© 2015 IJSRSET | Volume 1 | Issue 3 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

An Efficient Approach for Robust Fingerprint Recognition System

Rakesh Yadhav G B[*1], Choodarathnakara A L[2], Harshitha B K[3], Mohammed Zakirulla M[4]
[*1,2,3]Dept. of Electronics & Communication Engineering, GEC, Kushalnagar, Kodagu, Karnataka, INDIA
[4]Dept. of Electronics & Communication Engineering, RBMCE, Bellary, Karnataka, INDIA

ABSTRACT

In today's society, the use of electronic commerce, transaction of monetary assets and daily use of email is rapidly increasing. Along with the increased ease of purchasing and selling, there is also an increase in fraud mostly from false identification. Solutions to this problem have been in the field of biometrics, using the person's body as a form of identification. In particular, the uniqueness of fingerprints has made them very popular among law enforcement, banking establishments and commerce. Fingerprint identification using manual procedures has become a very common approach, but it has number of limitations like very low rate of positive identification, time consumption, mutilation of paper slips used for fingerprinting, etc.., rendering them ineffective. This has resulted in the dire need of speeding up the procedure and increasing its reliability by the use of computerized process. This paper addressing different procedures involved in acquisition of the fingerprint, operations of pre-processing to make the fingerprint compatible for feature extraction, feature extraction and finally authentication are discussed along with their associated difficulties.

Keywords: Biometrics, Fingerprint Image, Pattern Recognition, Image Processing and MATLAB

I. INTRODUCTION

In the current Information Technology (IT) age, identity authentication is very crucial. IT brings with it capability for electronic transaction where face-to-face or other means of personal contact is not necessary. The lack of actual contact makes identifying the real user necessary as well as difficult. Necessary because as the saying goes, in the Internet, even a monkey can be human! Difficult as it goes beyond the traditional means of identity authentication and where being anonymous and staying anonymous is the desired feature of the Internet.
Traditional means of identity authentication using tokens such as keys and cards, or Personal Recognition Numbers (PINs) and passwords are ill suited for such a task. For example, if you would like to purchase anything in the Internet, most likely you will require a credit card number to authenticate who you are and that you have means to pay for your goods. However, such a number can be obtained rather easily by hackers, not to mention the numerous fake credit cards in circulation. Bank cards need PINs to authenticate one's identity. However, in many situations, the PINs can be obtained easily either because the users wrote the PINs behind their cards or at some place or that the PINs were obtained through observation or fraudulent means. Some users have more than one bank account and thus it is challenging to remember all the PINs on top of the various PINs/passwords used daily. The most important issue is how to identify the "real" person without resorting to any complex and troublesome mechanism for verification.

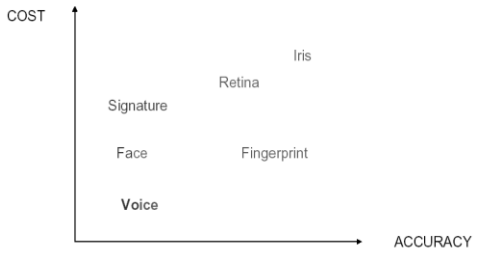


Figure 1: Comparison of different Biometric techniques

Biometrics [1] is seen as one of the best candidates to solve this problem. Essentially, biometrics is the automated approach to authenticate the identity of a person using the individual's unique physiological or

behavioural characteristics such as fingerprint, face, voice, signature etc. The different biometric techniques are compared as follows for cost versus accuracy.

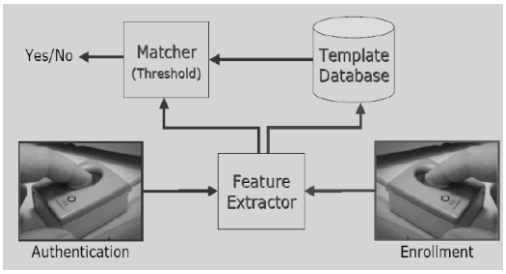


Figure 2: Overview of the complete Fingerprint Verification or Recognition System

The Fingerprint Verification and Recognition System process is roughly broken down into image enhancement, feature extraction, template generation and the actual verification or recognition step.

II. FINGER PRINT TECHNOLOGY

A fingerprint is a pattern of curving line structures, called ridges, where the skin has a higher profile than its surroundings, called the valleys. In most fingerprint images, the ridges are black and the valleys are white.

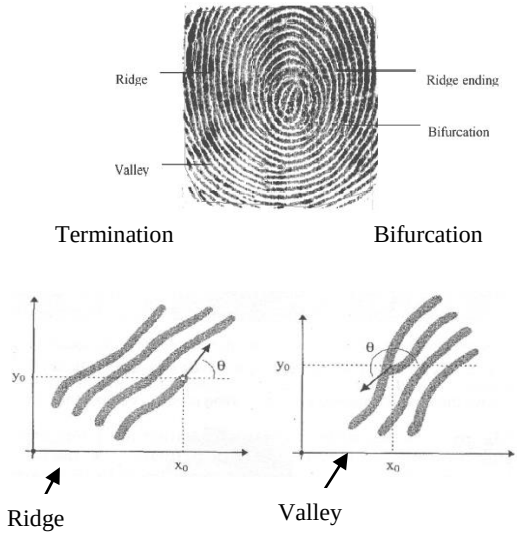


Figure 3: Different Feature Points in a Fingerprint

Fingerprint contains special features (minutiae)

- Ridge endings - a ridge that ends abruptly.
- Ridge bifurcation - a single ridge that divides into two ridges.
- Short ridges, island or independent ridge - a ridge that commences, travels a short distance and then ends.
- Ridge Enclosures - a single ridge that bifurcates and reunites shortly afterward to continue as a single ridge.
- Spur - a bifurcation with a short ridge branching off a longer ridge.
- Crossover or bridge - a short ridge that runs between two parallel ridges.

Most fingerprint recognition algorithms are typically based on extraction of minutiae points.

A. Classification of Fingerprints

Fingerprints are commonly classified as 5 different types as shown in Figure 4. Namely, Tented Arch, Arch, Right Loop, Left Loop and Whorl.

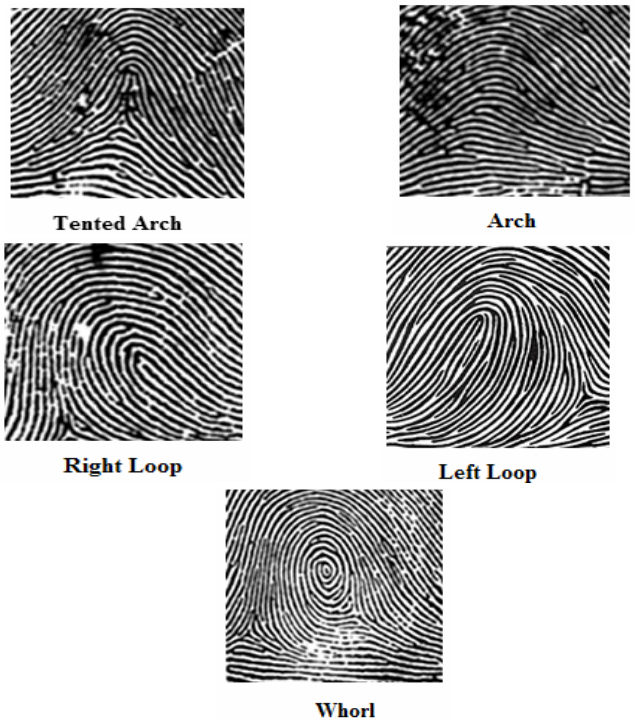


Figure 4: Fingerprint Classifications

B. Histogram Equalization

Histogram equalization is to expand the pixel value distribution of an image so as to increase the perception information. The original histogram of a fingerprint image has the bimodal type [Figure 5], the histogram after the histogram equalization occupies all the range from 0 to 255 and the visualization effect is enhanced [Figure 6].

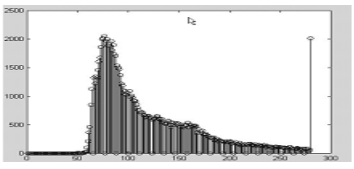


Figure 5: Original Histogram of Fingerprint Image

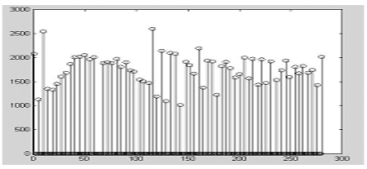
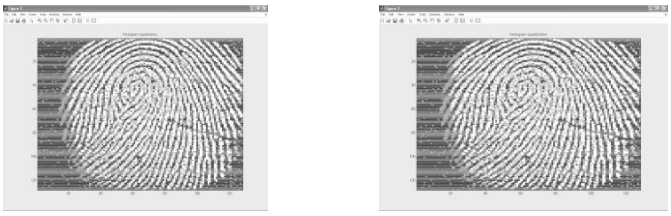


Figure 6: Histogram after Histogram Equalization



Input Image Output Image

Figure 7: Histogram Equalization

C. Fingerprint Enhancement by Fourier Transform

We divide the image into small processing blocks (32 by 32 pixels) and perform the Fourier transform according to [3].

$$F(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y) \times \exp\left\{-j2\pi \times \left(\frac{ux}{M} + \frac{vy}{N}\right)\right\} \quad (1)$$

For u = 0, 1, 2, ..., 31 and v = 0, 1, 2, ..., 31.

In order to enhance a specific block by its dominant frequencies, we multiply the FFT of the block by its magnitude a set of times,

Where the magnitude of the original FFT = abs (F (u, v)) = |F (u, v)|.

Get the enhanced block according to

$$g(x,y) = F^{-1}\left\{F(u,v) \times |F(u,v)|^{K}\right\} \quad (2)$$

Where $F^{-1}(F(u, v))$ is done by:

$$f(x,y) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} F(u,v) \times \exp\left\{j2\pi \times \left(\frac{ux}{M} + \frac{vy}{N}\right)\right\} \quad (3)$$

For x = 0, 1, 2, ..., 31 and y = 0, 1, 2, ..., 31.

The k in Equation 2, is an experimentally determined constant, which we choose k=0.45 to calculate. While having a higher "k" improves the appearance of the ridges, filling up small holes in ridges, having too high "k" can result in false joining of ridges.

Figure 8: Fingerprint Enhancement by FFT

The enhanced image after FFT has the improvements to connect some falsely broken points on ridges and to remove some spurious connections between ridges.

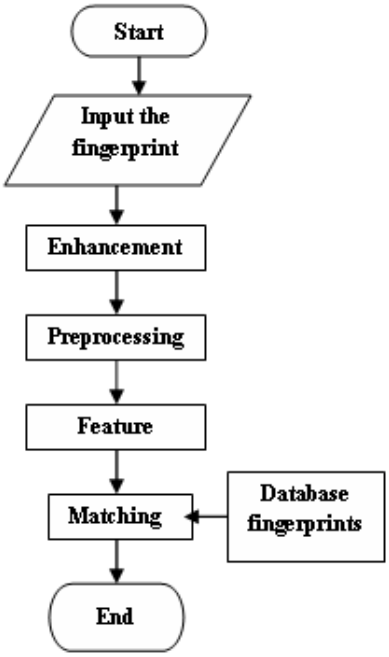
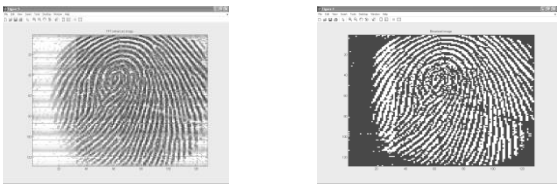


Figure 9: Flow chart of main program

D. Fingerprint Image Binarization

Fingerprint Image Binarization is to transform the 8-bit Gray fingerprint image to a 1-bit image with 1-value for ridges and 0-value for valleys. After the operation, ridges in the fingerprint are highlighted with white colour while valleys are black.



(a) (b)

Figure 10: Fingerprint image after adaptive Binarization

E. Morphological Operations

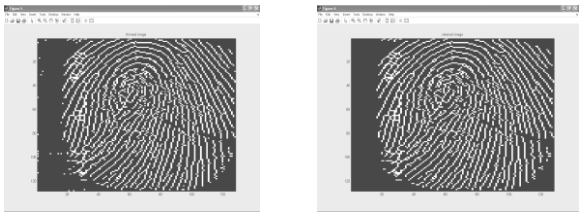
- **Thinning***:* Thinning is a morphological operation that successively erodes away the foreground pixels until they are one pixel wide.

(a) (b)

Figure 11: Result of Thinning

- **Cleaning:** Removes isolated pixels (individual 1's that are surrounded by 0's), such as the centre pixel in this pattern.

```
0 0 0
0 1 0
0 0 0
```



(a) (b)

Figure 12: Result of Cleaning

- **Removal of H Breaks**

Removes H-connected pixels. For example

```
1 1 1                 1 1 1
0 1 0    becomes      0 0 0
1 1 1                 1 1 1
```

Figure 13: Removal of H breaks

- **Removal of Spur**

Removes spur pixels. For example:

```
0 0 0 0                 0 0 0 0
0 0 0 0                 0 0 0 0
0 0 1 0    becomes      0 0 0 0
0 1 0 0                 0 1 0 0
1 1 0 0                 1 1 0 0
```

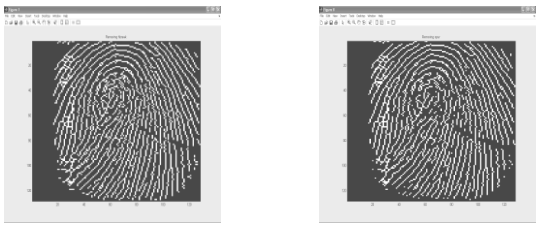


Figure 14: Removal of Spur

F. Feature Extraction

To be able to differentiate one fingerprint from another it is necessary to introduce some kind of measure. These measures are called features and the feature extraction algorithm finds and extracts characteristic features from the fingerprints have a certain degree of correlation.

The two main methods used for fingerprint feature extraction are

- Minutiae Extraction.
- Ridge Extraction.

G. Minutiae Extraction

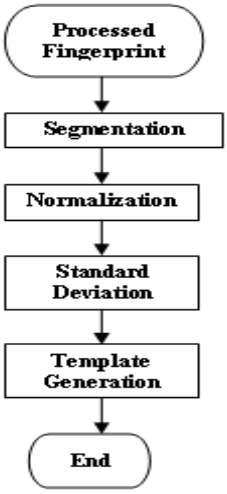


Figure 15: Flow Chart of Finger Print Extraction

The two most prominent ridge characteristics, called minutiae, are: ridge termination and edge bifurcation. A ridge ending is defined as the ridge point where a ridge ends abruptly. The minutiae are extracted by scanning the local neighbourhood of each ridge pixel in the image using a 3×3 window. The CN value is then computed, which is defined as half the sum of the differences between pairs of adjacent pixels in the eight-neighbourhood.

H. Ridge Extraction

Despite the efficacy of minutia-based fingerprint matching techniques for good-quality images captured by optical sensors, minutia-based techniques do not often perform so well on poor-quality images or fingerprint images captured by small solid-state sensors. Therefore, it is necessary to develop new fingerprint-matching techniques that utilize other features to deal with fingerprint images captured by solid-state sensors.

I. Normalization

Normalization is used to standardize the intensity values in an image by adjusting the range of grey-level values so that it lies within a desired range of values. Image normalization attempts to spread the pixel intensities over the entire range of intensities. Normalization offsets

and rescales image so that the minimum value is 0 and the maximum value is 1. If the image is color the image is converted to HSI and the value/intensity component is normalized to 0-1 before being converted back to RGB**.**

J. **Segmentation**

Segmentation is the process of separating the foreground regions in the image from the background regions. The foreground regions correspond to the clear fingerprint area containing the ridges and valleys, which is the area of interest. The background corresponds to the regions outside the borders of the fingerprint area, which do not contain any valid fingerprint information.

Segmentation identifies ridge regions of a fingerprint image and returns a mask identifying this region. It also normalizes the intensity values of the image so that the ridge regions have zero mean, unit standard deviation. This breaks the image up into number of blocks of size 8×8, and evaluates the standard deviation in each region. If the standard deviation is above the threshold it is deemed part of the fingerprint. Note that the image is normalized to have zero mean, unit standard deviation prior to performing this process so that the threshold we specify is relative to a unit standard deviation.

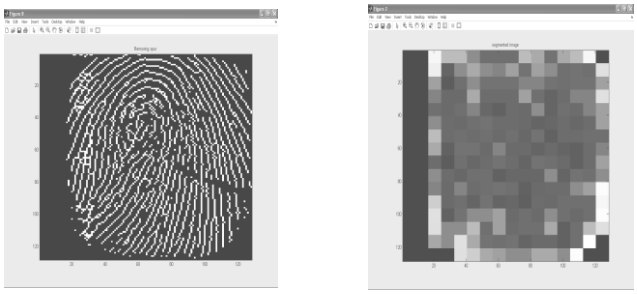


Figure 16: Result of Segmentation

K. **Matching**

Reliably matching fingerprint images is an extremely difficult problem, mainly due to the large variability in different impressions of the same finger (i.e., large intra-class variations). The main factors responsible for the intra-class variations are: displacement, rotation, partial overlap, non-linear distortion, variable pressure, changing skin condition, noise, and feature extraction errors.

- **Correlation-based matching**: Two fingerprint images are superimposed and the correlation (at the intensity level) between corresponding pixels is computed for different alignments (e.g., various displacements and rotations).

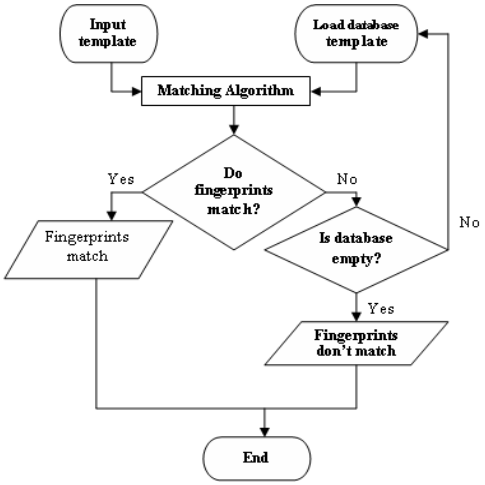


Figure 17: Flow Chart for Matching

- **Minutiae-based matching**: Minutiae are extracted from the two fingerprints and stored as sets of points in the two-dimensional plane.
- **Ridge feature-based matching**: Minutiae extraction is difficult in very low-quality fingerprint images, whereas other features of the fingerprint ridge pattern (e.g., local orientation and frequency, ridge shape, texture information) may be extracted more reliably than minutiae, even though their distinctiveness is generally lower. The approaches belonging to this family compare fingerprints in term of features extracted from the ridge pattern.

L. Hamming distance

The Hamming distance gives a measure of how many bits are same between two bit patterns. Using the Hamming distance of two bit patterns, a decision can be made as to whether the two patterns were generated from different fingerprints or from the same one.

In comparing the bit patterns X and Y, the Hamming distance, HD, is defined as the sum of disagreeing bits (sum of the exclusive-OR between X and Y) over N, the total number of bits in the bit pattern.

$$HD = \frac{1}{N} \sum X_j (XOR) Y_j N^{j-1} \quad (4)$$

M. Weighted Euclidean Distance

The Weighted Euclidean Distance (WED) can be used to compare two templates, especially if the template is composed of integer values. The weighing Euclidean distance gives a measure of how similar a collection of values are between two templates. This metric is employed by Zhu and is specified as;

$$WED(k) = \sum_{i-1}^{N} \frac{(f_i - f_i^{(k)})^2}{(\delta_i^{(k)})^2} \quad (5)$$

Where, f_i is i^{th} feature of the unknown fingerprint, and is the i^{th} feature of fingerprint template, k and is the standard deviation of the i^{th} feature in fingerprint template k. The unknown fingerprint template is found to match fingerprint template k, when WED is a minimum at k.

N. Normalized Correlation

WED makes use of normalized correlation between the acquired and database representation for goodness of match. This is represented as;

$$\frac{\sum_{i-1}^{n} \sum_{j-1}^{m} (p_1[i,j] - \mu_1)(p_2[i,j] - \mu_2)}{nm\,\sigma_1\sigma_2} \quad (6)$$

where, p_1 and p_2 are two images of size $n \times m$, μ_1 and σ_1 are the mean and standard deviation of p_1, and μ_2 and σ_2 are the mean and standard deviation of p_2.

O. Averaging Method

This paper performs matching of an input fingerprint with the set of fingerprints stored in the database using the averaging method with threshold criterion.. If the difference is greater than or equal to threshold then the fingerprints don't match and the above process is repeated for the rest of the fingerprints in the database until it finds a match.

III. CONCLUSION

Summarizing the features of this paper, it may well be said that it has proven to be an efficient approach towards design of a robust fingerprint recognition system. The designed method has been proven to work effectively. Experiments were then conducted using a combination of both synthetic test images and real fingerprint images in order to provide a well balanced evaluation on the performance of the implemented algorithm. The use of synthetic images has provided a more quantitative and accurate measure of the performance. Whereas real images rely on qualitative measures of inspection, but can provide a more realistic evaluation as they provide a natural representation of fingerprint imperfections such as noise and corrupted elements. The results of the simulation have been accurate and most importantly implementable. One of the greatest virtues of this software has been its ability to deal with even images obtained from ink pads. This makes its proficiency much more enhanced.

IV. SCOPE FOR FUTURE WORK

It is appropriate to conclude this paper with a brief summary of the future directions this work could do. When extracting the feature vectors, the wedge and ring classifiers could be adapted to handle rotational differences which could improve the robustness of the system. The program could be used to perform matches on other images. For instance, character recognition could be performed. The only requirement is that the letters have enough contrast and possess a large enough spacing between minutiae's. The system would perform extremely well with large printed images. The program could also be written in C to allow portability across different operating systems. The final task is to apply the program to a real world task. The current matching algorithm isn't very accurate with low quality images; however it could be integrated with other recognition systems such as face recognition. The resulting combination would be a formidable security system. Future scope in this field is practically unlimited because it involves security and recognition. Voter ID cards, ATM's, Passports, Driving License, Mobile communication, Door locking facilities etc., are some of the day to day fields of application.

V. REFERENCES

[1] A.K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1988.

[2] R.C. Gonzalez and R.E. Woods, Digital Image Processing (2nd edition), Prentice-Hall, Englewood Cliffs, NJ, 2002.

[3] R.C. Gonzalez and R.E. Woods, Digital Image Processing using MATLAB, Prentice-

[4] Hall, edition 2005.

[5] Ratha N.K., Chen S., and Jain A.K., Adaptive Flow Orientation-Based Feature Extraction In Fingerprint Images, Pattern Recognition, Vol.28, No. 11, pp.1657-1672, 1995.

[6] Aditya Vailaya, Member, IEEE, HongJiang Zhang, Senior Member, IEEE, Changjiang Yang, Feng-I Liu, and Anil K. Jain, Fellow, IEEE Automatic Image Orientation Detection.

[7] A. K. Jain, S. Prabhakar, L.Hong, and S.Pankanti, Filter Bank-Based Fingerprint Matching, IEEE Trans. Image Processing Vol.9, No.5, pp.846-859, 2000.

[8] www.biometrics.com

[9] www.cse.msu.adu/rgroups/biometrics

[10] www.dsprelated.com

[11] www.mathworks.com