

## AN EFFICIENT FINGERPRINT MATCHING SYSTEM

*Abu Ismail, Uwe Schnabel*

Biometrics Department. Omnikey AG  
In weiden 4b, D-99099 Erfurt, Germany  
E-mail: {abu.ismail , uwe.schnabel}@omnikey.com

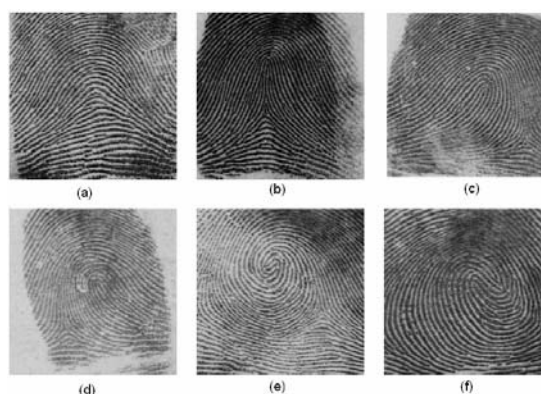
### ABSTRACT

A novel biometric authentication, based on fingerprint matching is presented in this paper. This improved approach adaptively traces the type of minutia with accordance to the minutia points unlike the conventional fingerprint matching techniques. A two in one matching algorithm, matches both minutia location and type succeeded by fingerprint group matching, has been proposed for the authenticity check between the template and the skeleton image. An easy and fast minutia feature extraction followed by false minutiae elimination has been incorporated in the system.

### 1. INTRODUCTION

Proper authentication becomes a very crucial issue of these days. Traditional knowledge-based (password or PIN) and token-based (passport, driver license, and ID card) identifications are prone to fraud. That's why identification through physical or behavioural characteristics so-called Biometric is being increasingly adopted with a high degree of confidence. Fingerprint is one of the unique identifications of the personnel, where the identity is checked through the fingerprint feature against an enrolled template. The informative template proposed in this paper contains the relative location of the minutia points and the type of the minutia as well as the pre-defined fingerprint group [2] (arch, tented arch, right loop, left loop, whorl, twin loop, or unknown) belongs to the template. The finger print group match speeds up the complete process confining the search in a subgroup as in the case of the most fingerprint matching systems dedicated to large volume of database. The involvement of two minutia features strengthens matching efficiency and reliability. In order to facilitate the matching, the skeleton image is converted to distance [1] based

point cloud image, and template matching is performed against this point cloud image.



**Fig. 1** Different type of fingerprint image:  
(a): arch, (b) tented arch, (c) right loop, (d) left loop, (e) whorl, (f) twin loop

This paper is organized as follows: chapter 2 summarizes the fingerprint image pre-processing steps, Chapter 3 describes the way of skeleton image derivation and the matching algorithm is the content of the chapter 4. The results and conclusion are in the successive chapters.

### 2. IMAGE PRE-PROCESSING

The matching performance mostly depends on the pre-processing of the fingerprint image due to the vulnerability of ridge and valley to the sensors physics and environment. Any deficiency in the pre-processing system could increase the false rejection (FRR) and false acceptance rate (FAR); on the other hand the efficient pre-processing is also a time issue for the real time application. Pre-processing steps, involved in the system are shown in the block diagram of figure 2. There are a lot of methods for these processing of images, already established in the literature [4] [7]. For a better thinned binary image, in the proposed system, image enhancement

has been done using histogram equalization and a computational approach proposed by Canny [8] has been implemented for edge detection.

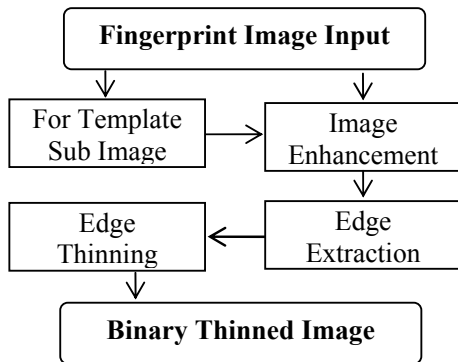


Fig. 2 Image pre-processing steps

A processed thinned binary image is shown in figure 3 with the marking of some minutia points.



Fig. 3 Thinned Binary image  $I(x, y)$

### 3. SKELETON IMAGE DERIVATION

From the binary image using a directional mapping [2] the fingerprint group is determined. For each group an eight-byte value is assigned e.g. for unknown  $g = \{00, 00, 00, 00, 00, 00, 00, 00\}$ , for arch  $g = \{00, 00, 00, 00, 00, 00, 00, 01\}$ , and so on. A less complex technique proposed in [3] is employed in the system for the feature extraction. A  $3 \times 3$  clockwise mask  $M(x, y)$  is used for determining the minutiae type and points.

x1	x2	x3
x8	M	x4
x7	x6	x5

Fig. 4 Minutiae Detection Mask

Thinned binary image  $I(x, y)$  and mask  $M(x, y)$  are taken through a mathematical operation as in equation

(1), only for the black pixels in  $I(x, y)$ , results the basis image  $B(x, y)$ .

$$B(x, y) = \sum_{k=1}^8 |M(k+1) \cdot I(k+1) - M(k)I(k)| \dots \dots \dots (1)$$

where  $X_n = 1$  ( $n = 1..9$ );

If  $B(x, y) = 3$ , the pixel is an end point, If  $B(x, y) = 7$ , the pixel is a bifurcation and so on.

In order to increase the matching efficiency the basis image is filtered with a so called orientation map [4] to eliminate the false minutiae. The false minutia free basis image pixel values are mapped according to predefined minutiae index (Mnt) defining the minutia types which is considered as the skeleton image  $S(x, y)$ . The Mnt is a user defined variable calculated as follows:

1. Ridge bifurcation # Mnt = 11p;
2. Ridge end # Mnt = 9p ;
3. Bridge # Mnt = 7p;
4. Spike # Mnt = 5p;
5. Delta # Mnt = 3p;
6. Pores # Mnt = 1p;

Where  $p =$  cloud window dimension, is set in such a way that the  $\max \text{dist} \leq$  considered max pixel value (sec 4). In case of template storage default cloud window size of  $21 \times 21$  is used. But if necessary the window size as well as the max pixel value may be increased in the run time of cloud image formation. Here the ridge bifurcation has been taken as most predominant feature and pores as lowest predominant.

#### 3.1 Template storage

In the enrolment phase a sub image of the fingerprint skeleton image is taken as the template due to the uncertainty of contact region of finger tip to the sensor. Wherever the template is stored in the smart card or a database, we prefer a DES [5] diversified template using a BCB mode. Where  $b-1 = g$  and the key is user supplied. For a verification based system or small volume database, we proposed to incorporate the fingerprint class information in the template hence building sub database or storing the information in unsecured ways. As example, column 0 of the template is the finger print type defined. We use a generic template window size of  $(m.8 \times n.8; m > 0, n > 0)$  for making the encryption and decryption easier. Figure 5 depicts a non encrypted template.

Fingerprint Group Indication

00	00	00	00	00	00
00	00	E7	00	00	00
00	00	00	00	00	00
00	00	00	15	00	00
01	00	00	93	00	00
00	00	00	00	00	00

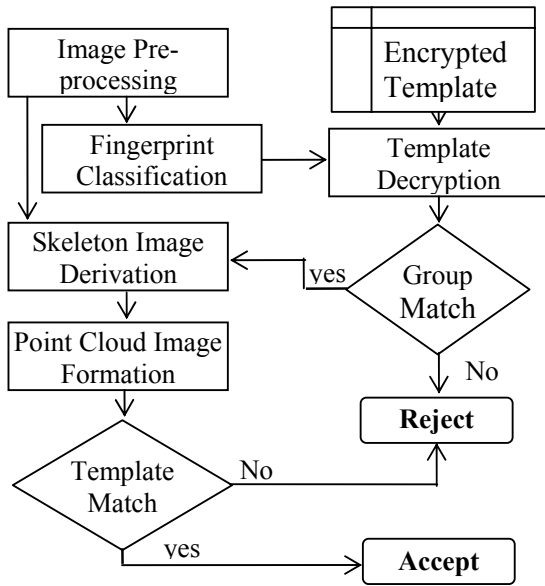
**Fig. 5** Example Template

#### 4. MATCHING ALGORITHM

The matching of the template against the fingerprint skeleton image is performed in two steps:

- Fingerprint group match
- Fine match using distance correlation between the skeleton cloud image and template image.

The success of fingerprint group match based on so called directional map [2] leads to the two in one template match. The complete matching scenario is shown in the block diagram of figure 6.



**Fig. 6** Complete Authentication System

As the match is intended to match the minutia point and type, the skeleton image is converted to cloud image.

##### 4.1 Cloud Image Formation

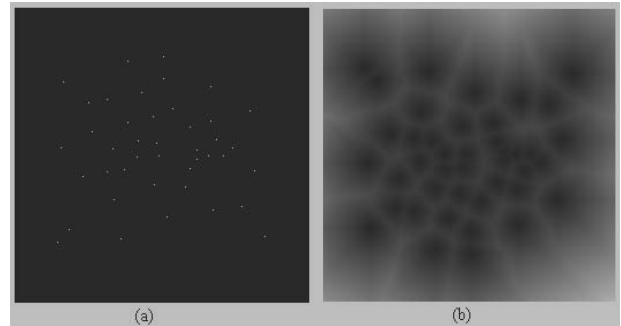
Cloud image is a special type of Distance Image [1]. The pixels in the usual distance image hold the distance value from the pixel of interest, most of the cases edge pixels. In our proposed cloud image the

pixel values are modified in such a way that it contains the distance information as well as the value of the pixel of interest. A 3x3 mask for the scheme is shown in figure 7.

Mnt+1	Mnt+1	Mnt+1
Mnt+1	Mnt	Mnt+1
Mnt+1	Mnt+1	Mnt+1

**Fig. 7** Mask for Cloud

At first the skeleton image is converted in such a way, that the pixels other than minutia pixels are converted to MaxDist (e.g. 255) grey value. Then the pixels values are updated according to the distance of the corresponding pixels from the pixel of interest and the grey value of the pixel. This operation could be accomplished by two scan from left top to right bottom and vice versa. Let's say the converted cloud image be  $D(x, y)$ . In figure 8 the skeleton image and the corresponding cloud image are shown, white pixels are far away from the pixel of interest. In the centre of the cloud Mnt located.



**Fig. 8** (a) Skeleton Image (b) Cloud Image

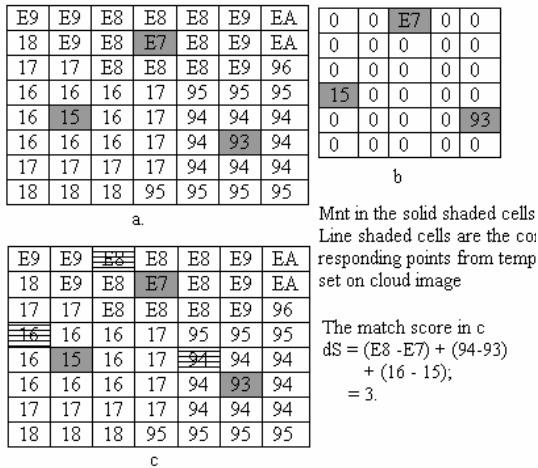
##### 4.2 Template Matching

For matching the template against the cloud image, the template is moved over the cloud image and the differences between the template pixel and cloud pixel are summed up for the correspondences, which is called match score (dS). Mathematically the score dS can be expressed as:

$$dS = \sum_{i=1}^n |D(T[template]) - T(x, y)| = f(\theta, tx, ty, S)..(2)$$

Where n is the total minutia points co-related, T is the transformed coordinates from the template,  $\theta$  is the rotational angle, tx, ty are the translation and S is the uniform scaling factor in the image plane. The lower the value of the dS, the better the match. A numerical example is shown in figure 9, where the

overlapping template on the cloud image produces a match score of 3.

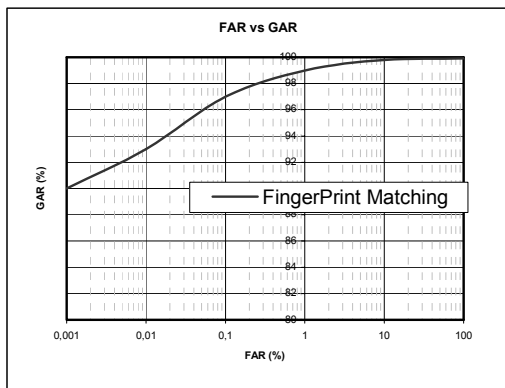


**Fig. 9** A numerical example of match (a) Cloud image, (b) Template, (c) Template overlapped on the cloud image,

In order to avoid any information gap between the template and skeleton image due to the uncertainty of having or having not the specific minutia point, the number of minutia matching and matching not is also taken into account.

## 5. RESULT AND DISCUSSION

The multidimensional matching features make the system very efficient and rigid. The DOF space is limited as the sensors are mounted in the system in such a way that the freedom of placing finger is also limited. The higher the space, the lower the efficiency of the system. The minimum value of dS is determined by Powell minimum search algorithm [6] with initial parameter values of equation 2 achieved from a grid search. In figure 10, a matching result has been shown:



**Fig. 10** Matching statistics

The matching system works very fine nearly 99.985% GAR (Genuine Acceptance Rate) with very

low FAR for the good quality of fingerprint images. For the bad quality of fingerprint images, the pre-processing e.g. filtering and minutia extraction needs to be taken care.

## 6. CONCLUSION

In matching we compensate the effect of pressure while placing the finger on the sensors by using the uniform scaling. Incorporating a pressure sensor could improve the efficiency of the system. Considering a higher grey level values and independent score producing cloud could also increase the matching performance. These two modifications are the ongoing investigation

## REFERENCES

- [1] Uwe Franke and Abu Ismail, "Recognition of Bus Stops through Computer Vision" in Proc. IV2003 IEEE Intelligent Vehicles Symposium Columbus, OH, USA, June 9-11, 2003, PP. 650 – 655.
- [2] Jain, A., and Pankanti, S., "Fingerprint Classification and Matching". Handbook for Image and Video Processing, A. Bovik (ed.), Academic Press, April 2000.
- [3] A. Wahab, S.H.Chin and E.C.Tan, "Novel approach to automated fingerprint recognition" in proc. IEEE Vision Technology and Image Signal Processing, Vol. 145, No. 3, PP. 160 – 166, June 1998.
- [4] Sanpachai Huvanandara, Changick Kim and Jenq-Neng Hwang, "Reliable and Fast Fingerprint Identification for Security Applications" in Proc. of International Conference of Image Processing ICIP 2000, Vancouver, Canada, pp 503-506.
- [5] Data Encryption Standard (DES), in Federal Information Processing Standards publication 46-3, available at <http://csrc.nist.gov/publications/fips/>
- [6] Chapter 10, Numerical Recipes in C++. The art of Scientific Computing, Second Edition. Saul A. Teukolsky, Brian P. Flannery- William H. Press, William T. Vetterling.
- [7] Foley, Van Dam, Feiner, Hughes, "Computer Graphics - Principle and Practice", 2nd Edition -1995, press Addison Wesley.
- [8] Jhon Canny, "A Computational Approach to Edge Detection", proc. IEEE Transactions on pattern Analysis and Machine Intelligence, 1986, 8(6):pp. 679-698.
- [9] R. Lotufo and F. Zampiroli "Fast multidimensional parallel euclidean distance transform based on mathematical morphology", in proc. of SIBGRAPI 2001, XIV Brazilian Symposium on Computer Graphics and Image Processing by IEEE Computer Society 2001, pp 100-105.