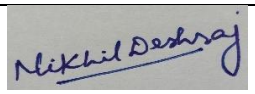
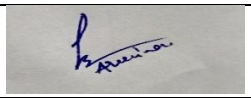
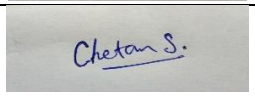
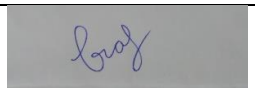
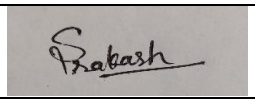
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

MANIT, BHOPAL

Minor Project Report (EC-326)

Group No.: 10

Project Title: FINGERPRINT MINUTIAE EXTRACTION.

S.NO.	SCHOLAR NO.	NAME	SIGNATURE
1	171114001	NIKHIL DESHRAJ	
2	171114006	ARUNIMA SAXENA	
3	171114007	CHETAN SONWANE	
4	171114028	PRAFUL LIKHITKAR	
5	171114149	SHREYA PRAKASH	

Name & Signature of faculty mentor ……..

Dr. LALITA GUPTA

1. INTRODUCTION

Looking towards the new digital era which has equally probable pros and cons, our electronically inter-connected society, reliable and user-friendly recognition and verification system is essential in many sectors of our life. BIOMETRICS is the science of uniquely recognising humans based on intrinsic physical traits such as Fingerprints, Iris scan, Face detection, Speech recognition, Signature verification, DNA testing etc. **Fingerprint recognition** is one of the most popular biometric techniques used in automatic personal identification and verification.

- A fingerprint is a distinct pattern of ridges and valleys on the finger surface of an individual. A **ridge** is defined to be a single curved segment whereas a **valley** is the area between two adjacent ridges. So, the dark areas of the fingerprint are called ridges and white area that exists between them is known as valleys.

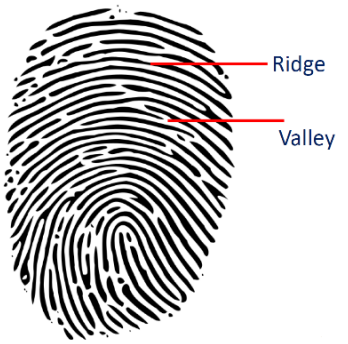
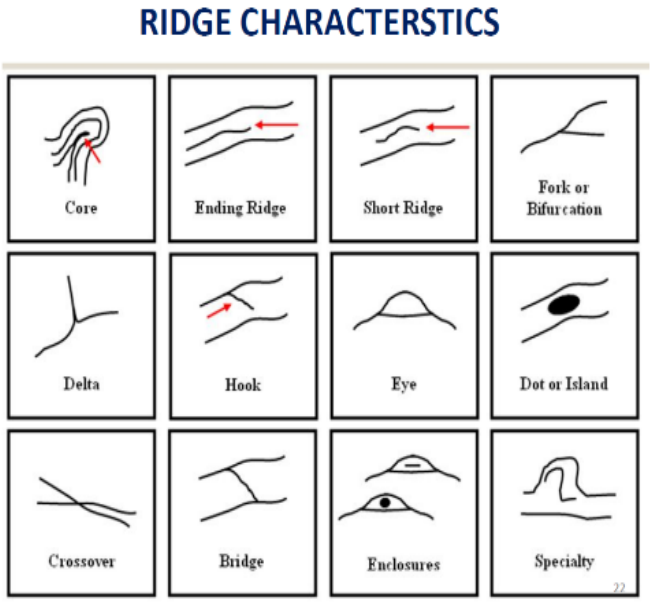


Fig: Fingerprint showing ridge and valley

- **Minutiae** can be defined as the points where the ridge lines end or fork. So, the minutiae points are the local ridge discontinuities and can be of many types.
  These types are –
  Ridge ending is the point where the ridge ends suddenly.
  Ridge bifurcation is the point where a single ridge branches out into two or more ridges
  Ridge dots are very small ridges.
  Islands are slightly longer than dots and occupy a middle space between two diverging ridges.
  Ponds or **Lakes** are the empty space between two diverging ridges.
  Spurs is a notch protruding from a ridge.
  Bridges are the small ridges that join two longer adjacent ridges.
  Crossovers are formed when two ridges cross each other.



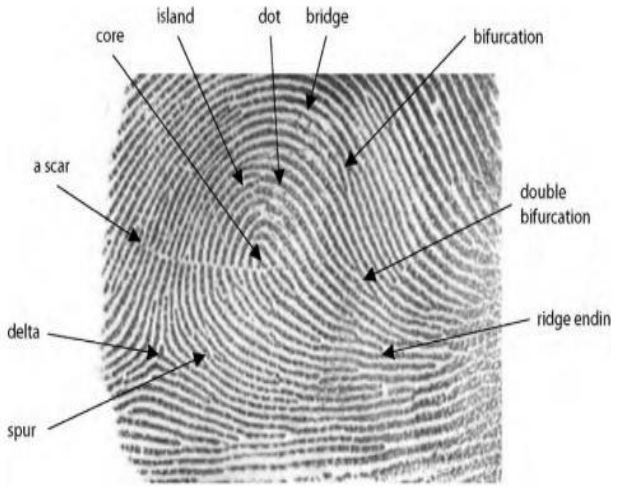


Fig: Different ridge characteristics shape and its location on fingerprint.

- Minutiae points are the major features of a fingerprint image and are used in the matching of fingerprints. These minutiae points are used to determine the uniqueness of a fingerprint image. A good quality fingerprint image can have 25 to 80 minutiae depending on the fingerprint scanner resolution and the placement of finger on the sensor.

- The uniqueness of a fingerprint is exclusively determined by the local ridge characteristics and their relationships. The ridges and valleys in a fingerprint alternate, flowing in a local constant direction. The two most prominent local ridge characteristics are: **1) Ridge ending** and, **2) Ridge bifurcation**. A ridge ending is defined as the point where a ridge ends abruptly. A ridge bifurcation is defined as the point where a ridge forks or diverges into branch ridges. Collectively, these features are called minutiae.

Due to its distinctiveness, compactness, and compatibility with features used by human fingerprint experts, minutiae-based representation has become the most widely adopted fingerprint representation scheme.

2. OBJECTIVE OF THE PROJECT

The aim of this project is to extract Minutiae from the fingerprint which can be used to match with the database in order to uniquely identify an individual or maintaining the database for matching process for any fingerprint recognition system.

3. BLOCK DIAGRAM

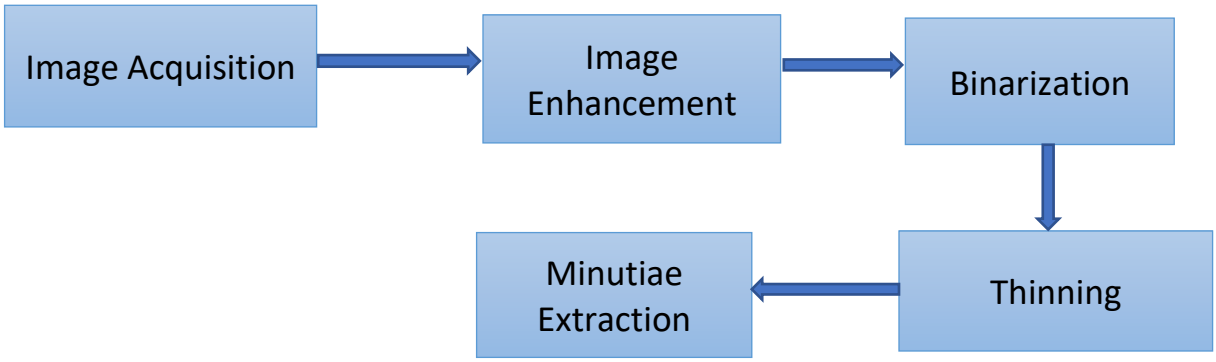


Image Acquisition:

- Image acquisition is the creation of a digitally encoded representation of the visual characteristics of an object, such as a physical scene or the interior structure of an object. It is done using Single sensor, Sensor strips or Sensor Array.

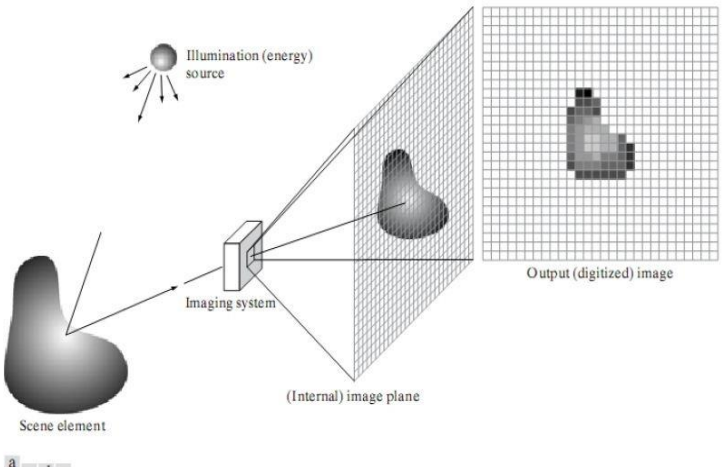


Fig: Digital imaging system.

Image Enhancement:

- Fingerprint minutiae extractors more or less require some sort of image pre-processing or enhancement to improve perceptibility. In doing so, they need to contend with two major types of problems: one is associated with image contrast such as insufficient dynamic range, and the other is associated with adverse physical factors such as scars, blurs, creases, sweat pores, and incipient ridges.

- Fingerprint image enhancement aims to minimize the undesired effects caused by such elements in order to extract a sufficient number of reliable features or minutiae.

- **Median filtering** is a nonlinear method used to remove noise from images. It is widely used as it is very effective at removing noise while preserving edges. It is particularly effective at removing **'Salt and Pepper'** type noise.

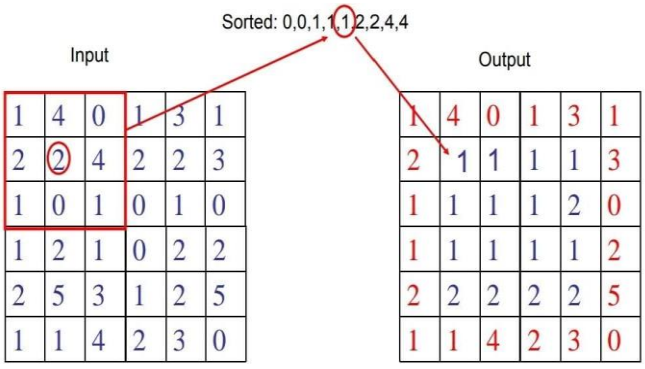


Fig: Image Enhancement using Median Filter

- The median filter works by moving through the image pixel by pixel, replacing each value with the median value of neighbouring pixels. The pattern of neighbours is called the **"window"**, which slides, pixel by pixel, over the entire image. The median is calculated by first sorting all the pixel values from the window into numerical order, and then replacing the pixel being considered with the middle (median) pixel value.

Binarization:

- Image Binarization is the process of taking a grayscale image and converting it to black-and-white, essentially reducing the information contained within the image from 256 shades of gray to 2: black and white, a binary image. This is sometimes known as **Image Thresholding**, although thresholding may produce images with more than 2 levels of grey.

- The aim of this technique is to compare the grey level of each pixel with threshold value to partitioning an image into foreground and background. If the pixel intensity is larger than threshold value then it is consider as 1 (white pixel) otherwise it is consider as 0 (black pixel).

- There are mainly two ways of Thresholding algorithm. One is **Simple Thresholding/Static Thresholding/Global Thresholding** and another is **Local Thresholding/Adaptive Thresholding/ Dynamic Thresholding**. We have gone through both ways.

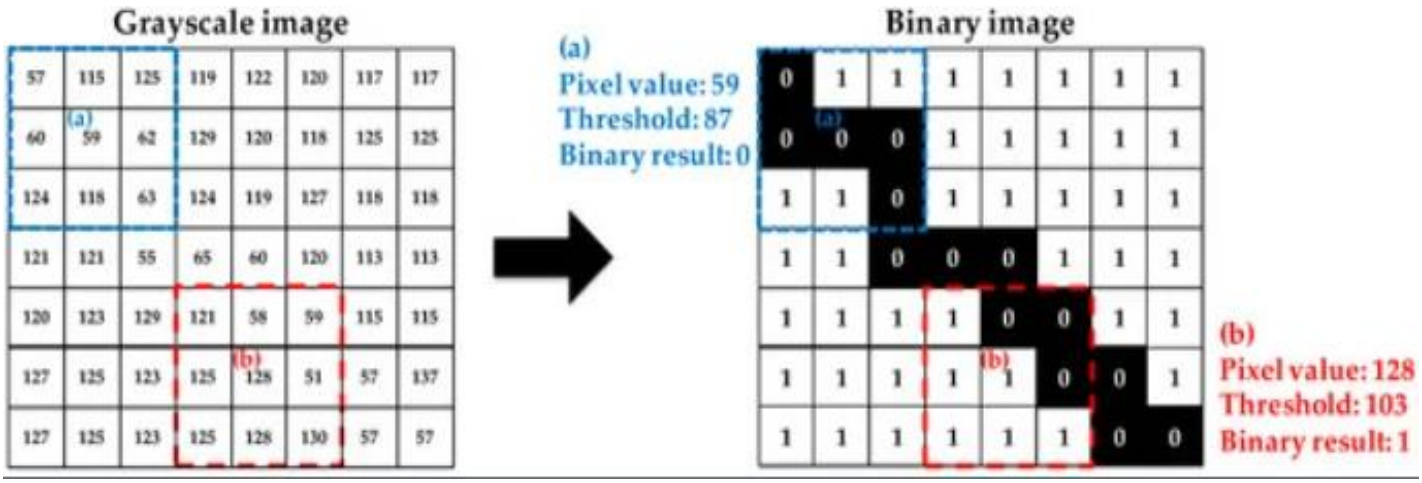


Fig: Converting grayscale image to binary image.

- The global threshold applicable when the intensity distribution of objects and background pixels are sufficiently distinct. In the global threshold, a single threshold value is used in the whole image. When the pixel values of the components and that of background are fairly consistent in their respective values over the entire image, global thresholding could be used.

- There is a difficulty that binarization cannot be done by using single intensity threshold because all fingerprint images do not have constant image contrast in the process of making binary images, and even the contrast ratio of the same person's fingerprints varies every time the device is pressed on. Therefore, the dynamic thresholding method is applied depending on image distribution pixel values and through it, the whole image is binarized into the ridge part and non-ridge part.

- Otsu method is a clustering based image thresholding. It works when the histogram is bimodal. The method basically tries to minimize the within class variance and at the same time it maximizes the between class variance.
  Total variance=Within class variance +Between Class Variance.

Thinning:

- Thinning is a process of extracting a skeleton from an object in a digital image. A skeleton of an image can be thought of as a one-pixel thick line through the middle of an object which preserves the topology of that object.

- The thinning operation is related to the hit-and-miss transform and can be expressed quite simply in terms of it. The thinning of an image *I* by a structuring element *J* is:

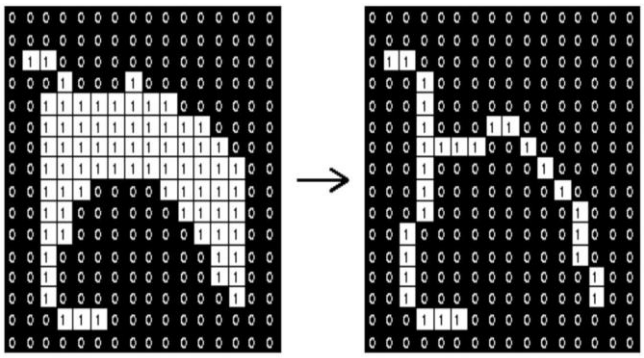


Fig: Binary image → Thinned image

$$thin(I,J) = I - hit\_or\_miss(I,J)$$

where the subtraction is a *logical subtraction* defined by- $X-Y = X \cap NOT\ Y$

- Thinning is a fundamental preprocessing step in many image processing and pattern recognition algorithms.Thinned images (skeletons) are easier to process and they reduce processing time for the subsequent operations. The behavior of a thinning algorithm is determined by its structuring element.

- At each iteration, the image is first thinned by the left hand structuring element, and then by the right hand one, and then with the remaining six 90° rotations of the two elements. The process is repeated in cyclic fashion until none of the thinnings produces any further change.
- **Structuring elements** are policies which define the situations at which foreground pixels will be set to background and hence deleted. Thinning is used in but not limited to applications that process handwritten and printed characters, fingerprints and palm prints, chromosomes and biological cell structures, and circuit diagrams.

0	0	0
	1	
1	1	1

	0	0
1	1	0
	1	

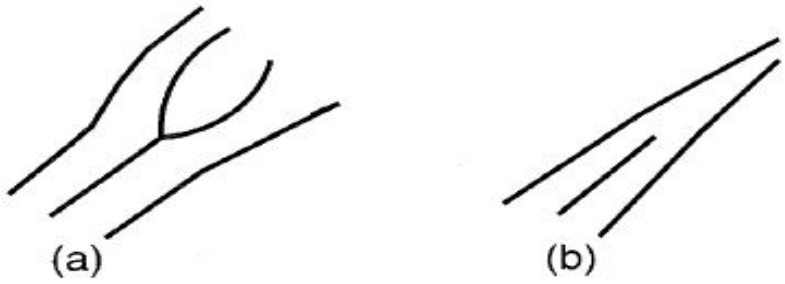
Fig: This structuring element is used to get the morphological thinning of above binary image.

- Generally, fingerprint recognition systems work by matching minutiae extracted from probe data, to reference minutiae and it consists of the following stages: fingerprint acquisition, image pre-processing, minutiae extraction and matching .

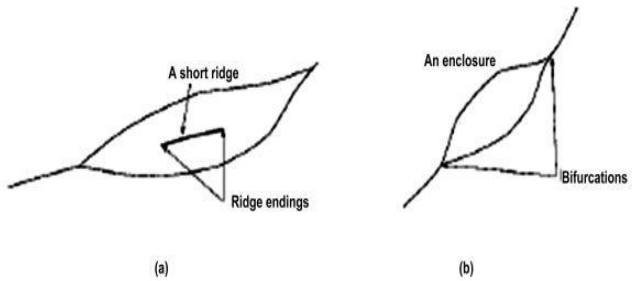
- Fingerprint thinning is an important image enhancement processing step in an **Automatic Fingerprint Identification System (AFIS)**. It plays an equally significant role with fingerprint classification and enhancement in practical AFIS. It can significantly improve the recognition performance of an AFIS.

Minutiae Extraction:

- Most of the fingerprint extraction and matching techniques restrict the set of features to two types of minutiae: ridge endings and ridge bifurcations. A good quality fingerprint typically contains about 40–100 minutiae.
- In a latent or partial fingerprint, the number of minutiae is much less (approximately 20 to 30). More complex fingerprint features can be expressed as a combination of these two basic features. For example, an enclosure can be considered a collection of two bifurcations and a short ridge can be considered a pair of ridge endings.
  (a) Ridge Bifurcation (b) Ridge Ending



Complex features as a combination of simple features: (a) short ridge; (b) enclosure

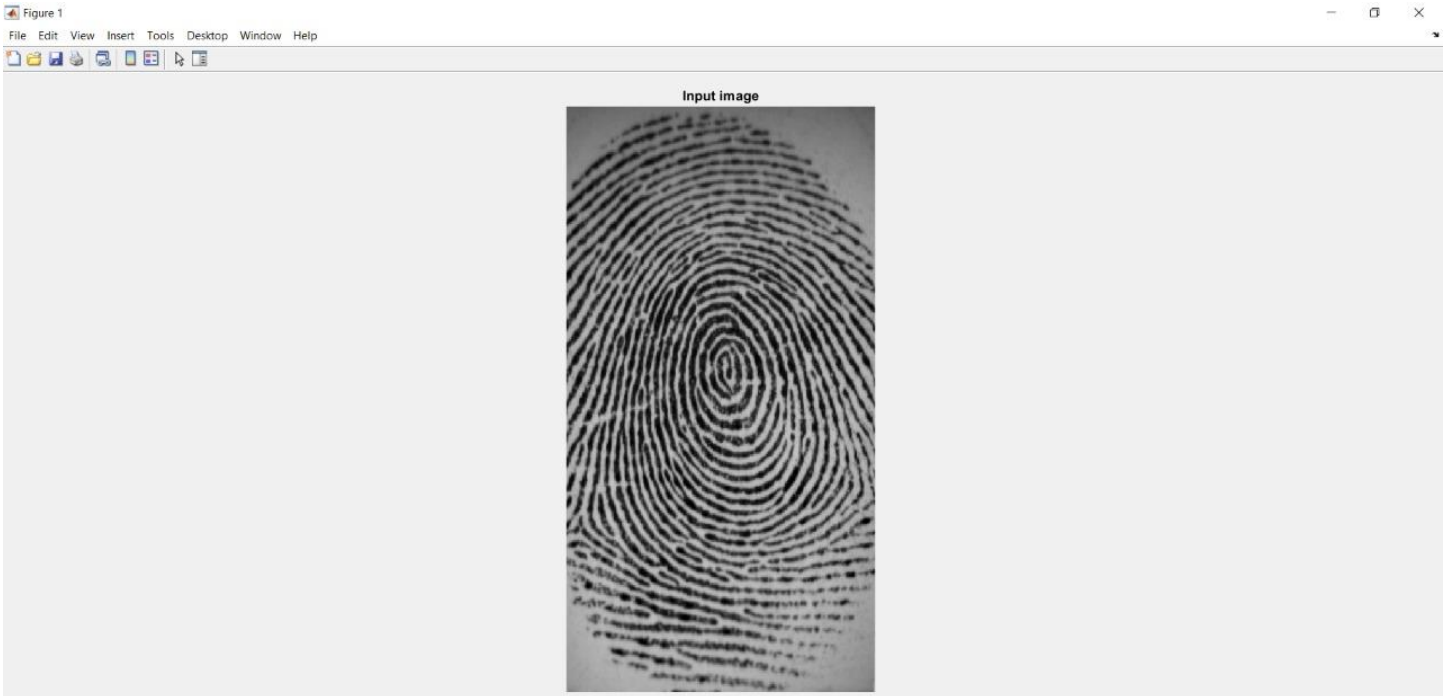


4. RESULT

Here is the result below on implementation of above code.

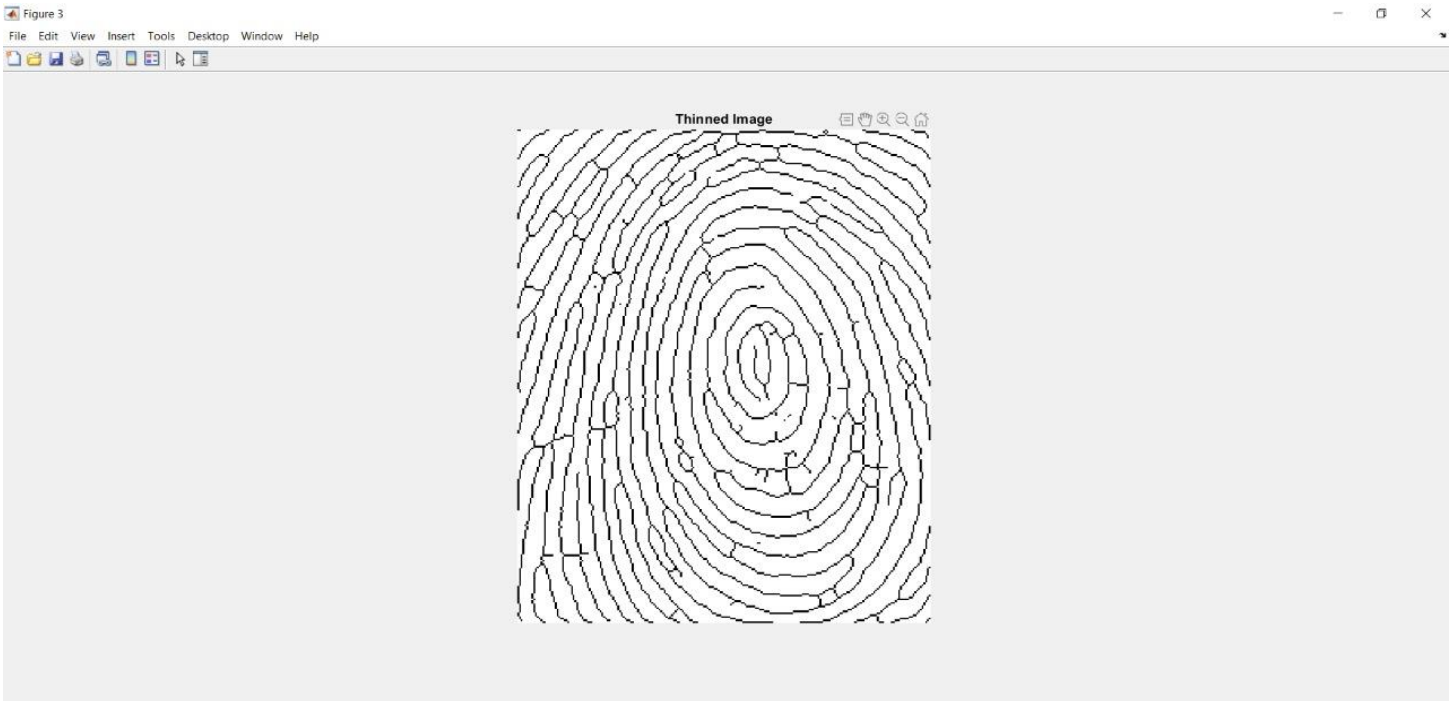
FOR IMAGE 1:

a) **Input Image 1-** This is the first input image whose minutiae points need to be extracted.

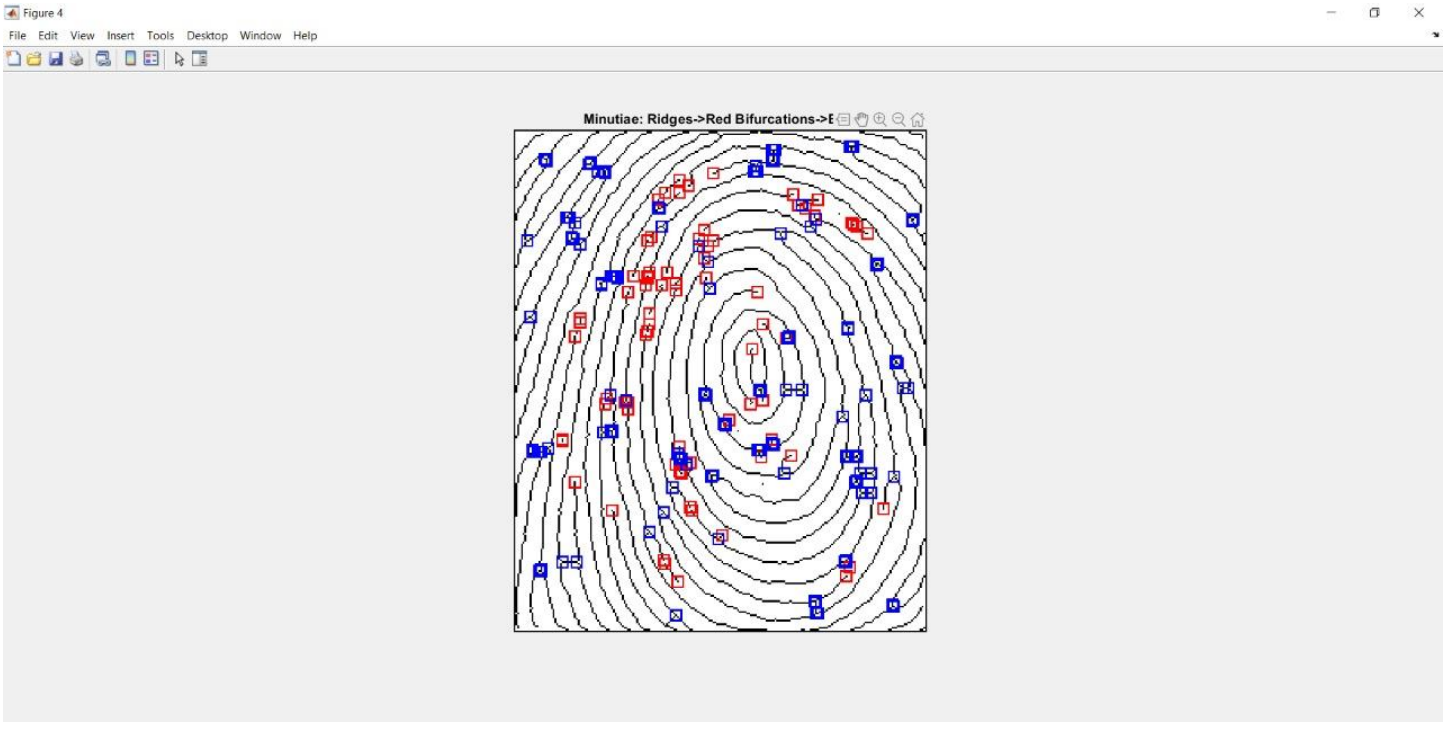


b) **Binary Image-** This is the result of applying binarization on input image 1.

c) Thinned Image- After thinning operation on binary image.

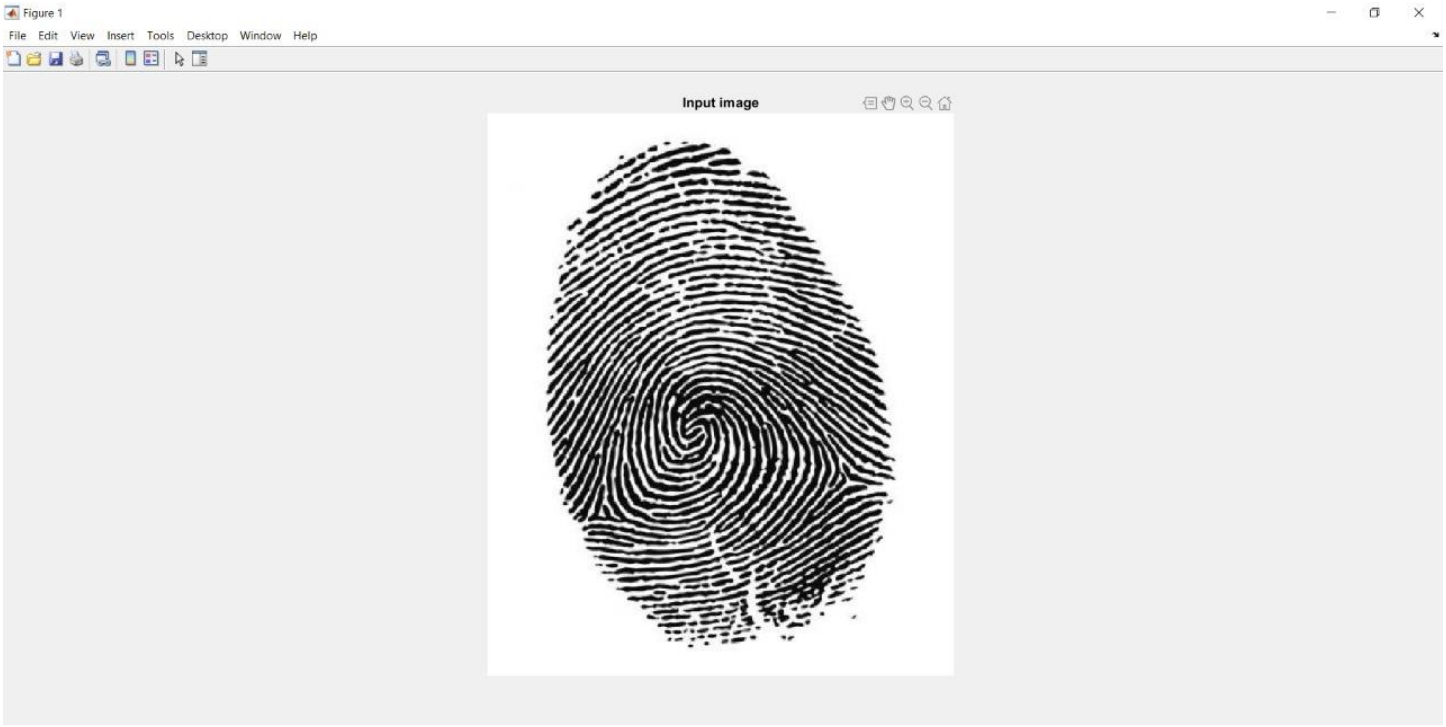


d) Minutiae Points- Extracted minutiae points on the thinned image.

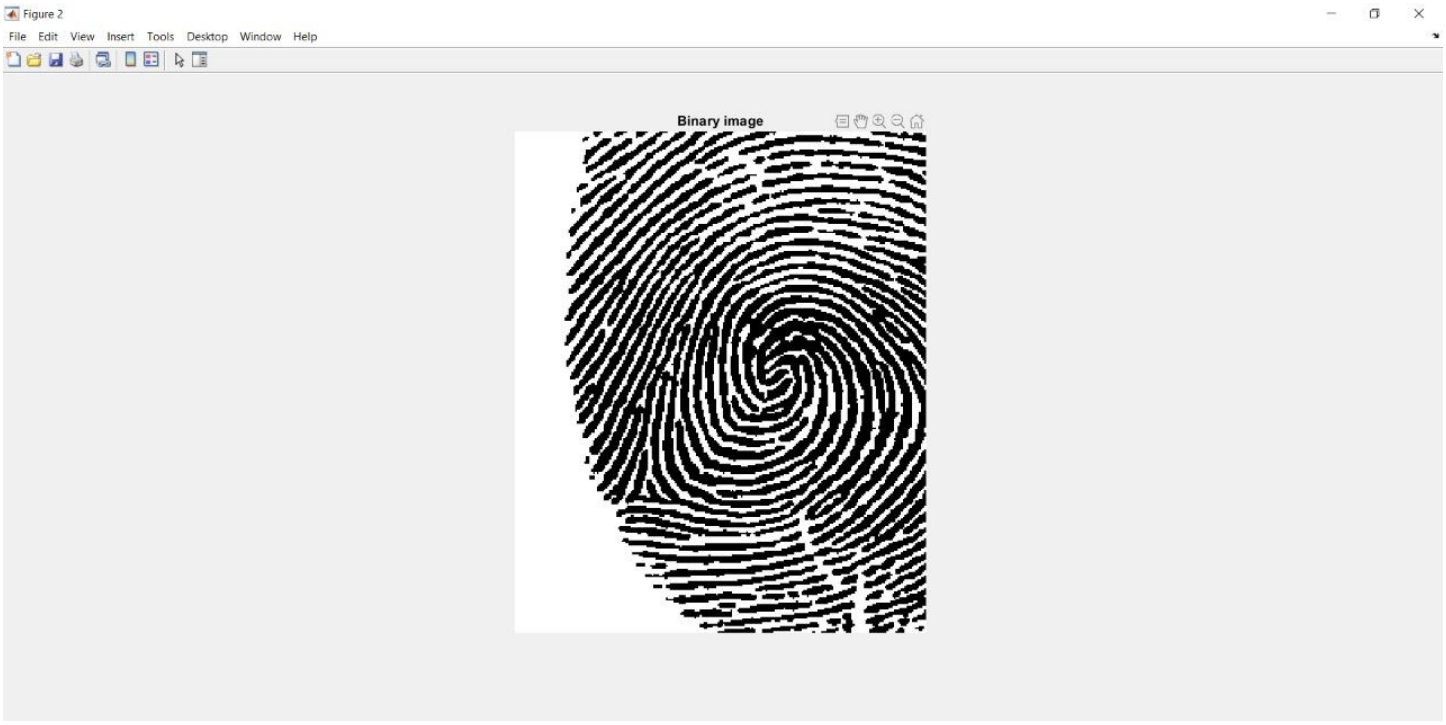


FOR IMAGE 2:

1) **Input image 2-** This is the second input image whose minutiae points need to be extracted.

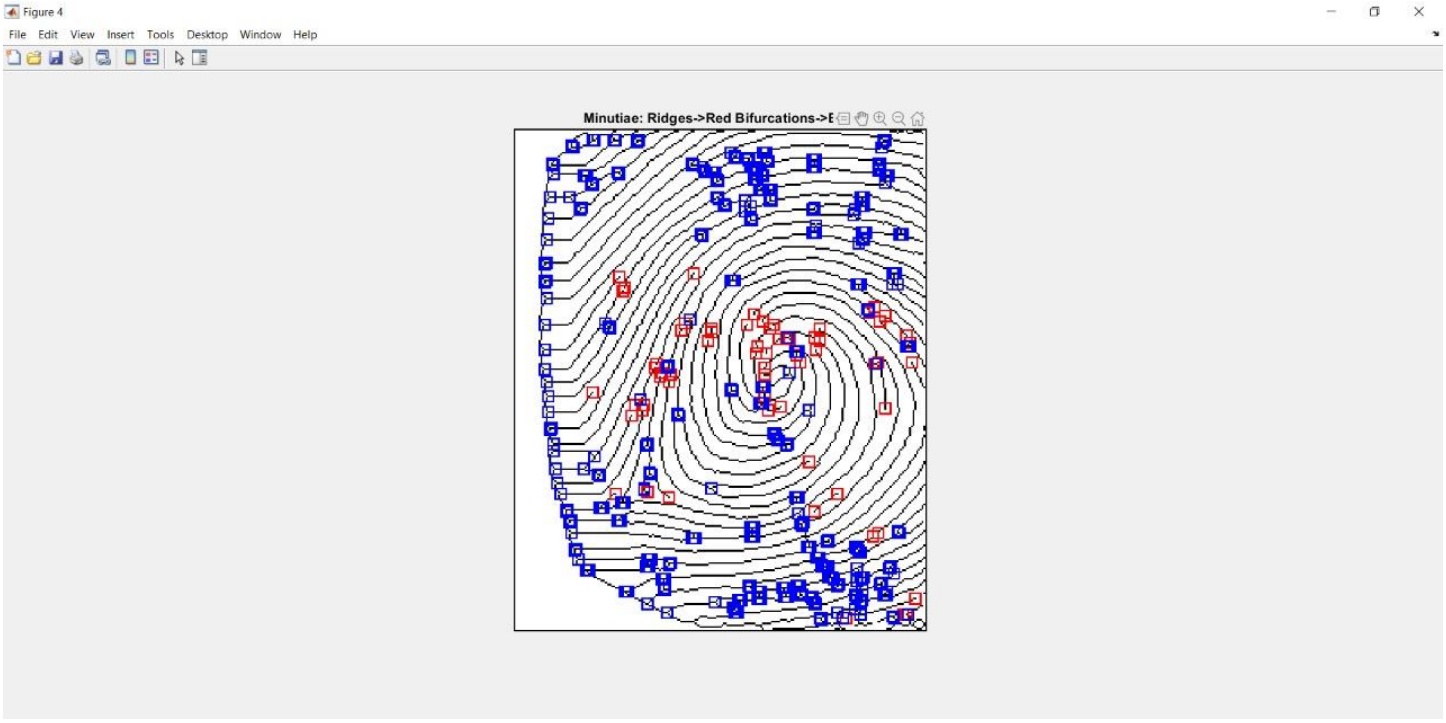


2) **Binary Image-** This is the result of applying binarization on input image 2.



3) **Thinned Image**- After thinning operation on binary image.

4) **Minutiae Points**- Extracted minutiae points on the thinned image.



- Compared to other fingerprint features, the minutiae point features having corresponding orientation maps are distinct enough to distinguish between fingerprints robustly. Fingerprint representation using minutiae feature reduces the complex issue of fingerprint recognition to an issue of point pattern matching.

- Since the original image cannot be reconstructed using only the minutiae information, the minutiae-based fingerprint identification systems can also assist privacy issues and the minutiae are actually sufficient enough to prove finger individuality. In terms of contrast, image resolution and global distortion the minutiae are more stable and robust in relation to other fingerprint matching schemes.

5. FUTURE SCOPE AND ADVANCEMENTS

- Overall, a set of reliable techniques have implemented for fingerprint image enhancement and minutiae extraction as an efficient minutiae extraction method leads to provide a better platform for matching techniques. These techniques can then be used to facilitate further study of the statistics of fingerprints.

- The future scope of the work is to match the extracted minutiae from the fingerprint by removal of false minutiae, which may get extracted due to the low quality of fingerprint, in order to uniquely identify an individual. Also, to improve the quality of the image either by improving the hardware to capture the image or by improving the image enhancement techniques so that the input image to the thinning stage could be made better which can improve the future stages and the final outcome.

- The fingerprint is the most important factor for authentication and authorization. The use of UID (User IDentification) to test fingerprints at different places with different applications can make it feasible to examine the originality of the person presented. If the Govt. election can be conducted using the UID card, then fake entries can be avoided. If the ATM and cards can be connected with the UID card system then, only authenticated people would be able to transact money.

6. REFERENCES

- Handbook of Fingerprint Recognition by Davide Maltoni, Dario Maio, Anil K. Jain & Salil Prabhakar.
- Fingerprint Recognition, Paper by WUZHILI (Department of Computer Science & Engineering, Hong Kong Baptist University) 2002.
- Fingerprint Classification and Matching by Anil Jain (Department of Computer Science & Engineering, Michigan State University) & Sharath Pankanti (Exploratory Computer Vision Group IBM T. J. Watson Research Centre) 2000.
- Fingerprint database - FVC2002 (Fingerprint Verification Competition 2002).
- Wikipedia link - http://en.wikipedia.org/wiki/Fingerprint_recognition

7. CODE USED

Programming language used is **MATLAB**.

Here is the code below:-

```
%Program for Fingerprint Minutiae Extraction

%Program Description
%This program extracts the ridges and bifurcation from a fingerprint image

%Read Input Image
img=imread('C:\Users\hp\Desktop\input_1.tif');
figure(1);
imshow(img);
title('Input image');

%Binarization
binImg=imbinarize(img);
binImg = binImg120:400,20:250); %Considering a portion of image
figure(2);
imshow(binImg);
title('Binary image');

%Thinning
thinImg=~bwmorph(binImg,'thin',Inf);
figure(3);
imshow(thinImg);
title('Thinned Image');

%Minutiae extraction
s=size(thinImg);
N=3; %window size
n=(N-1)/2;
r=s(1)+2*n;
c=s(2)+2*n;
double temp(r,c);
temp=zeros(r,c);
bifurcation=zeros(r,c);
ridge=zeros(r,c);
temp((n+1):(end-n),(n+1):(end-n))=thin_image(:,:); %n pixel thick boundary
outImg=zeros(r,c,3);

%For Display
outImg(:,:,1) = temp .* 255;
outImg(:,:,2) = temp .* 255;
outImg(:,:,3) = temp .* 255;
for x=(n+1+5):(s(1)+n-5)
   for y=(n+1+5):(s(2)+n-5)
```

```
      a=1;
      for k=x-n:x+n
         b=1;
         for l=y-n:y+n
            mat(a,b)=temp(k,l);
            b=b+1;
         end
         a=a+1;
      end;
       if(mat(2,2)==0)
         ridge(x,y)=sum(sum(~mat));
         bifurcation(x,y)=sum(sum(~mat));
       end
   end
end

% Finding ridge endings
[rx ry]=find(ridge==2);
len=length(rx);

%For Display
for i=1:len
   outImg((rx(i)-3):(rx(i)+3),(ry(i)-3),2:3)=0;
   outImg((rx(i)-3):(rx(i)+3),(ry(i)+3),2:3)=0;
   outImg((rx(i)-3),(ry(i)-3):(ry(i)+3),2:3)=0;
   outImg((rx(i)+3),(ry(i)-3):(ry(i)+3),2:3)=0;

   outImg((rx(i)-3):(rx(i)+3),(ry(i)-3),1)=255;
   outImg((rx(i)-3):(rx(i)+3),(ry(i)+3),1)=255;
   outImg((rx(i)-3),(ry(i)-3):(ry(i)+3),1)=255;
   outImg((rx(i)+3),(ry(i)-3):(ry(i)+3),1)=255;
end

%Finding  bifurcation
[bx by]=find(bifurcation==4);
len=length(bx);

%For Display
for i=1:len
   outImg((bx(i)-3):(bx(i)+3),(by(i)-3),1:2)=0;
   outImg((bx(i)-3):(bx(i)+3),(by(i)+3),1:2)=0;
   outImg((bx(i)-3),(by(i)-3):(by(i)+3),1:2)=0;
   outImg((bx(i)+3),(by(i)-3):(by(i)+3),1:2)=0;

   outImg((bx(i)-3):(bx(i)+3),(by(i)-3),3)=255;
   outImg((bx(i)-3):(bx(i)+3),(by(i)+3),3)=255;
   outImg((bx(i)-3),(by(i)-3):(by(i)+3),3)=255;
   outImg((bx(i)+3),(by(i)-3):(by(i)+3),3)=255;
end
```

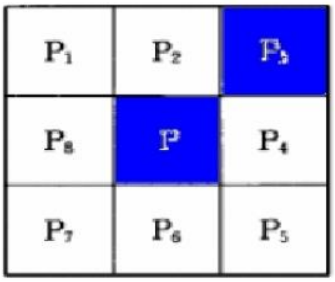


Fig: Sample for 3×3 matrix pattern of Ridge ending.

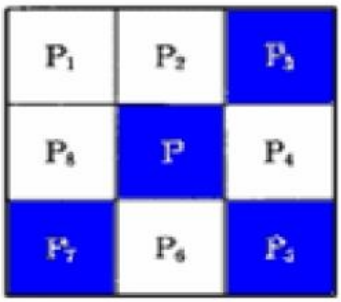


Fig: Sample for 3×3 matrix pattern of Bifurcation.

```
%Display output
figure(4);
imshow(outImg);
title('Minutiae: Ridges->Red Bifurcation-> Blue');
```

****