REPORT

ON

UNSUPERVISED LEARNING APPROCH FOR FINGERPRINT DETECTION

BY

Name of the student	Enrollment No.
P. Mohana uma sushmanth	1800248C203

Prepared in the partial fulfillment of the

Practice School II (THESIS) Course

AT

BML MUNJAL UNIVERSITY

A Practice School II Station of

BML MUNJAL UNIVERSITY

(July, 2020)

REPORT

ON

UNSUPERVISED LEARNING APPROCH FOR FINGERPRINT DETECTION

BY

Name of the student	Enrollment No.
P. Mohana uma sushmanth	1800248C203

Prepared in the partial fulfillment of the
Practice School II (THESIS) Course

AT

BML MUNJAL UNIVERSITY

A Practice School II Station of

BML MUNJAL UNIVERSITY

(July, 2020)

Table of contents:

BML MUNJAL UNIVERSITY
FROM HERE TO THE WORLD

Certificate of authenticity

CERTIFICATE

This is to certify that Practice School Project of ___________Penumarthi mohana uma sushmanth___________ titled ________________Unsupervised learning approach for fingerprint detection______ is an original work and that this work has not been submitted anywhere in any form. Indebtedness to other works/publications has been duly acknowledged at relevant places. The project work was carried during ___25/05/2020______ to _______24/07/2020________ under the guidance of ______Dr. Kiran khatter mam________

Signature of Supervisor:

Name:

Designation:

ACKNOWLEDGEMENT

On the accommodation of my Thesis report, I want to expand my sincere thanks and my earnest gratitude to my mentor Dr. Kiran Khatter mam, Department of CSE, for providing me chance to do research under her guidance and giving significant direction all through this exploration. Her vision, genuineness and inspiration have inspired me a lot. She has demonstrated the techniques to make the exploration and to present this work as clearly as possible. It was an implausible advantage and regard to work under her direction. I am amazingly thankful for what she has offered me. Her direction showed me the way in all the time of difficulties and composing of this theory. I was unable to have envisioned having a superior guide for this project.

Penumarthi mohana uma sushmanth

Reg No: 1800248C203

Batch 2018-CSE

ABSTRACT

Fingerprints are one of the best sources for identification of individuals. They are safe to use, different for everyone and not change in entire lifetime. The main objective of this research project is getting clear idea about how fingerprints are detected. There are different techniques to detect fingerprints. For example, Image based, correlation based, minutiae based etc. In this project minutiae-based detection is used. Fingerprints are high in details known as minutiae, which are utilized as recognizable proofs for security purposes. In this project, the complete information of fingerprint detection is explained clearly.

In this process, mainly there are three step that should be applied on fingerprint image. They are preprocessing, feature extraction, post processing. In these steps, feature extraction step is very important which is responsible for detection while matching fingerprints. Obtaining a clear and quality fingerprint image is not so easy. So, the images ought to be pre-processed before applying minutiae extraction step. Every step in this process is very important and interlinked. The image preprocessing, feature extraction and postprocessing steps are explained.

Objective:

Our main goal of this research project is getting clear idea about how fingerprints are detected. Fingerprint detection is very useful now-a-days for biometric purpose, for security purpose and people leave fingerprints almost everywhere and that's why forensics uses fingerprint databases for investigation purpose. So, in this project we are also going to try a different method's using python and OpenCV libraries and MATLAB software for feature extraction.

Problem statement:

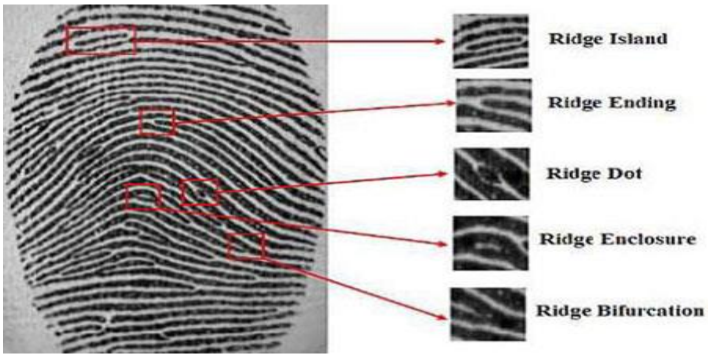
Fingerprint is very useful for biometric purpose, security purpose and mainly for investigation purpose. Forensics uses fingerprints to identify the people who involved in a crime. They used to maintain a large dataset of fingerprints. Actual problem is to get the good enhanced image and identify the fingerprints more accurately.

Our project is to design a system to perform the detection of human fingerprints using Python, OpenCV libraries, MATLAB software for feature extraction.

Introduction:

Fingerprints are the patterns formed on the epidermis of the fingertip. They structured from pressure on a baby's tiny, developing fingers in the womb. No two people have been found to have similar fingerprints; they are completely unique. There is just one out of 64 billion chance that your prints will facilitate accurately with another person's. Fingerprints are considerably best one of a kind than DNA Identical twins can have a comparative DNA, yet they cannot have comparative fingerprints. The fingerprints are of three types: arch, loop, and whorl. The unique finger impression is made out of valleys and ridges.

Identification through fingerprints is one of the well-known biometric techniques. Sir Francis Galton invented fingerprints. He is a British anthropologist, started observing on fingerprints as methods for detection in the 1880's. He is known as father of fingerprints. It is a strategy for confirming match between two unique fingerprints. It is for the most part utilized in the ID of an individual and in criminal investigations. It is the combination of ridges and valleys. Features in the ridge design are utilized for recognizable proof. These features are known as minutiae points. They are different types of minutiae points. In this project, we mainly consider only two types. They are ridge ending and ridge bifurcations.



There are three ways of identifying fingerprints. They are Image-based, Correlation-based, and Minutiae-based.

Image based Approach:

In Image based approach, directly the scanned image is taken as a final template. This process is extremely fast. This process is nothing but comparing the images and subtracting input template from the template which stored in the database. It does not require good quality image. But it needs exact arrangement of fingerprint image and isn't work for changes in direction, position, and scale.

Correlation based Approach:

The correlation-based approach can deal with low quality images where there is no need of extracting minutiae. First chooses suitable template in primary image. Locates the template in secondary fingerprint. And Both the fingerprint templates are compared. Good templates are uniquely positioned in the secondary fingerprint at correct position.

Minutiae based Approach:

This methodology is the last and best methodology which gives exact outcomes. Here, the features present on ridges are known as minutiae which will get separated and saved in templlate for identification. It is innvariant to rotation, translation, and scaling. But It shows wrong when we are applying this method on low quality images. This method needs good quality images.

Minutiae based approach is applied in this project. Generally, for extracting feature we require a good quality image. Getting good-quality image not so easy always. So, preprocessing step is performed before feature extraction. In this project I have focused mainly on preprocessing step and feature extraction step. Because we have many different methods and fixed algorithms for matching process. But to match properly, the input template should good. In this project, after performing first two steps we will get the fingerprint such that it suitable to proceed to next step.

The image preprocessing step consist of different step. They are image acquisition, image segmentation, normalization, binarization and some morphological operations like dilation, erosion, opening etc. The techniques involve in enhancement step are used to improve the clarity of ridges by reducing the noise. The output of pre-processing step will be the enhanced fingerprint image. And next feature extraction step will be applied.

Project Methodology:

This project comprises of three steps. They are pre-processing, features extraction and post processing. Internally, each step contains different process. Each process is explained clearly in this project. The first step is preprocessing.

Pre-processing:

Image preprocessing is a basic step which performs the following operations on the image to make the image better. A good fingerprint impression merits better performance of the fingerprint algorithm. Means it decreases the complexity, increase the contrast, and improve the quality of image. It is helpful to give better result in second stage. In this project, we have done the pre-processing step by using simple steps which also helps to reduce complexity. The steps involved in this stage are:

1) **Image-acquisition:**
   Image-acquisition is the first stage in this process. There are two modes of acquisition. Depending up on the acquisition modes, a scanning is separated into offline and online scan. An offline print is acquired by taking ink on the fingertip and making an inked print of the fingertip on paper. On other hand, an online print is obtained by detecting the tip of the finger utilizing sensors which can digitize the fingerprint on contact. Online scanning can finish by using sensors. There are three sorts of sensors which filter the fingerprint.They are ultrasonic sensors, optical sensors, and capacitance sensors.

Image show above is 106_2 taken from fingerprint dataset FVC2000

2) **Converting RGB image into grayscale:**
After capturing the image, we should convert the image to grayscale image to reduce the complexity. It converts from 3D pixel value (R, G, B) to 1D pixel value. Many tasks do not fare better with 3D pixels (example: edge detection). After conversion, the image pixels only lie between 0 to 255.

3) **Image segmentation:**
Image segmentation is performed to enhance the foreground image. Generally, this process is used to identify the object on an image like curves, lines etc. There are two types of regions in an image. They are fore-ground region and back-ground region. The fore-ground areas show the lines (ridges) and valleys while rest in the image is allocated to back-ground region. These both regions differ by their variance values. The fore-ground areas have high variance values where the back-ground areas have low values. Image segmentation is a proccess of separating the fore-ground region from back-ground image. First the given image is separated into square blocks. For each square, the grayscale variance is determined. If the value is not greater than the global threshold it is allotted to back-ground. Otherwise, it will be allotted to fore-ground. The variance of each pixel is calculated by following formulae:

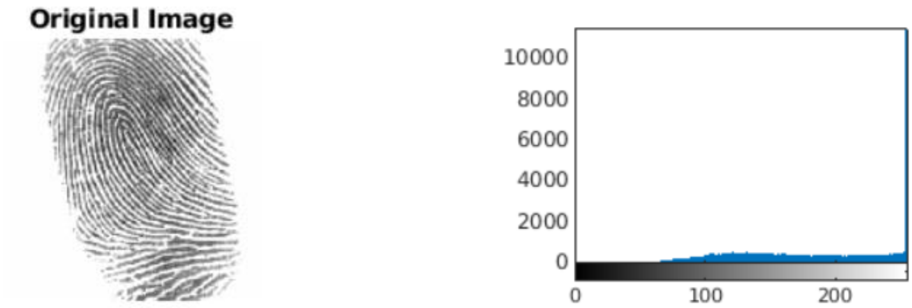
$$\mathrm{V(k)} = \frac{1}{n^2}\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}(I(i,j) - M(k))^2$$

Here V(k) is variance for square block of size n×n. I (i, j) is the grayscale value of pixel (i, j), and M(k) is mean gray value. The threshold limit should be given appropriately. On the off chance that the limit esteem is excessively high, foreground regions might be inaccurately allotted as background regions. Additionally, if the limit esteem is excessively little, background regions might be allotted as foreground region. In average, the variance value has taken around 126 which give best outcomes in terms of separating fore-ground and back-ground images.

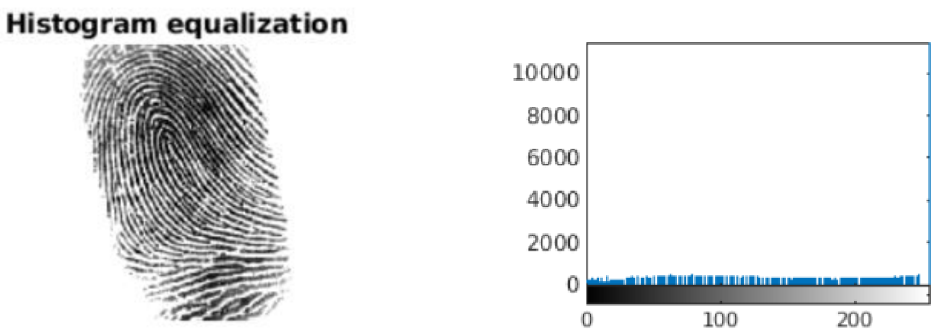
4) Image enhancement:

It is the process of improving quality of digital images so that the outcomes are more worthy for display or for operations performed further. For example, we can make the image sharpen or brighten, and making it easier to identify features by removing noise. There are many types of techniques in image enhancement. Some of them are using filters with morphological operations, removing noise by using wiener filter, LCA (linear contrast adjustment), normalization etc. In this project, we are going to discuss about histogram equalization.

Histogram equalization is a normalization technique. It is a process of shifting their intensity values to increase or decrease the contrast of images. Generally, in fingerprint image each pixel has different gray values. Some have high contrast, and some have low. We can say an image better by seeing their gray values (Having the gray value around its mean value). It can be occurred by histogram equalization. Local contrast of image will increase in this method. This helps the areas of lower contrast to gain a higher contrast without affecting the global contrast (mean of the contrast of all pixels). The increased intensity values can be distributed on histogram



Fingerprint image and its histogram before normalization



Fingerprint image and its histogram after normalization

The process of histogram equalization is explained below. First, we must note down the gray values contains in an image. And next we must count the pixels having the same gray value. Note down these terms in a table. After we have to compute the probability mass function (PMF). we have to equalize all the pixel values of an image.

So PMF helps us to evaluate the probability of each pixel value in a fingerprint image. Next, we have to calculate cumulative distribution frequency (CDF). The CDF is a cumulative sum of all the probabilities present in the table. The idea of this process is to give the outcome image with linear cumulative distribution function. Now multiply each CDF values with the value of (number of levels - 1). And do round off to the result we got and note down the number of pixels corresponding to newly obtained result. If two rows got same value, then sum the values contain in the row (number of pixels). Now we got a new data. Consider the round off values as a gray value. Plot the graph between gray values and number of pixels for each gray value. When we observe, after getting new values, there is no change in global contrast. The sum of all PMF values be 1.

we can see all the intensity values lie on the right-hand side of the 0–255 scale, with slow decreasing the pixels to the left-hand side in histogram of the original image. The range of intensity values has been adjusted in balance after performing histogram equalization. The contrast between the ridges and valleys will be increased. The shape of the original histogram plot will not change. The relative position of the values along the x-axis is shifted.

5) **Image binarization:**

Image Binarization is process of converting the image into binary from grayscale. This binary image only contains white and black pixels. In this, black pixel is indicated by 0 and white pixel is indicated by 1. Image is converted into binary using threshold method with THRESHOLD_BINARY as one parameter in python OpenCV. The parameters of threshold method are image, threshold value, maximum threshold value and type. Internal process of the method is first it takes an image and converts all the pixels which having their threshold value greater than threshold value given in parameter to maximum threshold value (given in parameter). In THRESHOLD_BINARY, black pixel is indicated by 0 and white pixel is indicated by 1. Conversely, in THRESHOLD_BINARY_INV black pixel is indicated by 1 and white pixel is indicated by 0.

Binarized image of 106_2 from dataset FVC2000

6) **Morphological operations:**

Morphological operation are some basic operations performed to make some changes on foreground image. Generally, it is performed on binary images. It needs two inputs; one is the original image and another one is structuring element or kernel which decides the nature of operation. Usually, the structuring element is of size n×n and has its origin at the center pixel. It is shifted over each pixel of the image and its elements are compared with the set of the underlying pixels. If the two sets of elements match the condition respective to operator, then the pixel in original image is set to pre-defined value (0 or 1 in binary image). The conditions for dilation and erosion are explained below. Some of the morphological operations are dilation and erosion. And Opening, Closing, Gradient etc. are their variant forms also comes into morphological operations. In this project, we are going to discuss about dilation and erosion

Erosion:

The basic idea of erosion is decomposition of foreground image. It erodes away the boundaries of foreground object (And always white is considered as foreground image). The center of the kernel or structural element is placed on each pixel in the binary image and checks the condition. The condition is, if all the pixels under kernel in binary image is 1, then the pixel in image is consider as 1, otherwise it is eroded (pixel will be marked as 0).

Example:

1	1	1	1	1	1
1	1	0	0	1	1
1	0	0	0	0	1
1	1	0	0	1	1
1	1	1	1	1	1

Input image

1
1
1

Kernel

0	0	0	0	0	0
1	0	0	0	0	1
1	0	0	0	0	1
1	0	0	0	0	1
0	0	0	0	0	0

Output image

Eroded image

White is considered as a foreground image. So, black region increases, and white region and white region get eroded.

After getting eroded image, we will apply dilation operation on it. We have to take the eroded image as an input of the dilation method. This helps to clear the noise (dots) on the ridges. After binarization, some of the ridges get separated. Dilation helps to join these types of ridges. When erosion is done, the ridges get expanded (Because they are black) and touches nearest one. After performing dilation, ridges will join and returns to its original width and some dots are removed.

Dilation:
This method is completely opposite to erosion. It expands the foreground region (white is considered as foreground region for binary image). Similarly, the center of the kernel or structural element is placed on each pixel in the binary image and checks the condition. The condition is, if atleast one pixels under kernel in binary image is 1, then the pixel in image is consider as 1, otherwise it will be marked as 0.
Example:

0	0	0	0	0	0
0	0	1	1	0	0
0	1	1	1	1	0
0	0	1	1	0	0
0	0	0	0	0	0

Input image

1
1
1

Kernel

0	0	1	1	0	0
0	1	1	1	1	0
0	1	1	1	1	0
0	1	1	1	1	0
0	0	1	1	0	0

Output image

Dilated image

When we compare the dilated image with binary image, we can notice some dots on the bottom ridges are removed and ridges become clearer than the binarized one. Now the image is ready for next stage. That is thinning and feature extraction.

Feature extraction:

There are two steps involved in this stage. They are thinning and minutiae extraction. First the enhanced image get thinned and minutiae point will get extracted.

Thinning:

Thinning is one of the morphological that is utilized to expel the chosen foreground pixels from binary image, alike erosion. Especially this is useful for skeletonization. It is utilized to delete the additional pixels of ridges till its size become only one pixel wide. The outcome will be a thinned binary image. This is the first step in second stage. We have tried two methods for thinning. One is implemented by using Zhang Suen algorithm and another one standard thinning algorithm.

Zhang suen algorithm:

The Zhang suen thinning algorithm is widely used thinning algorithm. This algorithm also known as 2-step algorithm, means for each iteration it performs two sets of checking to remove pixels from the binary image. Before checking the conditions, we have to define two values.

P9	P2	P3
P8	P1	P4
P7	P6	P5

X(P1) = Number of one-to-zero transactions in neighbors of P1.

Y(P1) = Number of black pixel neighbor's around P1

Step 1:

1) The pixel P1 should be black and contains eight neighbors (P1 should not be corner or border)
2) The number of black pixels (Y(P1)) should be at least 2 and not greater than 6.
3) The number of one-to-zero transactions (X(P1)) should be one (1).
4) Minimum any one of the pixels P4, P6, P2 should be 1(white).
5) Minimum any one of the pixels P8, P4, P6 should be 1(white).

Step 2:

1) The pixel P1 should be black and contains eight neighbors (P1 should not be corner or border)
2) The number of black pixels (Y(P1)) should be at least 2 and not greater than 6.
3) The number of one-to-zero transactions (X(P1)) should be one (1).
4) Minimum any one of the pixels P4, P2, P8 should be 1(white).
5) Minimum any one of the pixels P6, P2, P8 should be 1(white).

The following pixel should be removed if it is true for either one of these steps. Ignore and go to next pixel if it is not true for both steps. This method gives the accurate result but not fast. Time complexity for this method is $O(n^2)$. So, we preferred the second step. That is standard thinning algorithm in MATLAB.

Standard thinning algorithm:

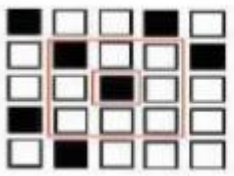
There is a standard algorithm, which plays out the thinning activity utilizing two sub cycles. This calculation is applied in MATLAB by "thin" operation under the bwmorph function. Each sub cycle starts by confirming the area of every pixel in the binary image and depending on a lot of pixel-deletion standards, it checks whether the pixel can be removed or not. This process checks until the ridge becomes one-pixel width.

Skeletonized image of 106_2

Feature extraction is the second stage to be applied on a fingerprint image. In this stage minutia are extracted from the thinned image. The technique we utilized for minutiae extraction is crossing number (CN). This CN technique extracts the minutiae points from the thinned image by checking the nearby neighborhood of each ridge pixel using window of size 3×3. After looking at the nearby neighborhood of pixels, this method gives the CN value for every pixel. CN value is mathematically defined as the half of the sum of difference between the pairs of adjacent pixels. The difference should take in positive.

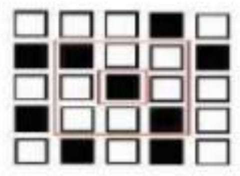
$$CN = \frac{1}{2} \times \sum_{i=1}^{8} |\mathrm{P}_{(\mathrm{i\ mod 8})} - \mathrm{P}_{(\mathrm{i+1})}|$$

Each pixel in the ridge can be divided into ridge ending, bifurcation, isolated point and crossing number depending upon their crossing number (CN value). The pixel having its CN value 1, is considered as ridge ending. It has only one neighborhood because it is the ending point. The pixel having its CN value 2, is consider as normal pixel. It has two neighborhood pixels and it locates somewhere at middle. The pixel having its CN value 3, is consider as ridge bifurcation. It has three neighborhood pixels because it is a common point when the ridge is divided into two ridges. The pixel having its CN value 4, is consider as crossing point. It has four neighborhood pixels. And the CN value can be zero. We considered that pixels as an isolated point, that can be removed in false minutiae removal step. We mainly concentrate on the pixels having their CN value 1 and 3 (Ridge ending and bifurcations).



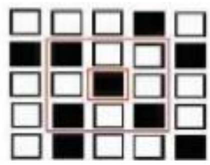
CN value = 1

Ridge ending



CN value = 2

Normal ridge pixel



CN value = 3

Ridge bifurcation

Bifurcations extracted from thinned image 106_2

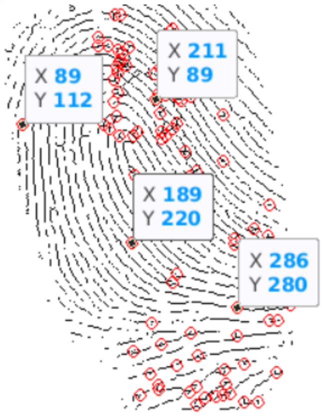
Ridge orientation:

Every minutiae point has its own property. They are expressed by some of the attributes, for example its location, orientation angle and sometimes the type of the minutiae will also be considered. Most of the algorithms only consider coordinates and orientation. The minutiae are set of coordinates and orientation. And image is set of minutiae. They are denoted by,

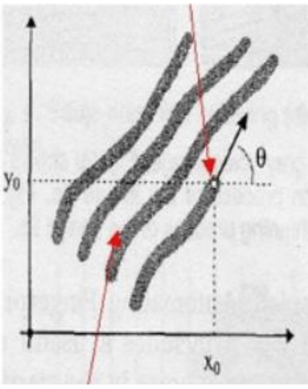
$$I_1 = \{m_1,\ m_2, m_3, \ldots\ldots, m_N\}, \text{ where } m_i = (x_i, y_i, \theta_i)$$

$$I_2 = \{m''_1,\ m''_2, m''_3, \ldots\ldots, m''_M\}, \text{ where } m''_i = (x''_i, y''_i, \theta''_i)$$

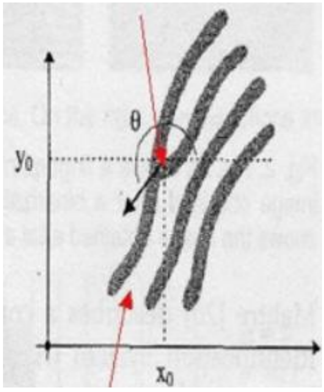
The above sets are disordered, and both may have discrete number of points (M and N may be equal or may not be.)



Coordinates



Termination angle



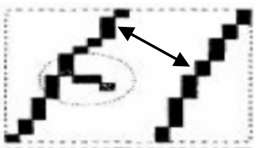
Bifurcation angle

Post processing:

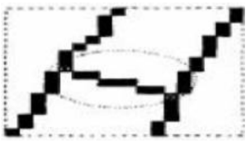
This is the final stage in fingerprint detection. In this project, there are two methods in post processing stage. One is false minutiae removal and matching process. While extracting minutiae points, the system extracts other points similar to actual minutiae because of the noise involved in it. In this stage, they are recognized through some conditions and get removed. The conditions are explained below.

False minutiae removal:

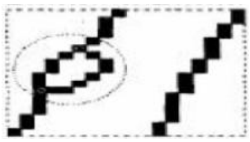
There are different types of false minutia. Some of them are spikes, bridges, holes, breaks, spurs, and ladders.



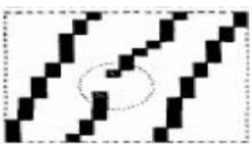
Spike



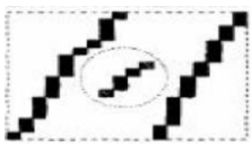
Bridge



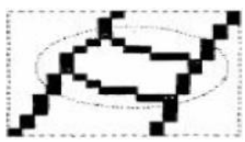
Hole



Break



Spur



Ladder

First, we have to calculate the average Inter-ridge width by considering distance between two parallel neighbor ridgees .Let us take it as a D. Each row in the thinned image will be scanned and pixels in a row should be added. Then divide the row length with the above sum to get an Inter ridge width. We will get Inter ridge width for each row. So, add them all and divide it by number of columns. Then we will get average inter-width (D).

Conditions:

1) If the separation between one bifurcation and one ending is not as much as D and two points are in same line (ridge), then it is considered as spike. It should be removed.

2) If the separation between two terminations is not as much as D and two points are in the same line (ridge), then it is considered as spur. It should be removed.

3) If the separation between two bifurcations is not as much as D and two points are in same line (ridge), then they are considered as bridge or hole or ladder. They should be removed.

And the last type of false minutiae is breaks between ridge. Indirectly, it is removed in preprocessing step (Erosion and dilation). Their we are applying erosion with kernel size 5 to increase ridge width. It results to join the breaks between ridges. And then we are going apply dilation with kernel size 3 to return to its original ridge width. The joining ridges remains same and dots on the ridge will get removed.

Minutiae matching:

It is the last step in the post processing stage. Here, the remaining minutiae points are transformed into polar coordinates and stored in a template. In this process, first we have to choose one reference minutiae point and convert all the minutiae points into polar coordinate system.

All minutiae points are in the form of (x_i, y_i, θ_i)t.

x_i – x-coordinate

y_i – y-coordinate

θ_i – orientation angle

t – type of minutiae (ridge ending or bifurcation)

All minutia points are converted into polar coordinates by applying formulas:

$$\begin{pmatrix} r_i \\ a_i \\ \theta''_i \end{pmatrix} = \begin{pmatrix} \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2} \\ tan^{-1}\left(\frac{y_i - y_r}{x_i - x_r}\right) \\ \theta_i - \theta_r \end{pmatrix}$$

Here, $(x_r,\ y_r\ , \theta_r)$ is the coordinates of reference minutia and $(r_i,\ a_i\ , \theta''_i)$ is in the polar coordinates.

r_i – Radial distance

a_i– Radial angle

θ''_i- Orientation of minutia with respective to reference minutia

Reference minutiae is marked in a new template. Based on the above parameters (Radial angle, Radial distance, and Orientation), we must locate each minutia point with respect to reference minutiae in that template. So, that template is stored in data base for verification. Till now, this process is same for every fingerprint. This process should also be applied for the fingerprint image which we have to check.

Now we must check for same minutiae points in both templates. This process is MOI (Minutiae of interest). First check the minutiae type, if they are not same, ignore that and go to next minutiae. If they are same, calculate the following.

Let us take S = number of points recorded.

1) Radail distance $= \frac{1}{S(\Sigma|r_i - r_j|)}$ for 0≤ i≤ S
2) Radial angle $= \frac{1}{S(\Sigma|a_i - a_j|)}$ for 0≤ i≤ S
3) Minutiae direction $= \frac{1}{S(\Sigma|\theta"_i - \theta"_j|)}$ for 0≤ i≤ S

Where $(r_i, a_i, \theta"_i)$ are input template attributes and $(r_j, a_j, \theta"_j)$ are template stored in database.

Now we can compute similarity score for the two sets of minutiae by adding the radial distance, radial angle, and direction. Lets α be the threshold assumed. Now compare threshold and the similarity score. Generally, the threshold value is considered as 0.8 to get accurate result. If the similarity score is more than α, then we can say that the two fingerprints are from same person (Matched). Then we can calculate matching score for this pair.

We can compute the Matching score by using:

$$\text{Matching Score} = \frac{M^2}{M_i \times M_t}$$

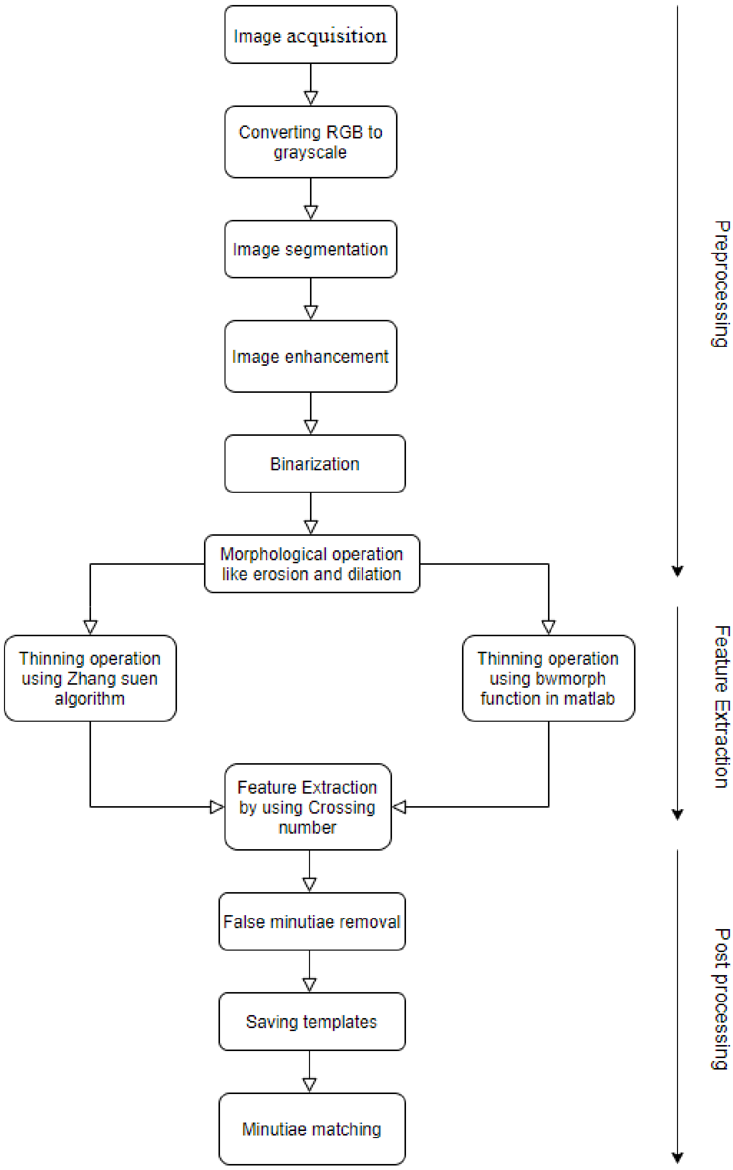
Where, M is the absolute number of matched pairs, M_i is the quantity of minutiae points in input fingerprint and M_t is the quantity of minutiae points in template stored in database.

Data collection/ Project design:

I have downloaded the two-fingerprint datasets. One is FVC2000 and another one is fingerprint colour image database. v1 from MATLAB official website. FVC2000 data set contains 80 fingerprint images of 8 different persons. In second database, there are 250 fingerprint images of 50 different persons. I have performed this algorithm on each image of different persons in both the data sets.

Project design:

Implementation flowchart



Result and discussion:

I have implemented the algorithm up to feature extraction stage. I have performed this algorithm on each image of different persons in both the data sets. For the thinning process, first I have used Zhang Suen algorithm in OpenCV. But it increases the thinning time. So later I used bwmorph inbuilt method in MATLAB. The input of the system is scanned fingerprint image. It need not to be a grayscale image (The system will convert). Each operation up to feature extraction mentioned above is performed on that image. Grayscale image, normalized image, binarized image, dilated image, thinned image and image with minutiae are showed as an output.

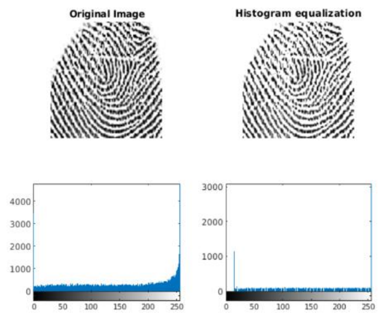
Conclusion:

The above implementation of this research project is getting clear idea about how fingerprints are detected. Although many systems for fingerprints detection exits, there is a necessity for further research in this topic to improve the reliability and performance of the current systems. Exact location of minutiae points is damaged by many factors. Among them, poor image quality gives much impact on damaging minutiae location. In this project we more concentrated on preprocessing and feature extraction stage. We have implemented the preprocessing stage which gives suitable image for thinning with simple techniques like erosion, dilation etc. The minutiae-based matching is highly realistic as, if the finger is moved even a tad it gives us big change in data set.

Appendix 1: Tables & Figures

Operations performed on another image from second database:

Original image



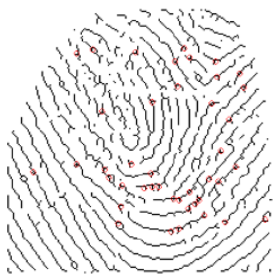
Enhanced image

Binarized image

Eroded image

Dilated image

Thinning



Feature extracted

Appendix 2: References:

1. "**Fingerprint Recognition and Matching using MATLAB",** Aliyu Tukur, Department of Electrical Engineering, Hassan Usman Katsina Polytechnic, Katsina-Nigeria.

2. **"Fingerprint Identification"** by Kuntal Barua, Samayita Bhattacharya, Dr. Kalyani Mali, University of Kalyani.

3. **"Fingerprint recognition for person identification and verification based on minutiae matching"** by Mouad.M.H.Ali, Vivek H. Mahale, Pravin Yannawar, A. T. Gaikwad from Dr. Babasaheb Ambedkar Marathwada University.

4. **"Fingerprint Identification Using Delaunay Triangulation"** George Bebist, Taisa Deaconut and Michael GeorgiopoulosS Department of Computer Science, University of Nevada, Reno Department of Electrical and Computer Engineering, University of Central Florida.

5. **"Fingerprint Image Enhancement and Minutiae Extraction"** by Raymond Thai.

6. **"Image Enhancement Techniques for Fingerprint Identification"** by Pankaj Deshmukh, Siraj Pathan, Riyaz Pathan. International Journal of Scientific and Research Publications, Volume 3, Issue 3, March 2013.

7. **"Fingerprint Verification System using Minutiae Extraction Technique"** by Manvjeet Kaur, Mukhwinder Singh, Akshay Girdhar, and Parvinder S. Sandhu. World Academy of Science, Engineering and Technology, International Journal of Computer, and Information Engineering**.**

8. **Study about Zhang Suen thinning algorithm:**
   http://rstudio-pubs-static.s3.amazonaws.com/302782_e337cfbc5ad24922bae96ca5977f4da8.html

9. **"Detecting and Counting Minutiae in Human Fingerprint"** by Ziad Abdel AlQadi, Yousf Eltous, Mohammad Abuzalata, Ghazi M. Qaryouti, Faculty of engineering technology, Jordan. Open Science Jornal.

10. Online MATLAB software: https://matlab.mathworks.com/

11. **"FINGERPRINT RECOGNITION USING MINUTIA SCORE MATCHING"** by RAVI. J, K. B. RAJA, VENUGOPAL. K. R, Dept. of Electronics and Communication Engineering, Global Academy of Technology, Bangalore.

12. **"Preprocessing and postprocessing for skeleton-based fingerprint minutiae extraction"** by Feng Zhao, Xiaoou Tang, Department of Information Engineering, The Chinese University of Hong Kong, Shatin, NT, Hong Kong.

BML MUNJAL UNIVERSITY
FROM HERE TO THE WORLD

13. **"Fingerprint Recognition Technique Based on Radial Distance and Radial Angle Matching Score"** by Sohini Banerjee**,** *Department of Computer Science and Engineering* Abacus Institute of Engineering and Management, Mogra, WB, India and Rupashri Barik Department of Information Technology, JIS College of Engineering, Kalyani, Nadia, India