

# Genetic Algorithm For Fingerprint Matching

B. POORNA

*Department Of Computer Applications,  
Dr.M.G.R.Educational And Research Institute,  
Maduravoyal, Chennai 600095,TamilNadu  
INDIA.*

*Abstract:-* An efficient method for fingerprint identification is proposed. Fingerprint comparison is usually based on minutiae matching and the efficiency of the matching method is dependent on the extraction of the minutiae. In this method, a simple process has been adopted, wherein the pixel based information is used. The extraction of minutiae is based on the gray level value of pixels as well the neighbouring pixels. Also, this method is adopted for low quality fingerprints, as there is a relative change in the gray level. The computational complexity of the method is comparable with other methods.

*Keywords :-* Genetic Algorithm – Minutiae Extraction – Normalization – Gray Level - Encoding – Genetic operators -

## 1.INTRODUCTION

The problem of automatic minutiae extraction has been thoroughly studied earlier but never completely solved. Fingerprint classification and identification has been addressed by many researchers in the past. They are the most widely used biometric feature. Automatic fingerprint identification systems popularly known as AFIS is gaining importance as the fingerprint is unique to a person and because of its various applications such as access control, financial security and other kinds of personal identification methods.

The singular points, namely the core and delta points, act as registration for comparing the fingerprints. A topological approach to detect core point was proposed in [1]. A Fourier transform method to reach the core point was given in [3]. A method for the automatic detection of these points using syntactic tree grammar has been reported in [9]. These methods rely on the accuracy of the immediated neighbourhood information. The matching techniques of most of the fingerprint identification systems presume a high level of accuracy of singular points.

The method proposed in [5] primarily deals with the description of fingerprint impressions by determining the location of ridge endings, bifurcations and enclosures. Optical techniques in fingerprint classification has been attempted in [6]. Here, the major drawback is the noise and the requirement of numerous matching filters [7].

Some of the methods require ridge width, ridge length, ridge direction and minutiae direction to decide spurious minutiae. Most of the approaches use local ridge directions and a locally adaptive threshold method. To improve fingerprint image quality, directional ridge enhancement is used. The adaptive flow orientation based feature extraction method proposed in [8] involves tremendous execution time. One of the problems encountered by many existing systems for fingerprint matching is that they are highly sensitive to imperfections introduced during fingerprinting.

The method proposed in this paper determines the core points using the algorithm given in [4]. This method does not depend on a particular data set and it can be tested on the entire database. Smoothing and finding the singularities and classifying them are substantially faster. After

determining the core points, the intensity of the pixels around the core points are chosen for further processing. Generally, the intensity changes relatively. Due to this fact, our method determines the matching even if the image under consideration is of low quality.

## 2. MINUTIAE EXTRACTION

Minutiae are local discontinuities (ridge anomalies) in the fingerprint pattern. Fingerprint identification is mainly based on the detection of the minutiae. There are generally four types of minutiae [2]. They are *terminations, bifurcations, crossovers and undetermined*. Of these only the first two types of minutiae are considered for the identification. The efficiency of fingerprint identification system is dependent on the method used for the extraction of minutiae. Most of the minutiae extraction methods transform fingerprint images into binary images form through an adhoc algorithm. The images obtained are subject to a thinning algorithm.

The method proposed in this paper for fingerprint matching uses a simple process to detect the minutiae. The **core** point is the top most point of the inner most ridge and a **delta** point is the triradial point with three ridges radiating from it and are known as singular points. The two singular points of interest, are identified using already existing methods [4]. In order to do the fingerprint matching accurately the images are to be normalized [4]. The normalization must account for translation, rotation and scaling. The singular points are good candidates for registration points and for core classification of the fingerprint pattern. An arch fingerprint does not contain any singular point. **Tented arch** contains one core and one delta point, whereas **whorls** and **twin loops** contain two core and two delta points. By connecting the core and delta point it is possible to decide on the type of the fingerprint. This classification reduces the search space. The area of the fingerprints containing the singular points and the ridges is called a **pattern area** .(Fig. 1)

The method proposed here first reduces the ridge thickness. Then, a circular pattern area, with the registration point as the centre of the circle and a particular radius  $r$  is formed. Whenever the fingerprint classification has two core points as in the case of whorl and twin loop the mid-point of those two core points are considered to be the point of registration. In the case of an arch type fingerprint classification, as there are no singular points the following step is

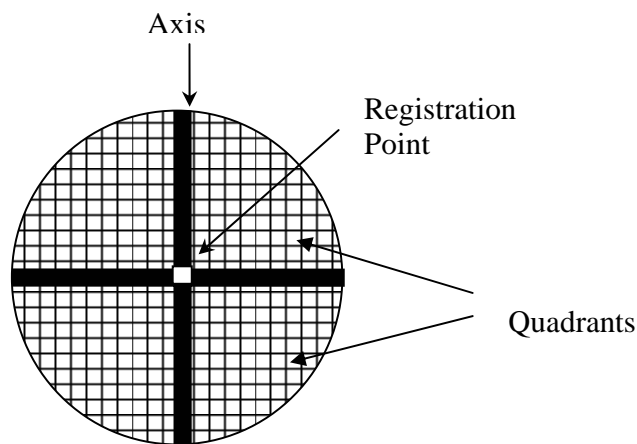


Fig. 1: Circular Pattern Area performed to determine the centre of the pattern area.

The arch type fingerprint has some ridges that are nearly horizontal at the bottom, above which arches are formed. The local maxima of each of the arches are considered, and if there are a number of consecutive points that satisfy the local maxima criteria, it determines a line and so the centre of this line is considered to be the local maxima of the arch shaped ridge. Similarly, the local maxima of other arch shaped ridges are to be determined. Now, the centroid of local maxima of these arches determines the registration point. Now, a circular pattern area is determined with radius  $r$  and the center as the registration point. The circular pattern area is divided into unit rectangular cells, each of one pixel wide and the value of  $r$  is so determined that the circular pattern area has atleast 512 pixels. The average gray value  $g$  of all the pixels in the pattern area is determined. The

cells are assigned a value zero if its intensity is below this value  $g$  and 1 otherwise.

The minutiae points are then determined as follows. Traverse the cells of circular area left to right within top to bottom order. For each cell that has a value of 1, the eight neighbouring cells that surround it are considered to form a rectangular area. Based on the gray value on these cells the type of the minutiae is decided. If the unit cell has a gray value 1, only in one of its neighbouring cells then that cell is a terminal minutiae. Similarly if the unit cell under consideration has only three neighbouring cells with gray value 1, then that point is considered to be a bifurcation minutiae [1]. Some example patterns are shown in Fig. 2.

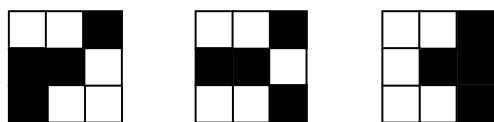


Fig. 2: Some Bifurcation Minutiae

Hence by traversing the given circular area, it is possible to determine the number and the type of minutiae. The choice of circular area is for recognizing fingerprints that appears in different directions. Moreover, the images to be matched should be at the same scale.

### 3.GENETIC ALGORITHM

Genetic algorithms are adaptive methods which may be used to solve search and optimization problems. A genetic algorithm is an iterative procedure which maintains a constant size population of candidate solutions. The initial population is chosen at random. A population is a set of chromosomes representing possible solutions. In order to search other points in the search space, some variation is introduced in the existing population by means of genetic recombination operators like **crossover** and **mutation**, and which generates new population. A genetic algorithm can be represented as a set of parameters  $(M,D,R,f,S,N)$  where  $M$  is the

encoding scheme which is a mapping from domain elements to coding values,  $D$  is the genetic operators applied for searching,  $R$  is the rates that controls the frequencies with which genetic operators are applied,  $F$  is the fitness function to evaluate the fitness value of chromosomes in population,  $S$  is the stop condition and  $N$  is the population size.

A genetic algorithm operates by maintaining a population of chromosomes. Chromosome population members are called individuals. Each individual is assigned a fitness value based on a problem specific evaluation function. The fitness function as well as crossover and mutation operators used in our approach is elaborated in the following sections.

### 3.1 Encoding

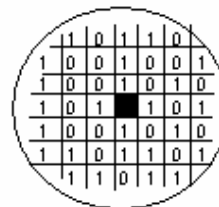


Fig. 3 :Encoding

Genetic algorithm basically deals with the concept of chromosome. Chromosomes represent an encoding of the information on which the algorithm operates. Encoding may be viewed in several forms, like binary encoding, real value encoding, permutation encoding etc. The type of encoding is purely depending on the problem to be solved. Here, we use binary encoding. The chromosome here is made of six parts. The length of each chromosome is  $n$  where  $n$  is the number of pixels that make up the circular area. The chromosome formed here is a binary string corresponding to the gray level value of cells exists in the parts. The first part is meant for the registration point, which consists of only one cell. Therefore, the gray level value of the registration point is considered for the first part. Basically, for left loop, right loop and whorl the bit value will be zero, and for tented arch the bit value will be one. However, in case of arch and twin loop it may be 0 or 1. These information leads to reduction of number of

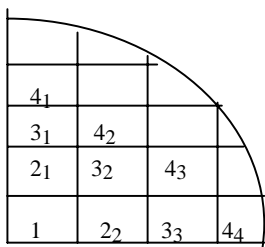


Fig. 4: Pixel ordering chord3

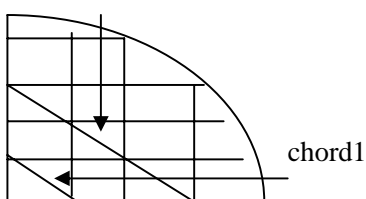


Fig. 5: Chords of the Quadrant chromosomes in the search space. The bit information for the next four parts are determined based on the gray level value exists in the clockwise order starting from the registration point towards the boundary of the circle corresponding to the four quarters. The sixth part consists of the gray pixel value of the axis, starting from the positive y-axis and then in clockwise order. For each axis, cells are to be considered from registration point towards the boundary of the circle. Fig. 4 and Fig. 5 illustrates the order of choosing cells for assigning bit values for each quarter, and that order be  $1, 2_1, 2_2, 3_1, 3_2, \dots$ . In our approach,  $i^{th}, 1 \leq i \leq 5$ , part corresponds to the  $i-1^{th}$  quarter, after normalization, which is explained in Section 3.3. Thus, the entire chromosome is made up of sequence of  $0_s$  and  $1_s$ .

### 3.2 Genetic operators

The genetic operators create new search chromosomes by altering the composition of parent chromosomes. The most basic forms of these operators are selection, crossover and mutation. The selection operator forms a new population by selecting chromosomes from old population based on their fitness value. The fitness value is found as follows. The circular pattern area already stored in the database that belong to the same

classification (arch, tented arch, left loop, right loop, whorl or twin loop) as that of the pattern to be matched are considered. The number and type of the minutiae of each quarter of the pattern to be recognised is matched with the patterns that belong to the same class. All the patterns that match so are considered and the weight of each pattern is found. The average weight  $W$  of these patterns is determined. The fitness value  $f_i$  is decided as  $W - \epsilon \leq f_i \leq W + \epsilon$ , where  $\epsilon$  is the allowable error. Given a circular pattern area with  $n$  cells, a maximum of  $2^n$  chromosomes are possible. All those chromosomes that satisfies the selection condition is chosen for processing. The fitness function gives each chromosome a fitness value which is a judgement of its surviving capability.

Crossover selects genes from parent chromosomes and creates a new offspring. Depending on the number of cells with gray level value  $1$ , each quarter is found to have a particular weight. The sum of the weights of all the five parts is considered to be the fitness value of each chromosome. Four crossover points are chosen, one after every quarter information in the chromosome. Two parent chromosomes are selected and are subject to crossover to form the child chromosome. The weight of each quarter is also available, as it is determined for normalization. The average weight of each quarter in that generation is determined. Any quarter that is having a weight value, which is much above this average weight is subject to crossover with a chromosome whose weight value in that quarter is much below the average weight. After the crossover operation the fitness value of each chromosome is recalculated. Mutation changes randomly the new offspring, by choosing bits at random and changing a 1 to 0 and a 0 to 1. The overall fitness value of each of the chromosome may satisfy the above condition, but the weights of corresponding quarters of the chromosome may vary by more than  $\epsilon$  from the average weight of the quarters. It is likely that atleast two parts of that particular chromosome may contribute to that

individual being fit. But, the quarters when considered individually may be such that one has a weight that is more and the other has a weight that is less than the average weight of that quarter. Those two parts are subject to mutation. For the quarters that has less weight more number of bits with a zero value are made a 1 and vice versa depending on the mutation rate. After the mutation operation again the fitness value of the chromosome is computed. Thus at the end of this generation all those chromosomes with fitness value as stated above is chosen to form the population for subsequent processing. Both crossover and mutation have an occurrence rate that controls the frequency with which the operator is applied. The higher the crossover rate, the more quickly new chromosome will be introduced in the population. The higher the mutation rate the algorithm would become similar to random search. A new population results from these operations. Over the course of a number of generations, the average fitness of the population increases, and the fittest individuals approach acceptable solutions to the application problem.

**3.3 Normalization**

Let us assume that there are  $n$  cells in the circular area. Assign a weight  $w_c$  to the core point. Depending upon the type of the fingerprint  $w_c$  will be 1, 2, 3, 4, 5 or 6 respectively for arch, tented arch, left loop, right loop, whorl or twin loop respectively. The circular area is considered to have 16 arcs with the first arc made of one pixel, the second has two pixels and so on with the sixteenth arc having 16 pixels. The arc that is nearest to the core point is assigned a weight  $w_1 = 0.95$ , followed by the next successive arcs being assigned the weight starting from 0.9 and reducing the weight by 0.5 and the value of  $w_{16}$  is 0.25. The values are chosen at random between 0 and 1. The circular pattern area is divided into four quarters along the line of pixels with respect to the core point as shown in Fig. 6. The weight of each quarter is determined as follows.

Let  $S_{ij}$  be the sum of the weight of each concentric circle starting from the one that is nearest to the core point of quadrant  $i$ , where  $S_{ij} = \sum_{j=1}^m \sum_{k=1}^r w_j$ , and  $r_j$  is the number of gray pixels in the circle  $j$  and  $1 \leq i \leq 4$  and  $1 \leq j \leq m$ .

Let us denote  $S_{ij}$  as  $q_i$ . Form a sequence with these  $q_i$ 's. Shift the values of  $q_i$ 's till you get a minimum sequence. For example if  $q_1 = 10, q_2 = 5, q_3 = 12$  and  $q_4 = 9$ , then the initial sequence would be (10)(5)(12)(9). This sequence gives (9)(10)(5)(12) after one right shift. The given sequence is shifted to the right till a minimum sequence is obtained.

In this case it is (5)(12)(9)(10) which is obtained after four right shifts. Hence  $q_3$  is considered to be the first quadrant, followed

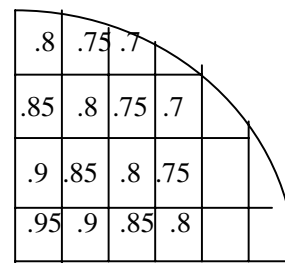
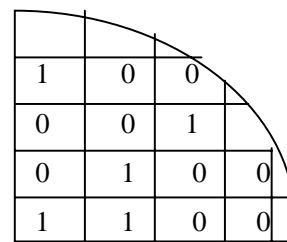


Fig. 6: Weight determination

by the other three quadrants in the clockwise order. As there are only four quadrants the minimum sequence can be obtained within four right shifts. Whenever chromosomes are to be formed they are first subject to normalization and only then the chromosomes are subject to the genetic operation. Each core point is assigned a different weight and it is the maximum of all the assigned weights. When the core points of two different finger points are not the same then the difference in weight is more and hence they cannot be similar. The

iterative procedure has to stop and is a critical consideration. The algorithm improves the search process by eliminating the chromosomes that do not adjust to the environment among the population and by creating new chromosomes by the genetic operators. To have good performance the algorithm is stopped after a limited time or after performing the iterations a particular number of time.

#### 4. Conclusion

A genetic algorithm based method for matching the fingerprints have been proposed. It has the advantage it works well for fingerprints with two different intensities. As the minutiae extraction is dependent only on the gray value of a particular unit cell and its eight neighbouring unit cells, it is still possible to find the minutiae in the case of low quality fingerprints. Also this method works well under translation, rotation and scaling. Moreover this method requires a simple data structure

#### REFERENCES

[1] Anil Jain and Lin Hong and Ruud Bolle, 1997, "On-Line Fingerprint Verification", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19( 4), pp..302-314.  
[ 2 ] Alessandro Farina and Zsolt M. Kovacs-Vajna and Alberto Leone , June 1999, "Fingerprint minutiae extraction from skeletonized binary images", *Pattern Recognition*, 32, pp. 877-889.

[3] D.A. Aushermann and others. 1973, "A proposed method for the analysis of dematoglyphics patterns", *Proc. Soc.Photoopt.Instrum Engrs*, 40.  
[4] Kalle Karu and Anil~K. Jain. July 1995, "Fingerprint classification", *Pattern Recognition*, 29(3), pp .389—404.  
[5] C.R. Kingston., 1967, "Problems in semi-automated fingerprint classification", *Law Enforcement Science and Technology, Academic Press.*  
[6] "E. Marom",1967, Fingerprint Classification and Identification Using Optical Methods.", *Law Enforcement Science and Technology, Academic Press*, pp.481-484.  
[7] B.~Moayer and K.S. Fu.,February 1975, "A syntactic approach to fingerprint pattern recognition" *Pattern Recognition*, 7, pp.1--23.  
[8] Nalini K. Ratha and Shaoyun Chen and Anil K. Jain, 1965, "Adaptive Flow Orientation-Based Feature Extraction in Fingerprint Images", *Pattern Recognition*, 28(11), pp.1657 – 1672.,  
[9] C.V.K. Rao and K.Black.,1980,"A syntactic approach to classification of fingerprints".*IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2, pp.:223--231.  
[ 10 ] J.T. Tou and W.J.Hankley, 1968, "Automatic Fingerprint Identification and Classification analysis via contextual analysis and topological coding", *Pictorial Pattern Recognition*, pp.411-456.