Sindh Univ. Res. Jour. (Sci.Ser.), Vol: 34-B (2) 35 - 46 (2002)

AUTOMATED FINGERPRINT IDENTIFICATION SYSTEM

Usman Ali Bukhari, Umer Irshad Khan, Nasir Mahmood, Hafiz Muhammad Aslam* and Noor M. Shaikh*

Department of Computer Sciences Lahore, University of Management Sciences, Lahore, Pakistan

Abstract

In this paper we present selected stages of an Automated Fingerprint Identification System. This software for the system is developed employing algorithms for two-tone conversion, thinning, feature extraction, and matching. Keeping FBI standards into account, it has been assured that no details of the image are lost in the process. We have deployed a general parallel thinning algorithm for specialied images like Fingerprints, and modified the original algorithm after a series of experimentation selecting the one giving the best results. We also propose an application-based approach for designing Automated Fingerprint Identification Systems keeping in view systems requirements. We will show that by using our system, the precision and efficiency of current fingerprint matching techniques are increased.

Keywords: Fingerprint Identification.

1. Introduction

Everyone is known to have unique and immutable fingerprints. A fingerprint is made of a series of ridges (lines) and furrows on the surface of the finger. The uniqueness of a fingerprint can be determined by the pattern of ridges and furrows as well as the minutia points. Minutia points are local ridge characteristics that occur at either a ridge bifurcation or a ridge ending. Captured fingerprint images are processed througha series of image processing algorithms to obtain a clear and unambiguous skeletal (thinned) image of the original gray tone impression.

All the coordinates of ridge ends (breakpoints) within the image are identified and saved. The process of fingerprint matching then compares the data from the input print against all records in the database to determine if a probable match exists.

1.1 Standard

The matching of Fingerprints is carried out either on the basis of breakpoints (ridge endings), bifurcation, short ridges, enclosures, trifurcation etc. that are the distinguishing features in any fingerprint image. A reliable system consists of matching based on at least two criteria, normally ridge endings and bifurcations. FBI (USA) has presently the largest database of fingerprint images in the world that comprises of more than 70 million images [1], and the matching criteria used by FBI is based on both ridge endings as well as bifurcations. FBI's criminal information Services Division in collaboration with National Institute of Standards & Technology has defined standards for fingerprint digitization and compression in 1933, 2,3,4, which are as follows:

Digitization Standard	
Resolution	500 dpi
Pixel Depth	8(8bit bmp)
Gray Levels per Pixel	256

Figure 1.1 a
FBI DigitizationStandards

* Department of Electrical Engineering University of Engineering & Technology, Lahore, Pakistan

CamScanner

1.2 Search Delimiters

In authomated systems, part of the process of narrowing the search involves decisions made by the operator during the process of entering the subject finger image. The subject's sex, age, height and pattern type etc. are often used to limit the amount of the database that must be searched. It should be noted that neither of these parameters is necessary for accurate matching. Rather, they are used to limit the required processing power of the system and therefore its size and cost. This is significant, because when properly used these parameters (known as search delimiters) typically restrict the search to less than 20 percent of the database. Compared to the alternative of searching the complete database every time, the effect on required processing power is obvious.

2. System design consideration

The algorithms that we are using in our different modules of the system are described here under the headings of the respective stages in the processing.

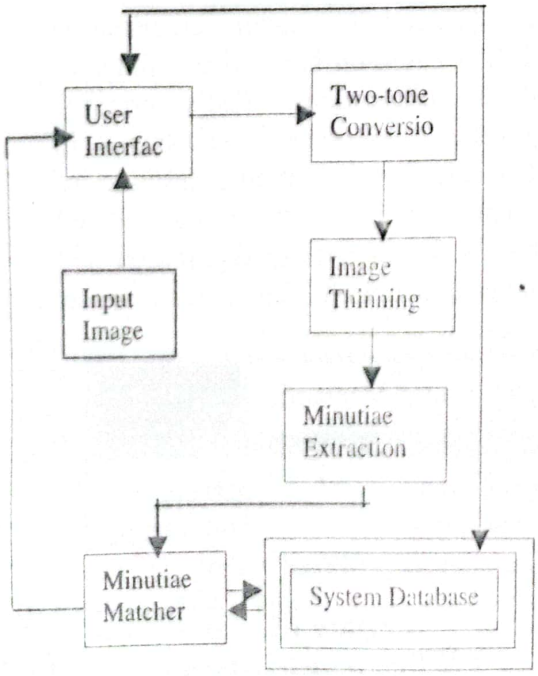


Figure 2. a
Block Diagram of Proposed AFIS System

2.1 Pre-processing

In order to ensure that the performance of an automatic fingerprint identification/verification system will be robust with respect to the quality of the fingerprint images, it is essential to incorporate a fingerprint enhancement or pre-processing algorithm before the minutiae extraction module. We have developed a fast pre-processing algorithm, which can adaptively improve the clarity of ridge and furrow structures of input fingerprint images by firstly converting the image into two tone image and then applying a thinning process to accurately reduce ridge thickness to one pixel thin.

This thinned image is then used by the feature extraction algorithm that computes the total number of Ridge endings as the charactertistic feature and records the relative coordinates of these ridge endings in the database. This extracted information is then compared with the candidate fingerprint image during the Matching process. Our algorithm makes a fast and accurate feature extraction possible in real time.

2.1.1 Fingerprint Smoothing

There are two purposes of smoothing: First is to reduce excessive or artificial contrast and second to clean the noise. Artificial contrast and false contours may appear which in turn may mask important information of the image. The simplest smoothing algorithm is mean filtering that is explained here.

2.1.2 Mean Filtering

The main idea behind mean filtering is to apply some sort of averaging to the image. Consider a neighborhood S of 3 * 3 pixels around and including the candidate pixel (x,y). Then the smoothed image at (x,y) is given by

CamScanner

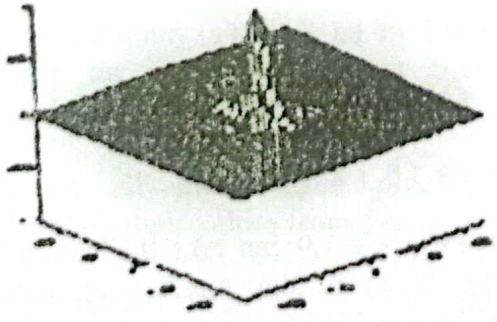
$F(x,y) = 1/9 \sum_{(x,y) \in S} f(x,y)$
The computation for mean filtering can be made more efficient using two stages

$F_1(a,y) = \sum_{x \in S1} f(x,y)$
$F(x,y) = 1/9 \sum_{y \in S2} f_1(x,y)$

where S1 is the three pixel row neighborhood and S2 is the three pixel column neighborhood in f and f_1 respectively. When computing in a row and column scanning fashion, the average number of additions in these relations are only two for any size of the neighborhood.

2.1.3 Filtering of Fingerprint Image

Fingerprints are flow-like patterns, which consist of locally parallel ridges and valleys. They have well defined local frequency and local orientation. A set of bandpass filters can efficiently remove the undesired noise and preserve the true ridge/valley structures. Gabor filters have both frequency-selective and orientation-selective properties and have optimal joint resolution in both spatial and frequency domains. Therefore, it is beneficial to use Gabor filters as bandpass filters to remove the noise and preserve true ridge/valley structures.



(a)

Figure 2.1.3.a Even Symmetric Gabor Filter Tuned to 60 cycles / width and zero degree orientation.

To obtain a filtered image, a FFT is first performed on the input fingerprint image. Then the corresponding gabor filter with tuned radial and orientation frequency is applied to the freqency image and an inverse FFT is performed to obtain the filtered image.

2.2 Gray-Scale to Two Tone Conversion

A Constraint Of matching algorithm is that the candidate image should be abinary or two - tone image. Thus the input gray scale image is converted in black and white by computing each part with a predefined grayscale threshold value. If the pixel value is less than the threshold value it is converted to 0 (i.e. White) and if the pixel value is greater than threshold it is con verted to I (i.e. Black). As a result all the background and valleys in the image are converted into white and all the ridges are converted into black in the resulting binary image.



Figure 2.2 a
Original Image



Figure 2.2.b
Two-Tone Image

2.2.1 Algorithm for Two-Tone Conversion

1. Input file verification
2. Header reading
3. Verification of file as a 256 color bitmap
4. Making header of monochrome bitmap by seetting

CamScanner

i. bytes offset = 62

ii. bits / pixel = 1

iii. color indexes = 0

5. Pallete of monochrome image {0,0,0,0,255,255,255,0}

6. Determining of no scan lines from image size

7. Loop for each scan line converting each line to monochrome

2.3 Thinning

Flawless matching and feature extraction of fingerprint image requires thinning of input image. Thinning process comprises deleting the border layer pixel of the image alternatively keeping in view whether the operation preserves the connectivity property.

The process is similar to peeling the successive layers of onion. While deleting, it is tested whether the operation preserves the connectivity poperty of the image. A Typical algorithm due to Zhang and Suen (1984) [5], is used for thinning by us. It is as follows:





Figure 2.3.a Original Two-Tone Image **Figure 2.3.b Thinned Image**

2.3.1 Zhang & Suen Algorithm

The input image required by this algorithm need to be two-tone (i.e, black & white) image. So the candidate image that has been converted to two-tone image in the preceding stage is then given to this algorithm as input. Note that in this image, ridges are black and background with valleys is white.

Let the object and background pixels be denoted by values 1 and 0, respectively. Consider the configuration of figure 2.3.1.a, where f 1 is the pixel under consideration for deletion.

f 9	f 2	f 3
f 8	f 1	f 4
f 7	f 6	f 5

Figure 2.3.1. a
Configuration for thinning algorithm

A two-stage deletion process is executed by the algorithm. In the first stage, the following four conditions are tested on all border pixels of the image.

1. $2 \leq = N\,1\,(f\,1) \leq = 6$

2. $N\,2\,(f\,1) = 1$

3. $f\,1 . f\,4 . f\,6 = 0$

4. $f\,4 . f\,6 . f\,8 = 0$

where f_i; i = 1,9 can take the values 0 or 1.NI (f 1) is the number of nonzero neighbors of f1 i.e.,

$$NI\,(f_i) = \Sigma^{9}_{i=2}\, f_i$$

And N2 (f 1) is the number of 0-1 transitions in the ordered sequence of f2,f3....f 9.

For example, N1 (f) = 4 and N2 (f) = 2 for the following figure 2.3.1.b,

0	0	1
0	f 1	1
0	1	0

Figure 2.3.1.b
A typical neighborhood of two-tone image.

In the second stage, conditions (1) and (2) remain the same but condition (3) and (4) are changed to

3. f 2. f 4. f 8 = 0

4. f 2. f 6. f 8 = 0

If at any stage, all the four conditions are satisfied for a candidate pixel, then the pixel is flagged for delection. If one or more conditions are violated, the candidate pixel must not be deleted. Thus algorithm is as follows:

1. Use condition (1) – (4) of the first stage on all border pixels and flag those satisfying the conditions.

2. Delete all the flagged pixels by changing their values from 1 to 0.

3. Use condition (1) – (4) of the second stage on all the remaining border pixels and flag them accordingly.

4. Delete all the flagged pixels by changing their values from 1 to 0.

5. If no pixel is flagged at step 2 or step 4 STOP.Else, go to step 1.

It may be noted that a pixel is not deleted until all border pixels are flagged. this prevents changing the structure of the border during execution. Also, it is suitable for parallel implementation.

On examining the conditions, it may be noted that condition (1) is violated when the candidate f 1 has only one or seven neighbors with value 1. Having one neighbor with value 1 means that f 1 is at the end of a skeleton and hence should not be deleted. If f 1 has seven neighbors with value 1 then it's deletion would cause erosion into the region. Condition (2) is the connectivity preserving condition. It is violated when the border is one pixel thick at f 1. Conditions (3) – (4) in the first stage are simultaneously satisfied by the minimum set of values f 4 = 0 or f 6 = 0 or (f 2 =0 and f 8 = 0). A candidate for which these as well as conditions (1) – (2) are satisfied, is either an east or south border pixel or a northwest corner pixel that can be removed. Similarly, these conditions being satisfied in second stage ensures removability of north or west border as well as southeast pixels. Removability of northeast and southeast corner pixels are ensured in conditions (3) – (4) of both stages.

2.3.2 Lu and Wang Modification

In addition to preserving the connectivity property, a good thinning algorithm should be less sensitive to the border noise as well as preserve the elongatedness of the original object. It has been pointed out by Lu and Wang (1986) [6], that for some typical figures, the above algorithm does not yield satisfactory results. They noted that the problems can be partly solved if condition (1) is changed to

$3 < N 1 (f 1) <= 6$

2.3.3 Specialized Thinning Algorithm for Fingerprint Images

During our process of work we found the above mentioned Zhang and Suen

CamScanner

algorithm does not work fine for our case i.e. specialized images like Fingerprints. The resulting thinned images are undully disrupted at some places and for some images the output is totally useless. After through research and experimentation we found that if we increase the number of 0 – 1 tasitions of the original hang and Suen algorithm in the ordered sequence of f 2, f 3,....f 9 and then again from f 9 to f 2, the results are almost desirable. The difference of the original and modified Zhang & Suen thinning algorithm is shown by experimented images in figure 2.3.3.



Figure 2.3.3.a
Thinning results of original Zang and Suen Algorithm.



Figure 2.3.3.b
Thinning results of modified Zang & Suen Algorithm.

2.3.4 A Brief Review of other Thinning Algorithms

There exist a number of techniques for thinning objects with definite elongated shape. The basic idea in these techniques is to compute the local thickness of the object at intervals and find the midline by joining successive midpoints at these thicknesses. The local thickness of an object may be found roughly as the intercept of the horiontal (vertical) scan on the object. The approach is fast and it can be executed in a small active memory space containing only a few lines of the image. But, it may fail to work successfully on images that are elongated along the scanning direction rather than across it (Pavlidis, 1982) [7,8]. It may be modified so that scanning along both horizontal and vertical directions are considered and shorter intercept is used for the thickness.

An alternative but somewhat slow aproach is to follow the border to find the thickness (Chaudhuri, 1978) [9]. The local thickness may be found by the intercepts of the normal at the point under consideration. The points of high curvature on the border are avoided because the normals at those points do not properly represent the thickness. Another similar algorithm is reported in Jimenez and Navalon (1982) [10]. Some distance-based modified algorithms are also proposed in Arcelli & Di Baja (1985), [11,12] that ensure connectivity property of a thinned skeleton image.

2.4 Feature Extraction

In forensic fingerprint identification, a trained fingerprint examiner looks for identical minutiae in identical locations on each fingerprint. These are known as corresponding minutiae. If both prints are clear and well defined, the examiner can easily find dozens of corresponding minutiae; but when a fingerprint is blurred,

smeared, faint, or if only a partial fingerprint is available, only a dozen or so corresponding minutiae may be discernible. The latter is often the case when a fingerprint image (a latent print) has been retrieved from an object or surface, rather than from an inked fingerprint card or live-scan imaging device.

Automated identification systems make the same types of comparisons using minutiae records, which are derived by computer ananlysis of a fingerprint that has been scanned electronically and converted to digital form. The minutiae record is an electronic "map" showing the location and orientation of the minutiae on the original print. it is separated from the image record, which is simply a digitzied electronic " photograph" of the fingerprint.

2.4.1 Minutiae based matching technique

The extraction of the characteristic features of a fingerprint image and mapping its relative coordinates is the main step of minutiae-based technique. These characteristic features can be ridge endings, bifurcation, delta points, enclosures, or any special features etc, depending upon the requirements of that particular technique and its usage. These features are extracted relative to a reference point and then they are compared with those kept in a database to match the candidate fingerprint.

The selection of a perfect matching scheme is the core issue of the process. In minutiae-based technique fingerprints are matched on the basis of certain characteristics that are unique to every fingerprint. These are

- Ridge Endings or Break points
- Bifurcations
- Delta Points
- Enclosures

All these are distinct features of a fingerprint & are never duplicated in nature. Any one of the above mentioned qualities of the fingerprint could be chosen as a matching criterion. But selection of best possible matching criteria depends upon the system requirements and specifications Breakpoints - based fingerprint matching is the most popular technique as they are abundant, vital structures and require comparatively simple procedures to be detected.

To match the fingerprint, the exact location and number of breakpoints in the image are recorded precisely and these are compared with the record of another fingerprint image to find a possible match. The same fingerprint should have the same locations of breakpoints and would have thus resulted in a perfect match. The uniqueness of the breakpoints for every person ensures that samples of different fingerprints have different coordinates of breakpoints and when they are compared, a mismatch would occur. This is the underlying scheme of such system in a nutshell.

2.4.2 Feature Extraction Algorithm

First of all the algorithm is applied on each ON pixel (black i.e. ridge part of thinned image) to check for a breakpoint and all the OFF pixels (white i.e. valley part of the thinned image) are left as they are.

Consider the figure 2.4.2.a shown below. It contains the ON pixel (black i.e. ridge or zero) under consideration named as F1 in the figure. Now algorithm is applied on it that check F2, F3, F4 , F5, F6, F7, F8, F9 to find whether any other pixel in it's neighborhood is black or not. if there is only one balck pixel in its neighborhood then it is a break point (ridge ending), otherwise it is simply a continuation of a line (ridge).
In case of a breakpoint the coordinates of that point (F1), in terms of rows and

CamScanner

columns, are saved. And in case if it is not a breakpoint, then the control is simply transferred to next pixel (F4), and now algorithm will be applied on F4 (only if it is also an ON pexel). And the process continues..

F 9	F 2	F 3
F 8	F 1	F 4
F 7	F 6	F 5

Figure 2.4.2.a
Configuration of Feature Extraction Algorithm

F 9	F 2	F 3
F 8	F 1	F 4
F 7	F 6	F 5

Figure 2.4.2.b
Showing the transfer of control to next pixel in second step of Feature Extraction Algorithm

2.5 Matching Process

After executing the code based on above-mentioned algorithm, the result is found in the form of number of breakpoints and their respective coordinates. These breakpoint and their coordinates are then stored in database as the extracted features of a particular fingerprint image.

During the process of matching, the input candidate image is processed upon by a series of algorithms, finally extracting the breakpoints and their respective coordinates called the " image map". Then only these extracted features of a paticular image (candidate image) are matched with all those already stored in the database. It makes a speedy search possible that is the main requirement of a civil AFIS system.

The procedure for matching the fingerprint image in a database works in three steps:

1. First of all candidate image (image to be matched with all the images of the database) is processed and it's characteristic features are extracted (breakpoints or ridge endings and their coordinates).

2. Then all the images in the database are checked against the found number of breakpoints of the candidate image.

3. If an image with the same number of breakpoints is found then it's respective coordinates, already stored in the databse are compared against those found for the candidate image.

4. In case of a match found a form or report of that matched person's information is displayed.

3. Summary

Fingerprint identification technology has been a mainstay of forensic (criminal) investigators since its development in the 19th century. With the advent of automated fingerprint identification systems (AFIS), fingerprinting has become even more effective as a method of establishing positive personal identification.

In this paper we present selected processes involved in Automated Fingerprint Identification System. The two main uses of such a system are Civil and Forensic applications. Although civil and forenisic fingerprint matching share the same underlying technology, they are used in very different environments and for different operational goals. For this reason, we propose that the technical requirements used in the design and/ or procurement of a forensic AFIS cannot simply be transferred to the civil arena.

We propose an application specific system design. This includes restriction of live

CamScanner

input image intake for Civil AFIS systems (Driving License, Passports, ID cards and Exit Control lists etc.) where fastest possible matching is the key reqirement, without any trade-off on result authenticity. tHe other restriction that is incorporated is processing every databse image each time it is to be matched, rather we implemented image map concept of keeping and processing only the extracted features of any image in the image databse. This also enhanced our system speed and helped improve the overall quality & efficiency of the system.

We proposed that for criminal database and latent fingerprint image handiling and matching, pattern based mechanism should be used with proper orientation estimation, pre-processing and live scan input image intake should be used for best possible results.

For further reading on Latest research in Biometrics visit the website of MSU, 13. A postscript version of this paper is available via anonymous ftp to uet. Edu.pk by December 1, 2000.

Acknowledgement

This work was partially sported by Fingerprint Bureau of Pakistan. Forensic Science Labs., and Department of Plice Punjab. Address all correspondence to D.I.G (Crimes), Khalid Farooq, Reference No. 11825/crimes/AC-1.

References

[1] FBI Standards-IAFIS Incremental Builds (http://www.fbi.gov/programs/iafis/iafis.htm)

[2] FBI Forensic Laboratory, http://www.fbi.gov/programs/lab/labhome.htm

[3] Handbook of Forensic Services-Latent Fingerprints Techniques, http://www.fbi.gov/programs/lab/handbook/examplatp.htm

[4] Forensic Science Communications. October 200, Vol.2,No.4. http://www.fbi.gov/programs/lab/fsc/current/index.htm

[5] Zhang, T.Y. and Suen, C.Y. (1984). A fast parallel algorithm for thinning digital patterns, Commun. ACM, Vol.27 236-239.

[6] Lu, H.E. and Wang, P.S.P. (1986). A comment on " A fast parallel algorithm for thinning digital patterns", Comm. ACM., Vol. 29 239-242.

[7] Pavlidis, T, (1980). A thinning algorithm for Discrete Binary Images, Comput. Graphics Image Process., Vol. 13, 142-152.

[8] Pavlidis, T. (1982 b). An asynchronous thinning algorith, Comput. Graphics Image Process., Vol. 20, 133-157.

[9] Chaudhuri, B.B. (1978), A simple method of thinning, J. Inst. Elect Telecom. Engg. Vol. 14, 264-265.

[10] Jimenez, J. And Navalon, J.L. (1982). Some experiments in image vectorization, IBM J. Res, Develop., Vol. 26 724-734.

[11] Arcelli, C. And Sanniti di Baja, g. (1985). A width independent fast thinning algorith, IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI-7, 463-474.

[12] Arcelli, C. (1981). Pattern thinning by contour tracing, Comput. Graphics Image Process., Vol. 13, 130-144.

[13] Michigan State University Biometrics. Department Research Publications. (http:// biometrics. cse.msu. edu/ publications. htm1).

CamScanner



Figure 3.3.3.a
Modified Zhang & Suen Algorithm Result



Figure 3.3.3.a
Original Zhang & Suen Algorithm Result



Figure 3.3.3.b
Modified Zhang & Suen Algorithm Result



Figure 3.3.3.b
Original Zhang & Suen Algorithm Result



Figure 3.3.3.c
Modified Zhang & Suen Algorithm Result



Figure 3.3.3.c
Original Zhang & Suen Algorithm Result

Figure 3.3.3
Comparison of thinning results on different fingerprint images
using original and modified Zhang and Suen Algorithm

CamScanner



Figure 4.a
Original Gray Scale Image



Figure 4.b
Two-Tone Image



Figure 4.c
Thinned Image

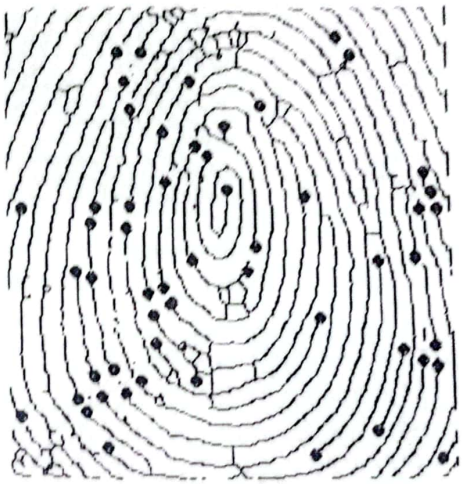


Figure 4.d
Feature Extracted Image

Figure 4
Stages of Fingerprint Image processing

CamScanner



Figure 5.a
Original Grayscale Image



Figure 5.b
Original Grayscale Image



Figure 5.a
Two-Tone Image



Figure 5.b
Two-Tone Image



Figure5.a
Thinned Image



Figure5.b
Thinned Image

Figure 5
Stages of fingerprint image processing on two different patterns of fingerprints.

CamScanner