.

2. EDGE PREDICTION BASED SKELTON FORMATION

The Edge Prediction based Skelton formation is totally based on the conditional thinning set of rules (You & Wang, 2003) [27], which are used to carry out thinning. Mark the target point 1, the background as zero. The main idea is here to use, eight-neighbourhood and there may be at least one background pixel or point, defined as a boundary point. This method considers segmented image, $I_{segment}$ as input for this process.

The output from this method is skeletonised or thinned image, denoted as $I_{skeleton}$. Initially the size of the $I_{segment}$is calculated. In this method 3 × 3 frame is moved across every pixel of the image. If $I_{segment}$ (i, j) =1, then it signifies that particular pixel is not part of the image's foreground, it's just background pixel. Where i, and j represents index of row and column dimension of the image. Image is traced in row and column order from second row and column to till the second last row and column. If $I_{segment}$ (i, j) =1, then for each pixel, including that pixel, a 3 × 3 frame is created based on following equations. The frame image is referred as temp. $temp = I_{segment}\big((i-1{:}\,i+1),(j-1{:}\,j+1)\big)$

In above assignment statement, i and j represents row and column position of the $I_{segment}$ image. The shape of the frame for $I_{segment}$ (2, 2), image matrix is as follows. The actual pixel position is (2, 2), which is surrounded by 8 pixels in different eight directions. The central pixel is surrounded by a ring shape starting from (1, 1) , (1, 2), (1, 3), (2, 3), (3, 3), (3, 2), (3, 1), (2, 1), and (1, 1). The central pixel is checked with all eight neighbouring pixels. The image type is logical so it contains either zero or one as its intensity values. One represents background of the fingerprint image and zero represents foreground of the image.

(1, 1)	(1, 2)	(1, 3)
(2, 1)	(2, 2)	(2, 3)
(3, 1)	(3, 2)	(3, 3)

Figure 2: Example of 3 x 3 frame used in Edge Prediction based Skelton formation

Next we find values of these eight positions and store that values in a variable called, RE_{temp}. Generally Ring structure for any pixel position is created as follows,

Trace from first column of the temporary variable to column size value-1 for following two statements

$RE_{temp}(i) = temp(1, i)$ \\$RE_{temp}$→Ring Extracted for temp Matrix

$RE_{temp}\left((C_{temp} - 1) + (R_{temp} - 1) + i\right) = temp(R_{temp}, C_{temp} + 1 - i)$ \\$C_{temp}$→column size, variable i, is index of column.

Trace from first row of the temporary variable to row size value-1 for following two statements.

$RE_{temp}(C_{temp} + i) = temp(1, C_{temp})$

$RE_{temp}\left((C_{temp} - 1) + (R_{temp} - 1) + i\right) = temp(R_{temp} + 1 - i, C_{temp})$ \\$R_{temp}$→Row size of temporary variable, variable i, is index of Row.

Assign the temporary variable first element value to Ring Extracted for temp Matrix's last position (RE_{temp}), which is usually becomes (9, 9) position in 3 × 3 frame. This assignment statement is shown below.

$$RE_{temp}((C_{temp} - 1) + (R_{temp} - 1) + (C_{temp} - 1) + (R_{temp} - 1) + 1) = temp(1,1)$$

Consider For example, central pixel position as (2, 2). The pixels positions of the RE_{temp} matrix are assigned as shown in table 2.

Table 2: Example of Edge Vector used in Edge Prediction based Skelton formation

Sr. No	Edge Vector ($\boldsymbol{RE_{temp}}$)	Temporary Image formed from $\boldsymbol{I_{segment}}$ (temp)
1	RE_{temp} (1)	temp (1, 1)
2	RE_{temp} (5)	temp (3, 3)
3	RE_{temp} (2)	temp (1, 2)
4	RE_{temp} (6)	temp (3, 2)
5	RE_{temp} (3)	temp (1, 3)
6	RE_{temp} (7)	temp (3, 1)
7	RE_{temp} (4)	temp (2, 3)
8	RE_{temp} (8)	temp (2, 1)
9	RE_{temp} (9)	temp (1, 1)

Next, we find the NP corresponds to total number of points which is having logical value 1 or which contains background part of the fingerprint segmented image. Simply, NP is Temporary Variable to save the result of that corresponding calculation. NP can be obtained using following equation.

$$N_p = \frac{\left(\sum RE_{temp}\right)^2 - (RE_{temp}(1))^2}{\left(\sum RE_{temp}\right) + RE_{temp}(1)} \qquad \text{(Eq. 1)}$$

The RE_{temp} matrix is reshaped as 1 × 9 logical matrix. Next we find total number of terminating point around the central pixel. We consider a variable T_p, corresponds to terminating point. Last position of

RE_{temp} matrix contains value of first position itself. So we trace and find only eight neighbourhood pixel position values and at a time we consider contiguous two pixel position starting from first position. If first pixel position contains zero and next succeeding pixel contains one then it is marked as T_p. This is shown in following statement,

if $(RE_{temp}(p) = 0)$ & $(RE_{temp}(p+1) = 1)$ $T_p = T_p + 1$

Here p can take any value from 1 to 8. When RE_{temp} contains NP > =2 and NP<=6 and TP=1 and ($RE_{temp}(2) * RE_{temp}(4) * RE_{temp}(6)$ =0) and ($RE_{temp}(4) * RE_{temp}(6) * RE_{temp}(8)$ =0) , we have to store the current central pixel row and column position or index value to one temporary matrix denoted as F1 with two columns and n number of rows. Where n represents total number of central pixels which is having value 1 or which is a part of background pixels of the segmented image.
We trace eight neighbourhood pixel position values starting from 1 position to 8 through P. The statement can be expresses as

if (N_p>=2) & (N_p<=6) & (T_p==1) &$\left(RE_{temp}(2) * RE_{temp}(4) * RE_{temp}(6) == 0\right)$ & $\left(RE_{temp}(4) * RE_{temp}(6) * RE_{temp}(8) == 0\right)$ (Eq. 2)

F1 (F+1, 1) = i and F1 (F+1, 2) = j where F is a temporary variable with initial value as 0.
Increment the value of F as F= F+1.

1	1	1
1	1	1
1	0	0

1	1	0
1	1	0
1	1	1

1	1	1
1	1	0
1	0	0

1	1	1
1	1	1
0	0	1

0	0	0
1	1	0
1	0	0

1	1	1
0	1	1
1	0	1

0	0	1
0	1	1
1	1	1

1	0	0
1	1	0
1	1	1

1	0	0
1	1	0
1	1	0

Figure 3: Examples of few different possibilities where thinning function is repeated

Figure 3 shows examples of different possibilities where temporary matrix F1 is initialized with position of central pixel position. In general words, near by pixel of the central pixel, which contains foreground pixel has to be thinned. Usually red color represents central pixel.
All the central pixel positions are stored in temporary matrix F1, if it satisfies the Eq. 2. The F1 matrix corresponding to $I_{segment}$ image is reassigned with value 0, which is shown below.

$$I_{segment}\left(F1(i,1), F1(i,2)\right) = 0$$

The above statement is repeated from first row position of F1 to till last row position with fixed column size as 2. Next we create another temporary variable N_{F2} and initialize it with value zero. Next we repeat same process starting from, if $I_{segment}$ (i, j) =1 to $I_{segment}\left(F2(i,1), F2(i,2)\right) = 0$. Until condition (F > 0 or $N_{F2} > 0$) fails, repeat all above steps of for $I_{segment}$.

3. EDGE PREDICTION BASED SKELETON FORMATION ALGORITHM

Input: Segmented Image, $I_{segment}$
Output: Skeletonised Image, $I_{skeleton}$
Step-1: [R C]=size($I_{segment}$) \\ R→Row Size of skeleton Image; C→Column Size of skeleton Image and assign F=0; \\ F→Temporary variable
Step-2: for i=2 to R-1
Step-3: for j=2 to C-1
Step-4: if $I_{segment}(i,j) = 1$
Step-5: $temp = I_{segment}\big((i-1:i+1),(j-1:j+1)\big);$
Step-6: for i=1 to $C_{temp} - 1$
Step-7: $RE_{temp}(i) = temp(1,i)$ \\RE_{temp}→Ring Extracted for temp Matrix
Step-8: $RE_{temp}\left((C_{temp}-1)+(R_{temp}-1)+i\right) = temp(R_{temp}, C_{temp}+1-i)$; end step-6 for
Step-9: for i=1 to $R_{temp} - 1$ \\R_{temp}→Row size of temporary variable
Step-10:$RE_{temp}(C_{temp}+i) = temp(1, C_{temp})$
\\C_{temp}→Column Size of temporary variable
Step-11: $RE_{temp}\left((C_{temp}-1)+(R_{temp}-1)+i\right) = temp(R_{temp}+1-i, C_{temp});$
Step-12: end step-9 for loop
Step-13: $RE_{temp}\big((C_{temp}-1)+(R_{temp}-1)+(C_{temp}-1)+(R_{temp}-1)+1\big) = temp(1,1)$
Step-14: $N_p = \frac{(\sum RE_{temp})^2-(RE_{temp}(1))^2}{(\sum RE_{temp})+RE_{temp}(1)}$
\\N_p→ Temporary Variable to save the result of that corresponding calculation
Step-14a: for p=1 to size(RE_{temp}, 2)-1
Step-14b: if $(RE_{temp}(p) = 0)$ & $(RE_{temp}(p+1) = 1)$
Step-14c: $T_p = T_p + 1$; end if end for loop
Step-15: for p=1 to size(RE_{temp}, 2)-1
Step-16: if (N_p>=2) & (N_p<=6) & (T_p==1) &$\big(RE_{temp}(2)*RE_{temp}(4)*RE_{temp}(6) == 0\big)$&$\big(RE_{temp}(4)*RE_{temp}(6)*RE_{temp}(8) == 0\big)$
Step-17: F1(F+1,:)=[i j];
Step-18: F=F+1; end if
Step-19: end if loop in step-4; end for loop in step-2
Step-20: if (F>0)
Step-21: for i=1 to size(F1,1)
Step-22: $I_{segment}\big(F1(i,1), F1(i,2)\big) = 0;$
Step-23: end for loop end if loop
Step-24: $N_{F2} = 0$
\\ N_{F2}→Temporary Variable which will change for every iteration of for loop on step-25
Step-25: for i=2 to $R_{segment} - 1$
Step-26: for j=2 to $C_{segment} - 1$
Step-27: if ($I_{segment}(i,j) = 1$)
Step-28: $temp = I_{segment}((i-1:i+1),(j-1:j+1))$
Step-29: for i=1 to $C_{temp} - 1$
Step-30: $RE_{temp}(i) = temp(1,i)$
Step-31: $RE_{temp}\left((C_{temp}-1)+(R_{temp}-1)+i\right) = temp(R_{temp}, C_{temp}+1-i)$; end step-29 for
Step-32: for i=1 to $R_{temp} - 1$
Step-33:$RE_{temp}(C_{temp}+i) = temp(1, C_{temp})$
Step-34: $RE_{temp}\left((C_{temp}-1)+(R_{temp}-1)+i\right) = temp(R_{temp}+1-i, C_{temp});$

Step-35: end step-32 for loop
Step-36: $N_p = \frac{(\sum RE_{temp})^2-(RE_{temp}(1))^2}{(\sum RE_{temp})+RE_{temp}(1)}$
\\N_p→ Temporary Variable to save the result of that corresponding calculation
Step-36a: for p=1 to size(RE_{temp}, 2)-1
Step-36b: if $(RE_{temp}(p) = 0)$ & $(RE_{temp}(p+1) = 1)$
Step-36c: $T_p = T_p + 1$;
Step-37: end if end for loop
Step-38: if (N_p>=2) & (N_p<=6) & (T_p==1) &$\left(RE_{temp}(2) * RE_{temp}(4) * RE_{temp}(6) == 0\right)$&$\left(RE_{temp}(4) * RE_{temp}(6) * RE_{temp}(8) == 0\right)$
Step-38a: F2($N_{F2} + 1$,:)=[i j];
Step-39: $N_{F2} = N_{F2} + 1$
Step-40: end step-38 if; end step-27 if; end step-25 for loop
Step-41: if $N_{F2} > 0$
Step-42: for i=1 to R_{F2}
Step-43: $I_{segment}\left(F2(i,1), F2(i,2)\right) = 0$;
Step-44: end for loop; end if statement
Step-45: if (F>0 or $N_{F2} > 0$)
Step-46: $I_{skeleton} = repeat\ all\ the\ steps\ for\ I_{segment}$; end if

4. FLOWCHART FOR EDGE PREDICTION BASED SKELETON FORMATION ALGORITHM

This proposed algorithm for skeleton Formation is explained using flowchart. The input for this algorithm is segmented grayscale fingerprint image which is represented as $I_{segment}$. The final output is segmented image denoted as $I_{skeleton}$. The different workflows of the proposed algorithm are listed out below. The flow chart of this algorithm is shown in figure 4.

- Check each pixel of the segmented image for background part of the image
- Create a 3×3 frame from $I_{segment}$ by keeping each pixel at centre and surrounded by 8 pixels starting from second row and second column to till last row and column-1.
- Create a ring of nine pixels (8+starting pixels) and extract ring values.
- Find the number of edges from the extracted ring for each central pixel.
- Find the count of terminating points of the ridge by simply counting the edge number (Edge number should be 0 or 1 for terminating point)
- Reduce the width of the foreground image, by making central pixel value zero.

Analysis of Edge Prediction based skeleton formation Algorithm

The Edge Prediction based Skeleton formation algorithm is analysed by considering FVC ongoing 2002 datasets. Figure 5 shows segmented image and skeleton image for sample image 101_1.tif.
In figure 5, 101_1.tif is a sample fingerprint image taken from FVC ongoing 2002 DB1_B dataset, which is of size 388 × 374 pixels. This image is initially converted into 256 × 256, before filtering process. After segmentation this image is again resized into 256 × 148. The figure 5 (a) represents this segmented image, and 5 (b) represents, its skeleton image. The skeleton image size is again resized into 254 × 146. Same way, in figure 5, 101_5.tif is a sample fingerprint image taken from FVC ongoing 2002 DB1_B dataset, which is of size 388 × 374 pixels. This image is initially converted into 256 × 256, before filtering process. After segmentation this image is again resized into 256 × 148. The figure 5 (c), represents this segmented image, and 3.19 (d) represents, its skeleton image. The skeleton image size is again resized into 254 × 156.

	
Segmented image-101_1.tif	Skeleton image of (a)
	
Segmented image-105_1.tif	Skeleton image of (c)

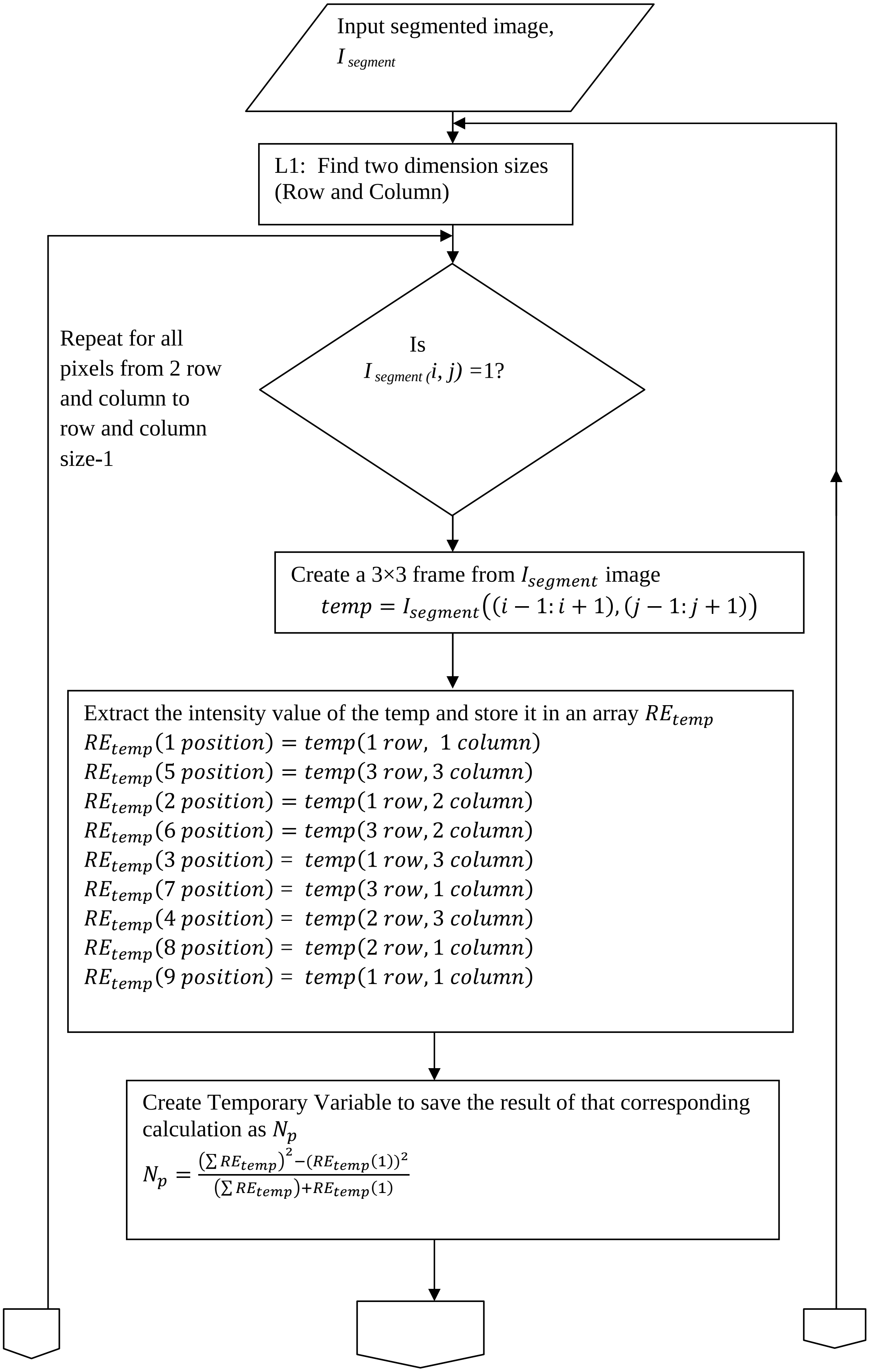
Figure 5: Example of Edge prediction based skeleton Formation

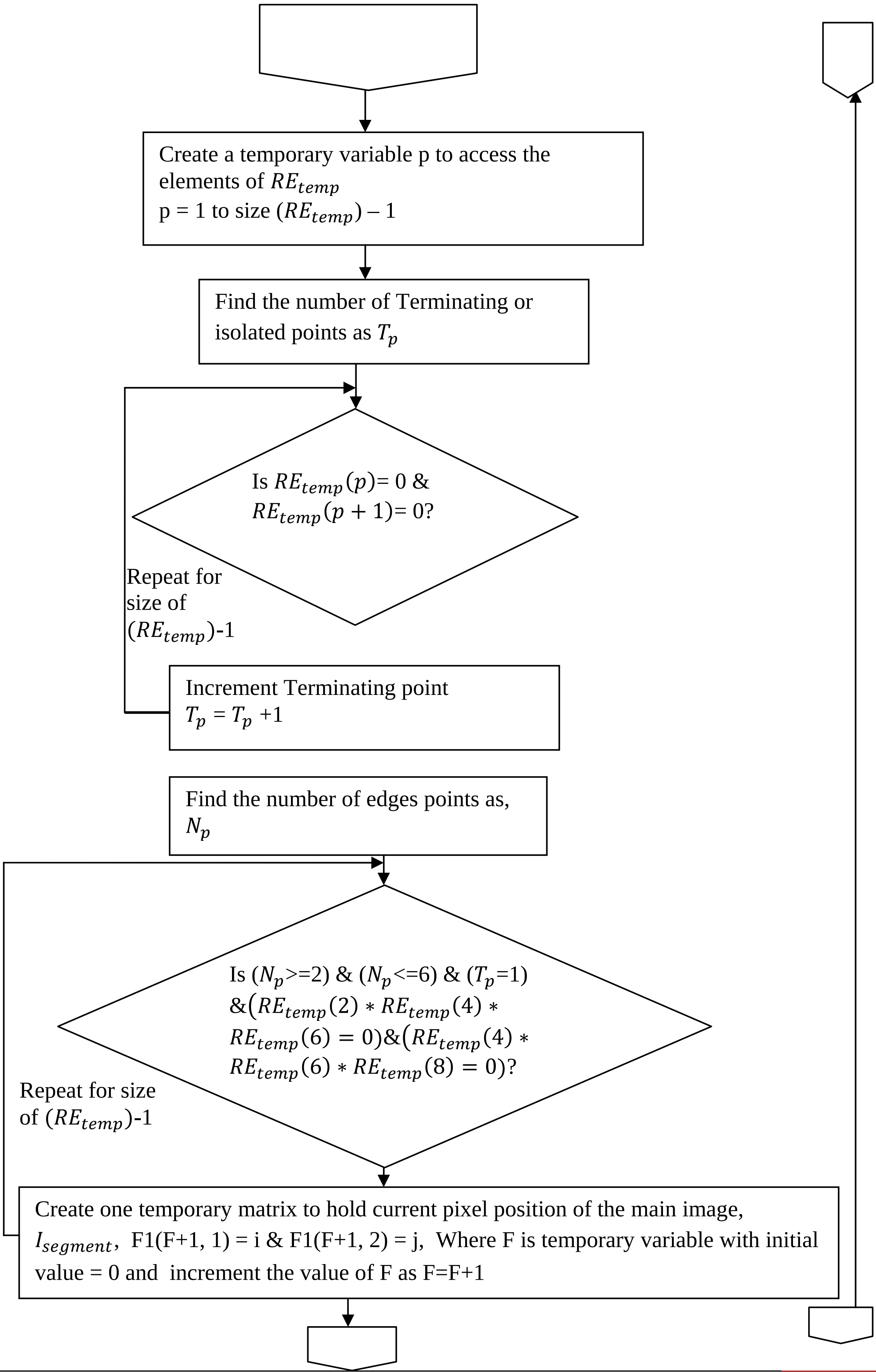
5. CONCLUSION

Biometrics innovation has ended up being a precise and proficient response to the security issue. Biometrics are a developing field of research as of late and has been dedicated to the distinguishing proof or authentication of people utilizing one or multiple inherent physical or behavioral characteristics. Thinning is a special process that consecutively wears away the foreground pixels and finally produces lines that are almost one-pixel. The first and foremost condition for thinning is input image should be a binary image and produces output as a binary image. Thinning is a final prior step to minutiae extraction in automatic fingerprint recognition system. Thinning is not achieved in a single step but it achieved through an iterative process. The connectivity of ridges and bifurcation can be reproduced from the thinning, means it preserves the basic structure of the image without affecting its original structure.

After extracting the minutiae from the thinned image a few post processing is carried to cast off any spurious minutiae and final features of fingerprint image is obtained. The strategies on this elegance are of types–crossing number based and morphology based. However, techniques based totally on thinning are sometime sensitive to noise and the skeleton shape does no longer conform to intuitive expectation. Non skeletonised feature extraction uses a binary image based totally technique. The principle problem within the minutiae extraction technique the use of thinning processes comes from the reality that minutiae within the skeleton image do not usually correspond to true minutiae inside the fingerprint image. In fact, quite a few spurious minutiae are determined because of undesired spikes, breaks, and holes. Consequently, put up processing is usually followed to keep away from spurious minutiae, which are based on each statistical and structural fact after characteristic or feature detection. The main idea in Edge Prediction based skeleton formation to use, eight-neighbourhood and there may be at least one background pixel or point, defined as boundary point.

This method considers segmented image, $I_{segment}$ as input for this process. The output from this method is skeletonised or thinned image, denoted as $I_{skeleton}$. Initially the size of the $I_{segment}$is calculated. This method is well known method for fingerprint thinning process and also effectively used for preprocessing skeleton in order to get better matching at the time of testing fingerprint phase.


Input segmented image, $I_{segment}$
L1: Find two dimension sizes (Row and Column)
Repeat for all pixels from 2 row and column to row and column size-1
Is $I_{segment}(i, j)$ =1?
Create a 3×3 frame from $I_{segment}$ image
$temp = I_{segment}((i-1:i+1),(j-1:j+1))$
Extract the intensity value of the temp and store it in an array RE_{temp}
$RE_{temp}(1\ position) = temp(1\ row,\ 1\ column)$
$RE_{temp}(5\ position) = temp(3\ row, 3\ column)$
$RE_{temp}(2\ position) = temp(1\ row, 2\ column)$
$RE_{temp}(6\ position) = temp(3\ row, 2\ column)$
$RE_{temp}(3\ position) = temp(1\ row, 3\ column)$
$RE_{temp}(7\ position) = temp(3\ row, 1\ column)$
$RE_{temp}(4\ position) = temp(2\ row, 3\ column)$
$RE_{temp}(8\ position) = temp(2\ row, 1\ column)$
$RE_{temp}(9\ position) = temp(1\ row, 1\ column)$
Create Temporary Variable to save the result of that corresponding calculation as N_p
$N_p = \frac{(\sum RE_{temp})^2 - (RE_{temp}(1))^2}{(\sum RE_{temp}) + RE_{temp}(1)}$


Create a temporary variable p to access the elements of RE_{temp}
p = 1 to size (RE_{temp}) – 1
Find the number of Terminating or isolated points as T_p
Is $RE_{temp}(p)$= 0 & $RE_{temp}(p+1)$= 0?
Repeat for size of (RE_{temp})-1
Increment Terminating point
$T_p = T_p + 1$
Find the number of edges points as, N_p
Is (N_p>=2) & (N_p<=6) & (T_p=1) &$(RE_{temp}(2) * RE_{temp}(4) * RE_{temp}(6) = 0)$&$(RE_{temp}(4) * RE_{temp}(6) * RE_{temp}(8) = 0)$?
Repeat for size of (RE_{temp})-1
Create one temporary matrix to hold current pixel position of the main image, $I_{segment}$, F1(F+1, 1) = i & F1(F+1, 2) = j, Where F is temporary variable with initial value = 0 and increment the value of F as F=F+1

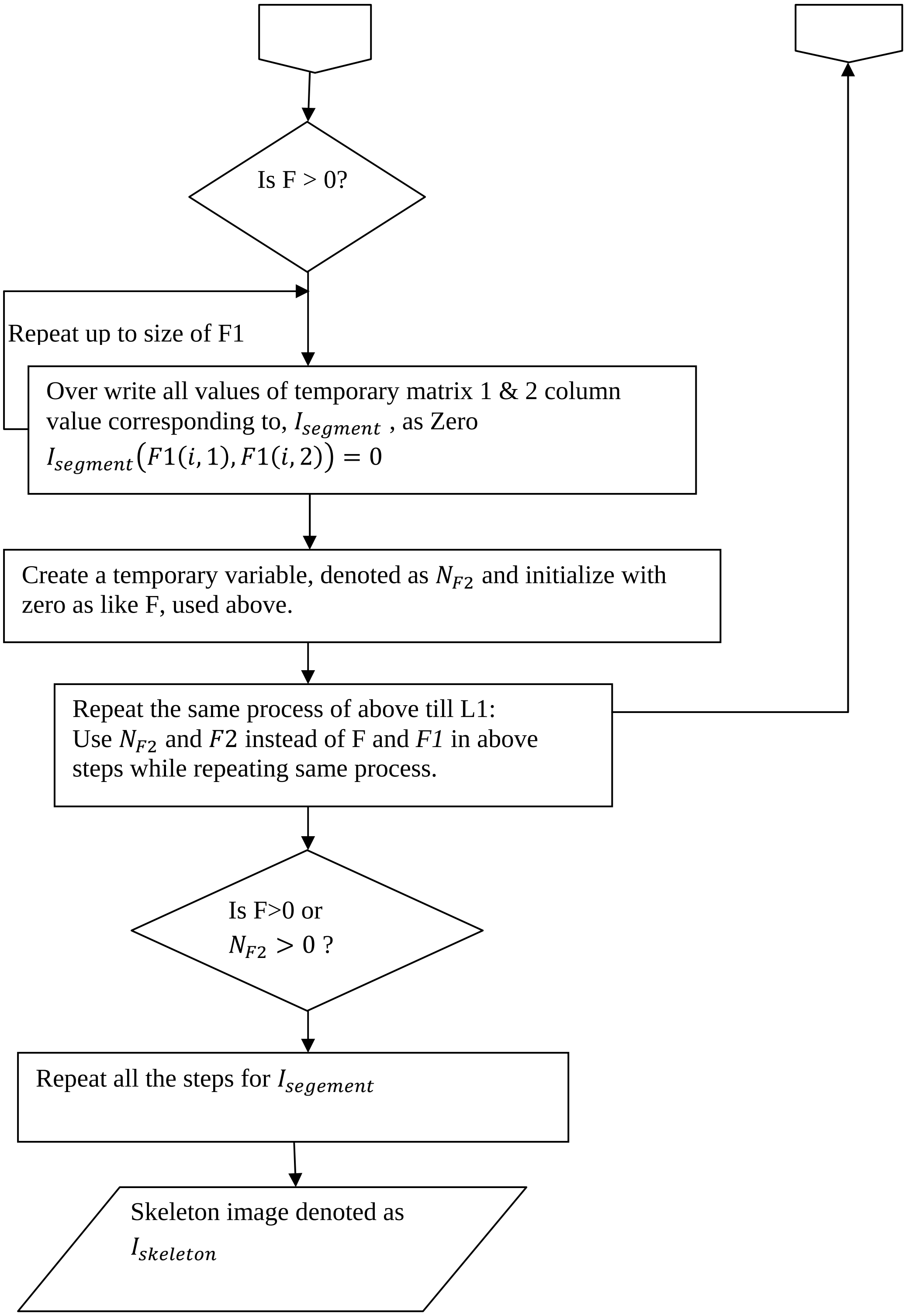


Figure 4: Flow chart for Edge Prediction based skeleton Formation Algorithm

REFERENCES

[1] Krishna Prasad, K. & Aithal, P. S. (2017). A Conceptual Study on Image Enhancement Techniques for Fingerprint Images. *International Journal of Applied Engineering and Management Letters (IJAEML),* 1(1), 63-72. DOI: http://dx.doi.org/10.5281/zenodo.831678

[2] Krishna Prasad, K. & Aithal, P. S. (2017). Literature Review on Fingerprint Level 1 and Level 2 Features Enhancement to Improve Quality of Image. *International Journal of Management, Technology, and Social Sciences (IJMTS)*, 2(2), 8-19. DOI: http://dx.doi.org/10.5281/zenodo.835608

[3] Krishna Prasad, K. & Aithal, P. S. (2017). Fingerprint Image Segmentation: A Review of State of the Art Techniques. *International Journal of Management, Technology, and Social Sciences (IJMTS),* 2(2), 28-39. DOI: http://dx.doi.org/10.5281/zenodo.848191

[4] Krishna Prasad, K. & Aithal, P. S. (2017). A Novel Method to Contrast Dominating Gray Levels during Image contrast Adjustment using Modified Histogram Equalization. *International Journal of Applied Engineering and Management Letters (IJAEML),* 1(2), 27-39. DOI: http://dx.doi.org/10.5281/zenodo.896653

[5] Krishna Prasad, K. & Aithal, P. S. (2017). Two Dimensional Clipping Based Segmentation Algorithm for Grayscale Fingerprint Images. *International Journal of Applied Engineering and Management Letters (IJAEML),* 1(2), 51-65. DOI: http://dx.doi.org/10.5281/zenodo.1037627.

[6] Hastings, E. (1992). A survey of thinning methodologies. *Pattern analysis and Machine Intelligence, IEEE Transactions*, 4(9), 869-885.

[7] Lam, H. K., Hou, Z., Yau, W. Y., Chen, T. P., & Li, J. (2008, December). A systematic topological method for fingerprint singular point detection. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on* (pp. 967-972). IEEE.

[8] Espinosa-Duro, V. (2002). Mathematical Morphology approaches for fingerprint Thinning. In *Security Technology, 2002. Proceedings. 36th Annual 2002 International Carnahan Conference on* (pp. 43-45). IEEE.

[9] Ahmed, M., & Ward, R. (2002). A rotation invariant rule-based thinning algorithm for character recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(12), 1672-1678.

[10] Patil, P. M., Suralkar, S. R., & Sheikh, F. B. (2005, November). Rotation invariant thinning algorithm to detect ridge bifurcations for fingerprint identification. In *Tools with Artificial Intelligence, 2005. ICTAI 05. 17th IEEE International Conference on* (pp. 8-pp). IEEE.

[11] You, X., Fang, B., Tang, V. Y. Y. & Huang, J. (2005). Multiscale approach for thinning ridges of fingerprint, Proc. Second Iberian Conference on Pattern Recognition and Image Analysis, Volume LNCS 3523, pp. 505–512.

[12] Jain, A. K., Hong, L., Pankanti, S., & Bolle, R. (1997). An identity-authentication system using fingerprints. *Proceedings of the IEEE,* 85(9), 1365-1388.

[13] Xiao, Q., & Raafat, H. (1991). Fingerprint image postprocessing: a combined statistical and structural approach. *Pattern Recognition*, 24(10), 985-992.

[14] Zhao, F., & Tang, X. (2007). Preprocessing and postprocessing for skeleton-based fingerprint minutiae extraction. *Pattern Recognition*, 40(4), 1270-1281.

[15] Akram, M. U., Tariq, A., Khan, S. A., & Nasir, S. (2008). Fingerprint image: pre-and post-processing. *International Journal of Biometrics*, 1(1), 63-80.

[16] Amengual, J. C., Juan, A., Pérez, J. C., Prat, F., Sáez, S., & Vilar, J. M. (1997). Real-time minutiae extraction in fingerprint images.

[17] Farina, A., Kovacs-Vajna, Z. M., & Leone, A. (1999). Fingerprint minutiae extraction from

skeletonized binary images. *Pattern recognition*, 32(5), 877-889.

[18] Gnanasivam, P., & Muttan, S. (2010). An efficient algorithm for fingerprint preprocessing and feature extraction. *Procedia Computer Science*, *2*, 133-142.

[19] Leung,W. F., Leung, S. H., Lau, W. H., Luk, A. (1991). Fingerprint Recognition using Neural Networks. Proc. IEEE Workshop on Neural Networks for Signal Processing, pp. 226-235.

[20] Humbe, V., Gornale, S. S., Manza, R., & Kale, K. V. (2007). Mathematical morphology approach for genuine fingerprint feature extraction. *International Journal of Computer Science and Security*, 1(2), 45-51.

[21] Bansal, R., Sehgal, P., & Bedi, P. (2010). Effective morphological extraction of true fingerprint minutiae based on the hit or miss transform. *International Journal of Biometrics and Bioinformatics (IJBB)*, 4(2), 71.

[22] Maio, D., & Maltoni, D. (1997). Direct gray-scale minutiae detection in fingerprints. *IEEE transactions on pattern analysis and machine intelligence*, *19*(1), 27-40.

[23] Maio, D., & Maltoni, D. (1998, August). Neural network based minutiae filtering in fingerprints. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on* (Vol. 2, pp. 1654-1658). IEEE.

[24] Jiang, X., Yau, W. Y., & Ser, W. (2001). Detecting the fingerprint minutiae by adaptive tracing the gray-level ridge. *Pattern recognition*, 34(5), 999-1013.

[25] Liu, J., Huang, Z., & Chan, K. L. (2000, September). Direct minutiae extraction from gray-level fingerprint image by relationship examination. In *Image Processing, 2000. Proceedings. 2000 International Conference on* (Vol. 2, pp. 427-430). IEEE.

[26] Nilsson, K., & Bigun, J. (2001). Using linear symmetry features as a pre-processing step for fingerprint images. In *Audio-and Video-Based Biometric Person Authentication* (pp. 247-252). Springer Berlin/Heidelberg.

[27] Wang, Y. G., & Carey, V. (2003). Working correlation structure misspecification, estimation and covariate design: implications for generalised estimating equations performance. *Biometrika*, 90(1), 29-41.

****