International Journal of
Emerging Research in Management &Technology
ISSN: 2278-9359 (Volume-3, Issue-8)

Research Article **August 2014**

Fingerprint Recognition by Classification Using Neural Network and Matching Using Minutia (Fingerprint Recognition by NNMM Method)

Samayita Bhattacharya
Department of Computer Science & Engineering,
University of Kalyani, Kalyani, West Bengal, India

Kalyani Mali
Department of Computer Science & Engineering,
University of Kalyani, Kalyani, West Bengal, India

Abstract-

Biometrics is one of the biggest tendencies in human identification. The fingerprint is the most widely used biometric. However considering the automatic fingerprint recognition a completely solved problem is a common mistake. The global level structures consist of many ridges to form some specific shape such as arch, loop, and whorl. Local level structures are called minutiae, which further classified as either endpoints or bifurcations. Either of which can be used to identify the fingerprint, our approach uses both methods. We have used soft computing for classification of fingerprints.

Key-words- Fingerprint, Global Structure, Local Structure, Average Gradient, Core, Delta, Point Orientation, Singular Point, Directional Field, Ridge, Valley, Bifurcation, Ridge Ending, Minutia, Soft Computing, Neural Network, Self Organizing Map.

I. INTRODUCTION

Fingerprints are the graphical flow-like ridges present on human fingers. Finger ridge configurations do not change throughout the life of an individual except due to accidents such as bruises and cuts on the fingertips. This property makes fingerprints a very attractive biometric identifier. Fingerprint-based personal identification has been used for a very long time [1]. Owning to their distinctiveness, stability durability, and convenience, fingerprints are the most widely used biometric features.

The fingerprint is a duplicate of a fingertip epidermis when a person touches a smooth surface, the fingertip epidermis characteristic transferred to the surface. The pattern of the ridges and valleys on the human fingertips forms the fingerprint images. Analyzing this pattern at different levels reveals different types of features that are, global feature and local feature.

Global features shape a special pattern of ridge and valleys, called singularities or Singular Point (SP) and the important points are the core and the delta. The core defined as the most point on the inner most ridges and a delta defined as the centre point where three different directions flows meet. The SP provides important information for fingerprint classification, fingerprint matching and fingerprint alignment.

Local features, so-called minutia are an important feature for fingerprint matching.

Fingerprint patterns are full of ridges and valleys. The information of the ridge structures can be treated as three levels. At the coarse level, the number and the relative positions of singular points, including cores and deltas, are concerned for classification. At the fine level, the minutiae, a group of ridge endings and bifurcations, are used as the features for matching. Between the above two levels, the middle level also contains important information, including local ridge orientation (LRO) and local ridge frequency (LRF). Conventionally, only the structures of LROs are used to find the singular points for classification or to enhance ridge structures for minutiae extraction.

There are more than 100 different types of local ridge structures that have been identified.

Although other approaches are possible, like, for instance, the hashing technique in the minutiae domain, the first step in an identification system is often continuous classification of fingerprints. This reduces the partition of the database to be searched for matches. To facilitate high-performance classification, algorithms for accurate singular-point estimation are needed. Singular point detection is a critical process for both fingerprint matching and fingerprint classification. The process of singular points detection must be fast and robust; otherwise, the performance of the whole fingerprint recognition system would be influenced heavily.

In high level fingerprint classification algorithms, extracting the number and precise location of singular points (SP), namely core and delta points are of great importance. According to the number and location of these robust characteristics, fingerprints can be classified in to main groups; arch, right loop, left loop, and whorl.

Using high-level classification process can efficiently reduces the search area in large fingerprint databases and therefore speeds up the subsequent matching algorithm.

Singular points detection is the most challenging and important process in biometrics fingerprint verification and identification systems. Singular points are used for fingerprint classification, fingerprint matching and fingerprint alignment.

We have used soft computing approach to classify fingerprints. The feature of Self Organizing Map to provide topologically preserved mapping from input to output spaces, makes it suitable to be used for classification.

Nowadays, most automatic fingerprint identification systems (AFIS) are based on matching minutiae, which are local ridge characteristics in the fingerprint pattern. The two most prominent minutia types are ridge ending and ridge bifurcation. Based on the features that the matching algorithms use, fingerprint matching can be classified into image-based and graph-based matching.
Image-based matching [2] uses the entire gray scale fingerprint image as a template to match against input fingerprint images. The primary shortcoming of this method is that matching may be seriously affected by some factors such as contrast variation, image quality variation, and distortion, which are inherent properties of fingerprint images. The reason for such limitation lies in the fact that gray scale values of a fingerprint image are not stable features.
Graph-based matching [5], [8] represents the minutiae in the form of graphs. The high computational complexity of graph matching hinders its implementation. To reduce the computational complexity, matching the minutiae sets of template and input fingerprint images can be done with point pattern matching. Several point pattern matching algorithms have been proposed and commented in the literature [3], [4], [9], [11], [12].
Fingerprint system can be separated into two categories *Verification* and *Identification.*
Verification system authenticates a person's identity by comparing the captured biometric characteristic with its own biometric template(s) pre-stored in the system. It conducts one-to-one comparison to determine whether the identity claimed by the individual is true. A verification system either rejects or accepts the submitted claim of identity.
Identification system recognizes an individual by searching the entire template database for a match. It conducts one-to-many comparisons to establish the identity of the individual. In an identification system, the system establishes a subject's identity (or fails if the subject is not enrolled in the system database) without the subject having to claim an identity.
In order to implement a successful algorithm of this nature, it is necessary to understand the topology of a fingerprint. A fingerprint consists of many ridges and valleys that run next to each other, ridges are shown in black and valleys are shown in white. The ridges bend in such ways as to form both local and global structures; either of which can be used to identify the fingerprint. The global level structures consist of many ridges that form arches, loops, whirls and other more detailed classifications, as shown in Figure 2. Global features shape a special pattern of ridge and valleys. On the other hand, the local level structures, called minutiae, are further classified as either endpoints or bifurcations. Minutiae are also given an associated position and direction, as shown in Figure 4.
We have used Self Organizing Map for classification and the matching procedure is based on minutia, as well as on global level structure for finding a reference point by which alignment of two template is to be accomplished.
In addition, scanned fingerprints are subject to distortions that must also be taken into account including rotation, translation, non-linear scaling and extraneous or missing minutiae between matching fingerprints. This creates difficulty in the matching phase because it causes the minutiae to differ between two identical fingerprints. We have introduced position invariant features to facilitate the recognition.
Most approaches to recognizing a fingerprint involve five basic stages:
(i) acquisition, where the image is obtained from hardware or a file;
(ii) pre-processing, which may include thinning, noise reduction, image enhancements and error correction;
(iii) structural extraction, where global and local structures may be found;
(iv) post-processing, where the structures are converted into a more useful format;
(v) and then matching, where fingerprints are compared against a database.

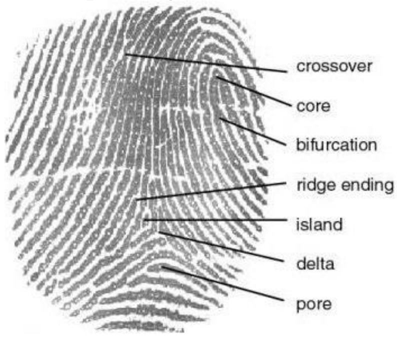
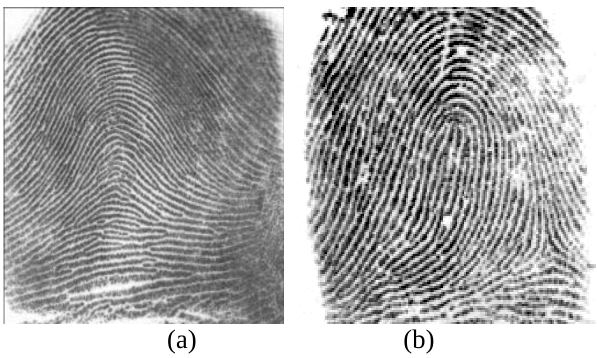


Fig 1: A fingerprint sample



(a) (b)

(c) (d)

Fig 2: Fingerprint types: (a) Arch, (b) Left Loop, (c) Right Loop, (d) Whorl

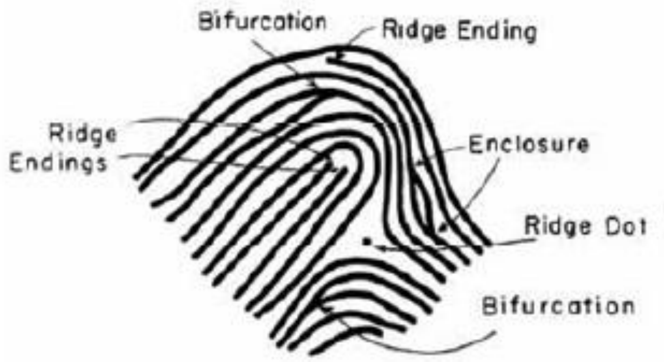
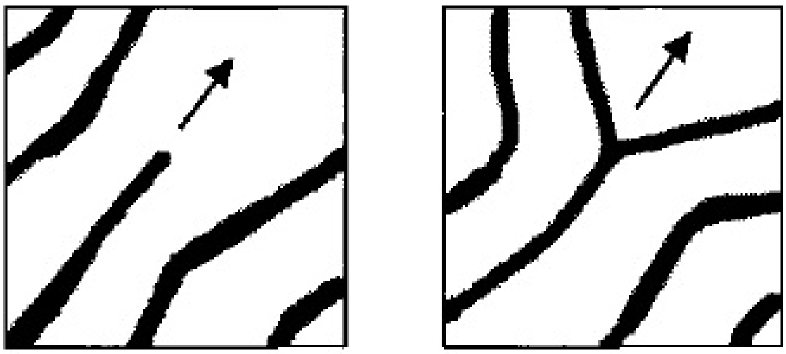


Fig 3: Fingerprint image showing different ridge features



(a) (b)

Figure 4: Types of fingerprint minutiae and their respective directions.

(a) an endpoint, (b) a bifurcation

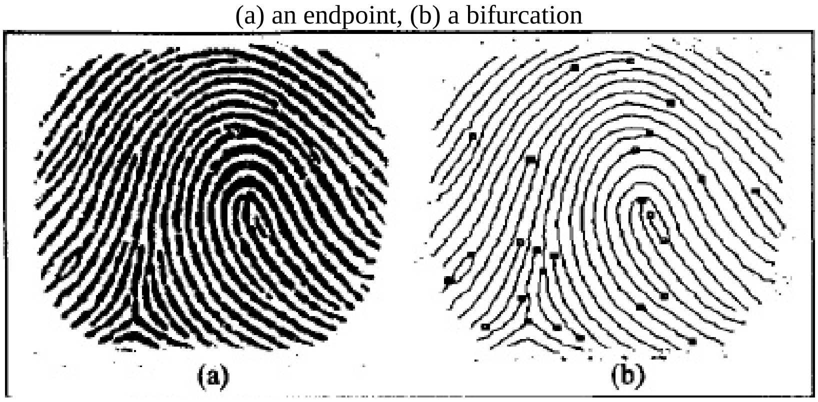


Figure 5: (a) Original fingerprint. (b) Detected minutia.

II. CLASSIFICATION

Classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known. We can use soft computing tools to classify fingerprints, and map them into major groups (i.e. arch, left loop, right loop, and whorl).

A. Soft Computing

Conventional computing or often called as hard computing, requires a precisely stated analytical model and often a lot of computation time. Many analytical models are valid for ideal cases, and real world problems exist in a non-ideal environment.

Soft computing differs from conventional (hard) computing in that, unlike hard computing, it is tolerant of imprecision, uncertainty, partial truth, and approximation. In effect, the role model for soft computing is the human mind. The guiding principle of soft computing is: Exploit the tolerance for imprecision, uncertainty, partial truth, and approximation to achieve tractability, robustness and low solution cost. Soft computing may be viewed as a foundation component for the emerging field of conceptual intelligence. Few soft computing tools are: Fuzzy Systems, Neural Networks, Evolutionary Computation, Machine Learning and Probabilistic Reasoning.

B. Neural Network

The term neural network was traditionally used to refer to a network or circuit of biological neurons. The modern usage of the term often refers to artificial neural networks, which are composed of artificial neurons or nodes. Artificial neural networks are composed of interconnecting artificial neurons (programming constructs that mimic the properties of biological neurons). For many years, artificial neural networks (ANNs) have been studied and used to model information processing systems based on or inspired by biological neural structures. They not only can provide solutions with improved performance when compared with traditional problem-solving methods, but also give a deeper understanding of human cognitive abilities. Among various existing neural network architectures and learning algorithms, Kohonen's self organizing map (SOM) is one of the most popular neural network models.

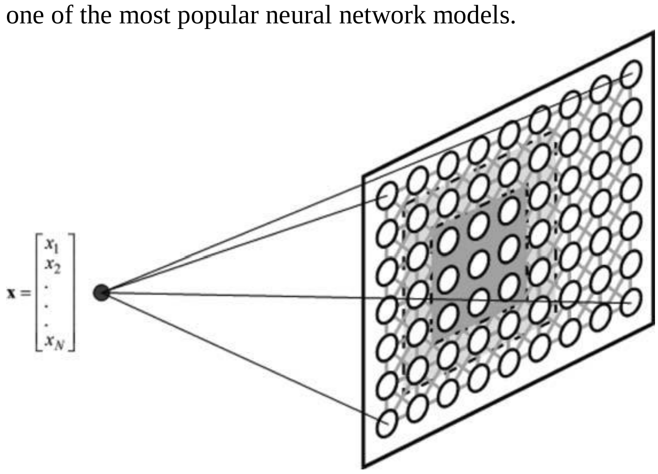


Figure 6: Kohonen's self-organizing map model. The input is connected to every cell in the postsynaptic sheet (the map). The learning makes the map localized, in other words different local fields will respond to different ranges of inputs. The lateral excitation and inhibition connections are emulated by a mathematical modification, namely local sharing, to the learning mechanism (so there are no actual connections between cells – grey lines are used to indicate these virtual connections)

Developed for an associative memory model, it is an unsupervised learning algorithm with a simple structure and computational form.

Self-organization in general is a fundamental pattern recognition process, in which intrinsic inter- and intra-pattern relationships among the stimuli and responses are learnt without the presence of a potentially biased or subjective external influence.

The SOM can provide topologically preserved mapping from input to output spaces. It is mainly used for data clustering and feature mapping. The learning process involves updating network architecture and connection weights so that a network can efficiently perform a specific classification/clustering task.

C. The SOM Algorithm

The SOM uses a set of neurons, often arranged in a 2-D rectangular or hexagonal grid, to form a discrete topological mapping of an input space, $\mathbf{X} \in R^n$.

At the start of the learning, all the weights $\{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M\}$ are initialized to small random numbers. $\mathbf{w}_i$ is the weight vector associated to neuron i and is a vector of the same dimension – n – of the input, M is the total number of neurons, and let $\mathbf{r}_i$ be the location vector of neuron i on the grid.

Then the algorithm repeats the steps shown in Algorithm, where $\eta(\nu, k, t)$ is the neighbourhood function, and Ω is the set of neuron indexes.

Although one can use the original stepped or top-hat type of neighbourhood function (one when the neuron is within the neighbourhood; zero otherwise), a Gaussian form is often used in practice – more specifically

$$\eta(\nu, k, t) = exp[-\frac{\|\mathbf{r}_\nu - \mathbf{r}_k\|^2}{2\sigma(t)^2}]$$

with σ representing the effective range of the neighbourhood, and is often decreasing with time.

Self-Organizing Map algorithm

repeat

1. At each time t, present an input $\mathbf{x}(t)$, and select the winner,

$$\nu(t) = \arg\min_{k \in \Omega} \| \mathbf{x}(t) - \mathbf{w}_k(t) \|$$

2. Update the weights of the winner and its neighbours,

$$\Delta \mathbf{w}_k(t) = \alpha(t)\eta(\nu, k, t)[\mathbf{x}(t) - \mathbf{w}_\nu(t)]$$

until the map converges

The coefficients $\{\alpha(t), t \geq 0\}$, termed the 'adaptation gain', or 'learning rate', are scalar-valued, decrease monotonically, and satisfy:
(*i*) $0 < \alpha(t) < 1$; (*ii*) $\lim t\to\infty \Sigma\, \alpha(t)\to\infty$; (*iii*) $\lim t\to\infty \Sigma\, \alpha2(t) < \infty$;

The SOM algorithm vector-quantizes or clusters the input space and produces a map which preserves topology.
It can also be and has been used for classification. In this case, the map is trained on examples of known categories.
The nodes are then classified or labelled so that the map can be used to classify unseen samples.

III. APPROACH TO EXTRACT REFERENCE REGION

In Fig. 1, a fingerprint is depicted. The information carrying features in a fingerprint are the line structures, called ridges and valleys. In this figure, the ridges are black and the valleys are white.
It is possible to identify two levels of detail in a fingerprint. The directional field (DF), shown in Fig. 7b, describes the coarse structure, or basic shape, of a fingerprint. It is defined as the local orientation of the ridge valley structures.
The DF is, in principle, perpendicular to the gradients. However, the gradients are orientations at pixel scale, while the DF describes the orientation of the ridge-valley structures, which is a much coarser scale. Therefore, the DF can be derived from the gradients by performing some averaging operation on the gradients, involving pixels in some neighborhood [13].

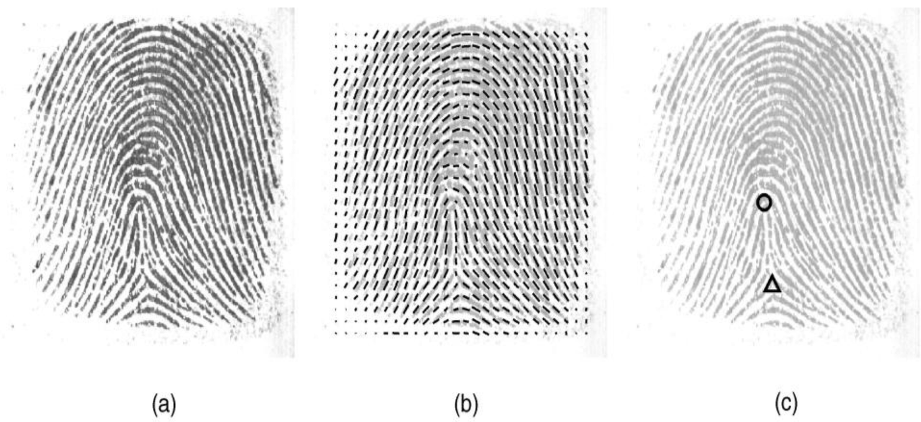


Figure 7: Examples of a fingerprint, its directional field and its singular points: (a) fingerprint, (b) directional field, and (c) singular points.

Various methods used to estimate the DF from a fingerprint are known from literature. They include matched-filter approaches [9], [14], [15], methods based on the high-frequency power in three dimensions [16], 2- dimensional spectral estimation methods [15], and micropatterns that can be considered binary gradients [10]. These approaches do not provide as much accuracy as gradient based methods, mainly because of the limited number of fixed possible orientations. This is especially important when using the DF for tasks like tracing flow lines. The gradient-based method was introduced in [7] and adopted by many researchers, see, e.g., [9], [17], [18], [20]. The elementary orientations in the image are given by the gradient vector $[Gx(x,y)\ Gy(x,y)]^T$, which is defined as:

$$\begin{bmatrix} Gx(x,y) \\ Gy(x,y) \end{bmatrix} = \text{sign}\,(Gx)\nabla I(x,y) = \text{sign}\,(\partial I(x,y)/\partial x) \begin{bmatrix} (\partial I(x,y)/\partial x) \\ (\partial I(x,y)/\partial y) \end{bmatrix} \quad \text{.......... (1)}$$

where I(x,y) represents the gray-scale image. This equation is used to calculate the elementary orientations in the image by the gradient vector.

The first element of the gradient vector has been chosen to always be positive. The reason for this choice is that in the DF, which is perpendicular to the gradient, opposite directions indicate equivalent orientations.
Since the image has background noise, the minutia extraction algorithm may generate minutiae outside the fingerprint area. So selection of the interest area is one important step.
This step is carried in few phases:
(a) divide the image into blocks,
(b) find the average gradient of each block,
(c) find the position of the image where the average gradient of two successive blocks has the zero crossing and the maximum absolute value,
(d) take the approximate middle point of these two particular blocks as the reference point,
(e) crop out a suitable region around this reference point.

IV. APPROACH TO EXTRACT MINUTIA

The minutiae provide the details of the ridge-valley structures, like ridge-endings and bifurcations. Minutia are, for instance, used for fingerprint matching, which is a one-to-one comparison of two fingerprints.
Minutia detection is a trivial task. Without a loss of generality, we assume that if a pixel is on a thinned ridge (eight-connected), then it has a value 1, and 0 otherwise.
Let (x, y) denote a pixel on a thinned ridge, and N_0, N_1, ..., N_7 denote its eight neighbours.
A pixel (x, y) is a ridge ending if $(\sum_{i=0}^{8} Ni) = 1$ and a ridge bifurcation if $(\sum_{i=0}^{8} Ni) > 2$
However, the presence of undesired spikes and breaks present in a thinned ridge map may lead to many spurious minutiae being detected. Therefore, before the minutia detection, a smoothing procedure is needed to remove spikes and to join broken ridges.

A. Image Acquisition

The first stage of any vision system is the image acquisition stage. *Image acquisition* is hardware dependent. A number of methods are used to acquire fingerprints. Among them, the inked impression method remains the most popular one. Inkless fingerprint scanners are also present eliminating the intermediate digitization process [6].
The basic two-dimensional image is a monochrome (greyscale) image which has been *digitized*. Describe image as a two-dimensional light intensity function $f(x,y)$ where x and y are spatial coordinates and the value of f at any point (x, y) is proportional to the brightness or grey value of the image at that point.

A digitized image is one where

- spatial and greyscale values have been made discrete
- intensity measured across a regularly spaced grid in x and y directions
- intensities sampled to 8 bits (256 values)

The method chosen for acquisition of a fingerprint image depends on many different factors, including the cost and reliability of an input device.

B. Pre-processing

This is an essential part of fingerprint recognition. In this step the image is made ready for the actual matching. The input of this phase is the original fingerprint image and the final output of this step is the minutia of that image.
Our proposed algorithm for pre-processing is as followed.

1) Fingerprint Image Enhancement

Three types of degradations affect the quality of the fingerprint image. The ridges get some gaps; parallel ridges connected due to noise and natural effect to the finger like cuts, wrinkles and injuries. The Fingerprint enhancement is anticipated to improve the contrast between ridges and valleys and reduce noises in the fingerprint images.
High quality fingerprint image is very important for fingerprint verification or identification to work properly. In real life, the quality of the fingerprint image is affected by noise like smudgy area created by over-inked area, breaks in ridges created by under-inked area, changing the positional characteristics of fingerprint features due to skin resilient in nature, dry skin leads to fragmented and low contrast ridges, wounds may cause ridge discontinuities and sweat on fingerprints also leads to smudge marks and connects parallel ridges.

2) Noise Reduction

Noise is an unwanted perturbation to a wanted signal. Image noise is generally regarded as an undesirable by-product of image capture. Noise reduction is the process of removing noise from a picture (here it is the fingerprint image).
We have checked and used different types of filtering methods like median filter, global and adaptive thresholding to reduce the noise [23].

3) Image Normalization

The objective of this stage is to decrease the dynamic range of the gray scale between ridges and valleys of the image in order to facilitate the processing of the following stages.

The processing of fingerprint normalization can reduce the variance in gray-level values along ridges and valleys by means of adjust the gray-level values to the predefined constant mean and variance. And normalization can remove the influences of sensor noise and gray-level deformation.

Let I(i,j) denote the gray-level value of pixel (i,j) in acquired image, the size of fingerprint image is m x n , M and V are the estimated mean and variance of input fingerprint image, respectively, and N(i, j) denote the normalized gray-level value at pixel (i, j). The normalized image is defined as follows:

$$N(i,j)=\begin{cases} M_0+\sqrt{\left(\frac{V_0}{V}\right)(I(i,j)-M)(I(i,j)-M)}, if\ I(i,j)>M \\ M_0-\sqrt{\left(\frac{V_0}{V}\right)(I(i,j)-M)(I(i,j)-M)}, \quad otherwise \end{cases} \tag{2}$$

where M_0, and V_0 are the expected mean and variance values, respectively. Normalization is a pixel-wise operation and does not change the ridge and valley structures.

4) Selection of the Interest Region

Since the image has background noise, the algorithm may generate minutia outside the fingerprint area. So selection of the interest area is one important step.

This step is carried in few phases: (a) divide the image into blocks, (b) find the average gradient of each block, (c) find the position of the image where the average gradient of two successive blocks has the zero crossing and the maximum absolute value, (d) take the approximate middle point of these two particular blocks as the reference point, (e) crop out a suitable region around this reference point.

Equation No. 1 is used to calculate the elementary orientations in the image by the gradient vector.

Most authors process fingerprints block-wise for calculating Directional Field (DF) [4], [21]. This means that the directional field is not calculated for all pixels individually. Instead, the average DF is calculated in blocks of, for instance, in our approach we used 8 by 8 pixel block.

We are using gradient calculation method to calculate the reference point. The reference point is being calculated by calculating the average gradient of 8 x 8 pixel block of the fingerprint image [7], [10].

5) Binarization

In the pre-processing stage, the image is converted from greyscale to black and white. This is done by calculating the average background intensity and subtracting this value from the greyscale image.

Next greyscale threshold (basic global and adaptive thresholding) is calculated so pixels above this value become black, and the ones below become white [23].

6) Thinning

Next the ridges must be thinned to a width of one-pixel. In this step two consecutive fast parallel thinning algorithms are applied, in order to reduce to a single pixel the width of the ridges in the binary image. These operations are necessary to simplify the subsequent structural analysis of the image for the extraction of the fingerprint minutiae. The thinning must be performed without modifying the original ridge structure of the image.

During this process, the algorithms cannot miscalculate beginnings, endings and or bifurcation of the ridges, neither ridges can be broken.

7) Minutia Extraction

In the last stage, the minutiae from the thinned image are extracted, obtaining accordingly the fingerprint biometric pattern. This process involves the determination of:

i) whether a pixel, belongs to a ridge or not and,

ii) if so, whether it is a bifurcation, a beginning or an ending point, obtaining thus a group of candidate minutiae.

Next, all points at the border of the interest region are removed.

8) Cancellation of improper minutia

This is an important step of minutia based fingerprint reorganization system. In this step, the improper minutia which are mainly result of spurious noise of input image, are cancelled.

9) Position Invariant Feature of Minutia

Ideally, two sets of planar point patterns can be aligned completely by two corresponding point pairs. A true alignment between two point patterns can be obtained by testing all possible corresponding point pairs and selecting the optimal one.

We have added the position invariant feature by aligning the fingerprints, by matching the two nearest minutia, and then the image such a way that the third nearest minutia matches, in case there is no match the fingerprints are different.

V. HIERARCHICAL CLASSIFICATION

We have used SOM to do the hierarchical classification (to major groups like loop, whorl, arch and then classify the fingerprints further down, to left loop and right loop, plain arch and tented arch, etc). We have used 40% of the data to train the system. The features used were the minutia, the number, location and direction of it. If two fingerprints don't belong to the same class, no further steps are performed.

VI. MATCHING

Matching is a key operation in the current fingerprint identification system. One of the most important objectives of fingerprint systems is to achieve a high reliability in comparing the input pattern with respect to the database pattern. 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. Therefore, fingerprints from the same finger may sometimes look quite different whereas fingerprints from different fingers may appear quite similar.
A minutia matching essentially consists of finding the alignment between the template and the input minutia sets that result in the maximum number of minutia pairings. In Minutia based matching the similarity between the input and stored template are computed.

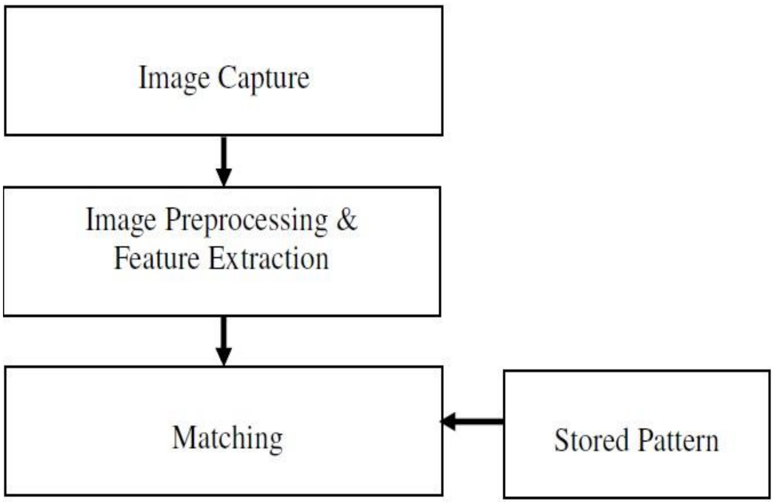


Figure 8***:*** Fingerprint Matching steps.

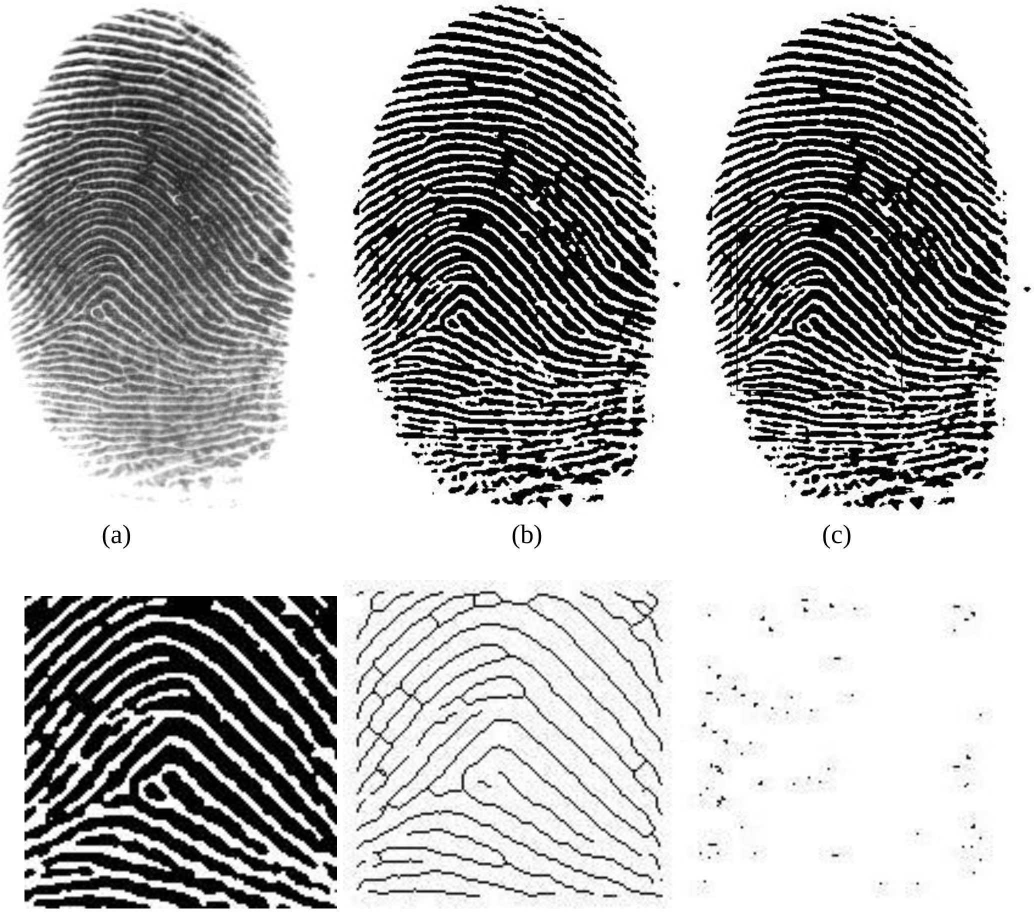
VII. EXPERIMENTAL RESULT

This Fingerprint Recognition System works for two types of matching. One fingerprint image is fed into the system to check (a) whether it belongs to a particular database and if so then for which entry/entries of the database (Identification), or (b) if it confirms to be fingerprint of a particular person (Verification). This project further automatically adds new fingerprint to the database, if the fingerprint to be matched does not exist in the database previously, for probable future use.
At first we removed the noise by median filtering followed by basic adaptive global thresholding.
Our proposed method is based on pixel to pixel matching for minutia. In this method the fingerprint image is being cropped with respect to a particular point (reference point) of the image; this cropped area is called the region of interest. In our system, the region of interest is taken as the 68 x 68 pixel block around the reference point.
We have processed fingerprints block-wise for calculating DF. This means that the directional field is not calculated for all pixels individually. Instead, the average DF is calculated in blocks of, for instance, in our approach we used 8 by 8 pixel block. We are using gradient calculation method to calculate the reference point. The reference point is being calculated by calculating the average gradient of 8 x 8 pixel block of the fingerprint image. For the maximum value of the average gradient of two successive blocks that has the zero crossing, the middle point of the successive blocks is taken as the reference point.
Here we have kept a threshold value for checking the number of minutiae matched. By changing the threshold value we get different rate of acceptance and rejection.

Steps used:

1. Classification, using SOM
2. Extraction of Reference Region using Gradient
3. Extraction of minutia and matching

Result:
Accuracy: 96%
FAR: 1%
FRR: 3%



(a) (b) (c)

(d) (e) (f)

Figure 9: Result of pre-processing steps.
(a) original image, (b) image after noise reduction and normalization, (c) region of interest,
(d) cropped image, (e) thinned image, (f) extracted minutia.

VIII. DISCUSSION

The reliability of any automated fingerprint based recognition system strongly relies on the precision obtained in the extraction process. Extraction of appropriate features is one of the most important and also difficult tasks for a recognition system.

There have been many algorithms developed for extraction of both local and global structures. Most algorithms found in the literature are somewhat difficult to implement and use a rather heuristic approach.

Here in our proposed method, we have used 8 x 8 pixel block for gradient calculation. And we have taken 68 x 68 pixel values around the reference point, for a database having larger fingerprint image sizes, the pixel values can be suitably changed.

For noisy database, we have seen that the 4 x 4 pixel block for gradient calculation is giving us a better result. It is also seen that this gradient approach is not suitable for all kinds of fingerprints, and further attributes are also required in order to accomplish the matching. In our future work, we are trying to solve this shortcoming.

In this paper, minutia extraction based fingerprint detection was applied with gradient detection as a step, to find the reference point.

Just like in this paper, SOM algorithm is used for classification, the singular point detection method can also be applied as a step to cluster the fingerprint images into major groups (i.e. arch, tented-arch, left loop, right loop, whorl), and then minutia extraction based method can be applied on the clusters to achieve a hierarchical fingerprint detection algorithm.

Clustering the fingerprint images in five major groups is quite easy if it is done manually by visual checking, but implementing an automated system for this is quite a hard job.

We have also applied position invariant features, for minutia detection. We are working on making the system more accurate.

REFERENCES

[1] H. C. Lee and R. E. Gaensslen, Eds., *Advances in Fingerprint Technology*. New York: Elsevier, 1991.

[2] R. Bahuguna, "Fingerprint verification using hologram matched filterings," presented at the 8th Meeting Biometric Consortium, San
Jose, CA, Jun. 1996.

[3] A. Ranade and A. Rosenfeld, "Point pattern matching by relaxation," *Pattern Recognit.*, vol. 12, no. 2, pp. 269–275, 1993.
[4] A. Jain, L. Hong and R. Bolle, "On-Line Fingerprint Verification", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 19, No.4, pp. 302-3 14, Apr. 1997.
[5] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 4, pp. 377–388, Apr. 1996.
[6] Image Digitization Process by Dorothy VanDeCarr
[7] M. Kass and A. Witkin, ªAnalyzing Oriented Patterns,º Computer Vision, Graphics, and Image Processing, vol. 37, no. 3, pp. 362-385, Mar. 1987.
[8] D. K. Isenor and S. G. Zaky, "Fingerprint identification using graph matching," *Pattern Recognit.*, vol. 19, no. 2, pp. 113–122, 1986.
[9] A. K. Jain, L. Hong, S. Pankanti, and R. Bolle, "An identity authentication system using fingerprints," *Proc. IEEE*, vol. 85, no. 9, pp. 1365–1388, Sep. 1997.
[10] Asker M. Bazen and Sabih H. Gerez, "Systematic Methods for the Computation of the Directional Fields and Singular Points of Fingerprints", *IEEE Trans. Pattern Anal. Mach. Intell*, vol. 24, no. 7, July 2002.
[11] A. Rao, *A Taxonomy for Texture Description and Identification*. New York: Springer-Verlag, 1990. 1110 IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 15, NO. 5, MAY 2006.
[12] C. Studholme, D. L. G. Hill, and D. J. Hawkes, "An overlap invariant entropy measure of 3D medical image alignment," *Pattern Recognit.*, vol. 32, no. 1, pp. 71–86, 1999.
[13] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain, ªFVC2000: Fingerprint Verification Competition,º Biolab internal
report, Univ. of Bologna, Italy, Sept. 2000, available from http:// bias.csr.unibo.it/fvc2000/.
[14] G.A. Drets and H.G. LiljenstroÈm, ªFingerprint Subclassification: A Neural Network Approach,º Intelligent Biometric Techniques in Fingerprint and Face Recognition, L.C. Jain, U. Halici, I. Hayashi, S.B. Lee, and S. Tsutsui, eds., pp. 109-134, Boca Raton, Fla.: CRC Press, 1999.
[15] C.L. Wilson, G.T. Candela, and C.I. Watson, ªNeural Network Fingerprint Classification,º J. Artificial Neural Networks, vol. 1, no. 2, pp. 203-228, 1994.
[16] L. O'Gorman and J.V. Nickerson, ªAn Approach to Fingerprint Filter Design,º Pattern Recognition, vol. 22, no. 1, pp. 29-38, 1989.
[17] A.R. Rao and R.C. Jain, ªComputerized Flow Field Analysis: Oriented Texture Fields,º IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 14, no. 7, pp. 693-709, July 1992.
[18] N. Ratha, S. Chen, and A. Jain, ªAdaptive Flow Orientation Based Feature Extraction in Fingerprint Images,º Pattern Recognition, vol. 28, pp. 1657-1672, Nov. 1995.
[19] P. Perona, ªOrientation Diffusions,º IEEE Trans. Image Processing, vol. 7, no. 3, pp. 457-467, Mar. 1998.
[20] T. Lindeberg, Scale-Space Theory in Computer Vision. Boston: Kluwer Academic Publishers, 1994.
[21] K. Karu and A.K. Jain, ªFingerprint Classification,º Pattern Recognition, vol. 29, no. 3, pp. 389-404, 1996.
[22] M. Kawagoe and A. Tojo, ªFingerprint Pattern Classification,º Pattern Recognition, vol. 17, no. 3, pp. 295-303, 1984.
[23] Rafeal C. Gonzalez, Richard E. Woods, "Digital Image Processing," Pearson Education Asia, 2002.
[24] A. K. Jain, S. Prabhakar and L. Hong, A Multichannel Approach to Fingerprint Classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21 (4), (April 1999) 348-358
[25] B. G. Sherlock and D. M. Monro, A Model for Interpreting Fingerprint Topology, *Pattern Recognition* 26 (7) (1993) 1047-1055.
[26] P. R. Vizcaya and L. A. Gerhardt, A Nonlinear Orientation Model for Global Description of Fingerprints, *Pattern Recognition* 29 (7) (1996) 1221-1231.
[27] N. K. Ratha, K. Karu, S. Chen and A. K. Jain, A Real Time Matching System for Large Fingerprint Databases, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18 (8) (August 1996) 800-813
[28] V. S. Srinivasan and N. N. Murthy, Detection of Singular Points In Fingerprint Images, *Pattern Recognition* 25 (2) (1992) 139-153
[29] A. P. Fitz and R. J. Green, Fingerprint Classification Using a Hexagonal Fast Fourier Transform, *Pattern Recognition* 29 (10) (1996) 1587-1597
[30] G. T. Candela, P. J. Grother, C. I. Watson, R. A. Wilkinson, C. L. Wilson, PCASYS : A Pattern-Level Classification Automation
[31] System for Fingerprints, *U.S. DEPARTMENT of COMMERCE Technology Administration, National Institute of Standards and Technology (NIST). Advanced Systems Division, Computer Systems Laboratory, Gaithersburg*, MD 20899, (August 1995)
[32] B. H. Choi et al., Core Based Fingerprint Image Classifcation, *ICPR'2000'S proceeding*, 2 (2000) 859-862
[33] S. Kasaei, M. R. Rahimi, E. Pakbaznia, A Multi Stage Approach to Fingerprint Classification, *CEE'2001'S proceeding*14.1-14.8.

