Fingerprint Classification Based on Extraction and Analysis of Singularities and Pseudoridges

Qinzhi Zhang, Kai Huang and Hong Yan
School of Electrical and Information Engineering
University of Sydney, NSW 2006, Australia

qzzhang@ee.usyd.edu.au

Abstract

In this paper, we introduce a new approach to fingerprint classification based on both singularities and traced pseudoridge analysis. Since noise exists in most of the fingerprint images including those in the NIST databases which are used by many researchers, it is difficult to get the correct number and position of the singulairities such as core or delta points which are widely used in current structural classification methods. The problem is we may miss the true singular points and/or get false singular points due to the poor quality of fingerprint images. Classification based on exact pair of singulairities will fail in such conditions. With the help of the pseudoridge tracing and analysis of the traced cvrve, our method does not rely on the extraction of the exact number and positions of the true singular points, thus improving the classification accuracy. This method has been tested on the NIST-4 fingerprint database. For the 4000 images in this database, the classification accuracy is 95.3% with 11.8% reject rate for 4-class problem (combining Arch and Tented Arch as one class).

Keywords: Biometrics, fingerprint classification, pseudoridge and singularity analysis

1 Introduction

Fingerprints have long been used for identification in many social conditions such as access control, crime investigation, and personal trust, since they will remain almost constant during people's life time as mentioned by Jain, L. C. et al(1999). Nowadays, Automatic Fingerprint Identification System(AFIS) and Automatic Fingerprint Recognition System(AFRS) as stated by Jain, L. C. et al (1999) are very popular due to their lower changability and easier accessability than other methods such as signature and hand geometry. Several public databases avsaifrom NIST also make it even more convenient for researchers in this area.

However, fingerprint identification is computationally demanding especially for a large database. So an effective indexing scheme can be of great help. Fingerprint classification, which classifies fingerprint images into a number of pre-defined categories, faciliates the matching task accordingly. Most classification methods are based on the Henry classes which classify the fingerprint images into arch, tented arch, left loop, right loop and whorl using the accurate position and type of the singular points such as cores and deltas. The following table and images give the general idea.

Pattern Class	Core	Delta
Arch	0	0
Tented Arch	1	1 (middle)
Left Loop	1	1 (right)
Right Loop	1	1 (left)
Whorl	2	2

Table 1 : Fingerprint pattern classes and the corresponding number of singular points

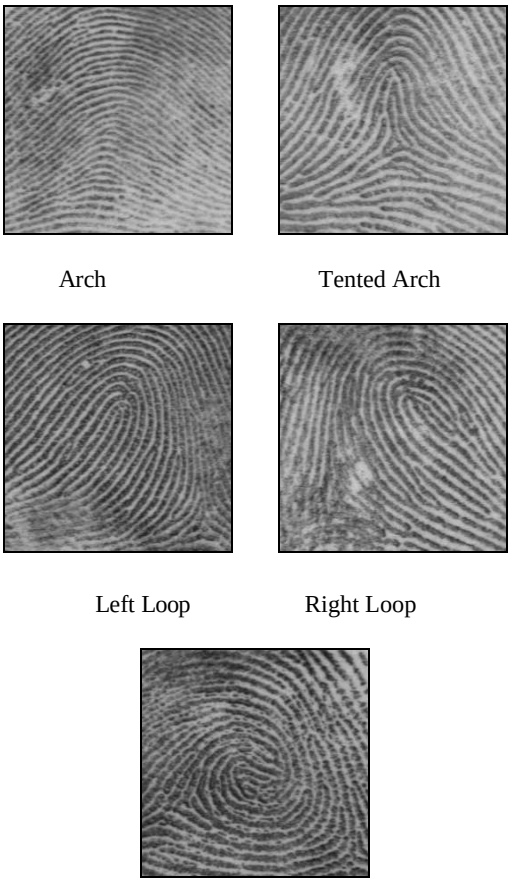


Fig. 1 Five Major fingerprint classes

There are two types of features to be extracted from a fingerprint image: high-level or global features, and low-level or local features. The high-level features form special patterns of ridges and furrows in the central region of the fingerprint. And the low level features characterize an individual fingerprint image by detailed ridge characteristics or minutiae.

The important high-level features are the core and delta points (also referred to as singular points or singularities). The core point is defined as the top most point on the inner most ridge and a delta point is defined as the point where three flows meet as stated by Jain, L. C. et al (1999). These points are highly stable and also rotation and scale invariant. Thus, the number and the location of core and/or delta points are widely used by most of the classification methods referred to the works (Cappelli, R. et al(1999), Karu and Jain(1996), Kawagoe and Tojo(1984)).

The quality of the fingerprint images may be poor, due to noise such as that caused by scars, breaks, too oily or too dry, or having a partial image, usually with the delta point outside the print. Even in one image, different regions have different quality. So, all of these factors make it extremely hard to classify such images according to the rules in Table 1.

We propose a fingerprint classification mehtod based on both the singularities and the analysis of the traced pseudoridge. With the help of the analysis of the traced pseudoridge used by Candela, G. T. et al(1995), Kawagoe and Tojo(1984), our method doesn't necessarily rely on the exact number and position of such singular points, and it also gives the general idea of the stuctural ridge pattern.

In the following sections, we will present the details of our fingerprint classification approach. Section 2 presents the singularities detection method. Section 3 presents pseudoridge tracing and analysis. In Section 4, we present our classification scheme. In Section 5, we present our experiment results on the NIST-4 database. The conclusions are presented in Section 6.

2 Singular Points Detection

Let $I(x, y)$ denote the gay level of the pixel (x, y) in an $M \times N$ fingerprint image.

Since most of the fingerprint images consist of the print itself, background and some handwritten letters or lines which may produce false singular points, it will be better to reduce the image to the print area only.

The blockwise average grayscale and standard deviation are used to segment the images. The block is considered as foreground if its grayscale mean and standard deviation satisfy some predefined standard, otherwise, the background. Then two iterations of dilation and erosion as stated by Gonzalez and Woods(1993) are used to remove holes resulting from inhomogenous regions. Also, such a segmented result will be used in the pseudoridge tracing step to define the stop boundary. All the processes discussed below are carried out on such foreground regions. The block size we use is 8.

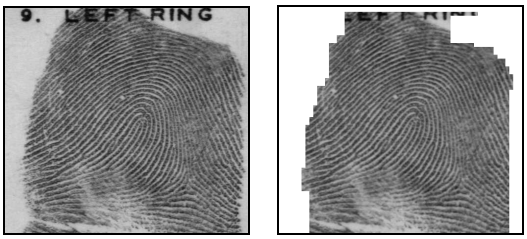


Fig. 2 The original image and the image after segmentation

2.1 Orientation Field Computation

Since fingerprints can be regarded as flow-like textures or oriented textures as mentioned by Rao(1990) and Kass and Witkin(1987), it is reasonable to use direction as the representation of fingerprint ridges or valleys. Thus, the orientaion field defined by Rao(1990), which is a matrix of directions representing the ridge or valley orientations at each location on the fingerprint image, is most commonly used in fingerprint verification systems.

The methods developed by Rao(1990), Cappelli, R. et al (1999), Vizcaya and Gerhardt(1996), Candela, G. T. et al(1995), Kass and Witkin(1987), Srinivasan and Murthy (1992) can be used to estimate the direction at each pixel in the image. Among them, due to the growing importance of the gradient vector field in orientation analysis as mentioned by Singh(1991), methods based on gradient have recently been widely used in the processing of fingerprint images by Kass and Witkin(1987), Rao(1990), Jain A. et al(1999), Jain, A. et al(1997).

Let $\tilde{\theta}(x, y)$ represent the orientation of the anisotropy, and $\theta(x, y)$ represent the local dominant orientation (or flow direction). The local dominant orientation $\theta(x, y)$ equals $\tilde{\theta}(x, y)+\pi/2$ since the flow orientation is perpendicular to the direction of anisotropy. For fingerprint images, in case the opposite directions cancel each other, we define the range of the direction angles as $(0, \pi)$.

Since the gradient of a Gaussian filter can give a good estimate of the underlying oriented pattern as stated by Kass and Witkin(1987) and Rao(1990), we adopt its orientation as the local direction. The image is smoothed with a Gaussian filter whose impulse response is given by $g(x, y)= e^{-(x^2+y^2)/2\sigma^2}$, then the optimum 3x3 operators developed by Ando(2000) are used to get the gradients

G_x and G_y, and the orientation of the gradient vector is given by

$$\alpha(x,y) = \arctan\left(\frac{G_y}{G_x}\right). \quad (1)$$

We use the mean direction of the neighborhood centered on the pixel defined by Mardia(1972) as its anisotropy orientation estimate, that is

$$\tilde{\theta}(x,y) = \frac{1}{2}\arctan\left(\frac{\sum_{(i,j)\in W_2}\sin 2\alpha(i,j)}{\sum_{(i,j)\in W_2}\cos 2\alpha(i,j)}\right). \quad (2)$$

For the fingerprint images in the NIST databases, the average width of the ridge or valley is five to eight pixels, so $W_2 = 16$ gives a better orientation estimate and saves computation time.

The orientation field is usually noisy, and in order to make singular points detection easier we smooth the orientation field using the method similar to the one developed by Jain, A. et al(1997).

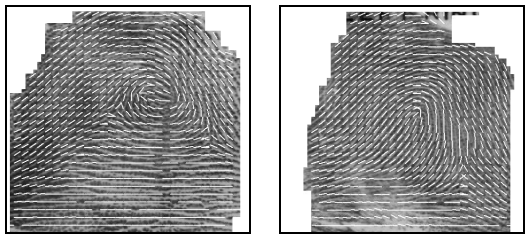


Fig. 3 Orientation overlayed on the original images

2.2 Singular Points Detection

Until now, a number of methods developed by Jain, L. C. et al(1999), Halici and Ongun(1996), Kawagoe and Tojo(1984),), Srinivasan and Murthy(1992) have been proposed for the detection of singular points in fingerprint images, while the Poincare index which is derived from continuous curves is the most popular one. As for digital fingerprint images, a double core point has a Poincare index valued as 1, a core point 1/2 and a delta point –1/2. So, the Poincare index is used first to find all the possible singular points in the orientation field.

The Poincare index at pixel (x,y) which is enclosed by a digital curve (with N_p points) can be computed as follows:

$$Poincare(x,y) = \frac{1}{2\pi}\sum_{k=0}^{N_p-1}\Delta(k), \quad (3)$$

where

$$\Delta(k) = \begin{cases} \delta(k), & if\ |\delta(k)| < \frac{\pi}{2} \\ \pi + \delta(k), & if\ \delta(k) \le -\frac{\pi}{2} \\ \pi - \delta(k), & otherwise \end{cases} \quad (4)$$

$$\delta(k) = \theta\left(x_{(k+1)\bmod N_p}, y_{(k+1)\bmod N_p}\right) - \theta(x_k, y_k) \quad (5)$$

and it goes in a counter-clockwise direction from *0* to N_p-*1*. For our method, N_p is selected as *4*.

Then the nearest neighbor cluster method is used to remove the false ones. Any types of potential singular points within some distance of each other are regarded as one cluster. After the clustering, we analyze the numbers of cores N_{ci} and deltas N_{di} in cluster *i*.

1. If $N_{ci} = N_{di}$, no singular point exists;
2. If $N_{ci} - N_{di} = 1$, the geometric center is regarded as a core point;
3. If $N_{ci} - N_{di} = -1$, the geometric center is regarded as a delta point;
4. If $N_{ci} - N_{di} = 2$, the geometric center is regarded as a double core;
5. Otherwise, no singular points exist in the cluster.

After this, there will be verification of such singular points. Five windows centered on the singular point are used to calcute their Poincare indices and the window size increases by *3* pixels. The singular points for which most Poincare indices conform to its type are verified. The numbers of cores and deltas are N_c and N_d.

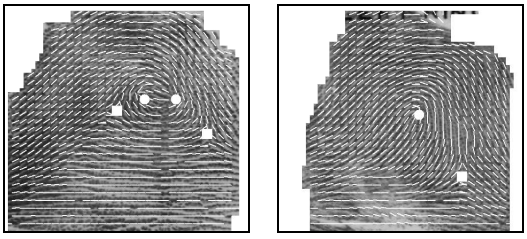


Fig. 4 Singular points found (circle as core and square as delta)

3 Pseudoridge Tracing and Analysis

If there exists one core point, we will start the Pseudoridge tracing. The detailed steps are as follows:

1. Starting from the core point, trace the orientation in two opposite directions and the tracing step size is predefined (2 for our approach); the tracing stops whether it reaches the segmented boundary or the steps reach a predefined value (80 in our algorithm);
2. From the two traces we have, we form two curves connecting at the core point. Then we select the one

that conforms to the left-hand rule from the start to the end;

3. Smooth the curve and compute the turn number.

A turn is defined if the angle between the two vectors satisfies the expression below as stated by Chong, M. M. S. et al(1997)

$$a\cos\left(\frac{\vec{V}_{i,i+2}\cdot\vec{V}_{j,j+2}}{\left|\vec{V}_{i,i+2}\right|\cdot\left|\vec{V}_{j,j+2}\right|}\right)\geq\frac{5\pi}{6},\ j-i\geq 2. \quad (6)$$

If only one turn is encountered, label the point where the turn happens as the turning point.

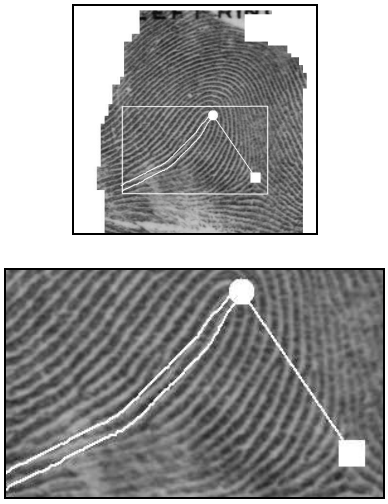


Fig. 5 Traced pseudoridge overlaid on the original image and the partly magnified subimage

4 Classification Rules

The classification rules are as follows:

1. If $N_c=N_d=0$, an Arch is assigned to the image;
2. If $N_c=2$, a Whorl is assigned;
3. If $N_c=1$, trace the pseudoridge and analyze the curve:
   - 3.1 If there are more than two turns, a Whorl is assigned;
   - 3.2 Otherwise, classify the image according to the positions of the character points of the curve:
     - 3.2.1 If both the start point and the end point are to the left of the turning point, a Left Loop is assigned;
     - 3.2.2 If both the start point and the end point are to the right of the turning point, a Right Loop is assigned;
     - 3.2.3 For the start point and the end point, if one is to the left of the turning point and the other to the right, a Tented Arch is assigned.
   - 3.3 If $N_d=1$, analyze the core/delta pair according to the following rules:
     - 3.3.1 Select a set of 16 rectangles with their axes along 16 directions starting from 0 to 15 and the core as the middle point of one of the shorter sides (as shown in the figure below) to obtain the orientation θ_c of the symmetrical axis of the region near the core,

$$\theta_c=\theta_{\min},$$

$$\underset{0\le k<16}{Min}\left(\sum_{(i,j)\in\psi_k}\left|\sin(\theta(i,j)-\theta_k)\right|\right)_{k=\min}$$

$$\theta_k=k*\pi/16 \quad (8)$$

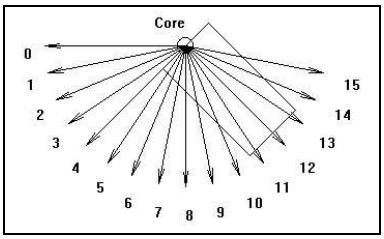


Fig. 6 16 directions and one of the corresponding rectangles

     - 3.3.2 Calculate the difference between θ_c and the oreintation θ_{cd} of the line connecting the core and delta points

$$\Delta\theta=\theta_c-\theta_{cd}; \quad (9)$$

     - 3.3.3 Then the class is determined as follows:

$$class=\begin{cases}TA, & |\Delta\theta|<\pi/12.\\ otherwise\begin{cases}L, & \Delta\theta<0.\\ R, & \Delta\theta>0.\end{cases}\end{cases} \quad (10)$$

   - 3.4 Combine the results from Step 3.2 and 3.3.3, then assign one class to the image.
4. Otherwise, reject the image.

5 Experiment Results

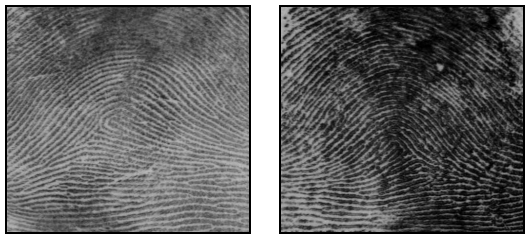
The classification algorithm described above has been tested on the fingerprint imges in the NIST-4 database. For all the 4000 images in the database, the 5-class results are shown in the following table. The first column shows the class index stored in the image file header, and the first row gives the assigned class using our method. The class index of one image in the NIST-4 database does not necessarily belong to only one class, so we consider our classification result to be correct if it matches any of the classes as the other researchers did in their works.

From the table, the 5-class classification result is around 84% without rejection. Because of the subtle differences between the Tented Arch and the Arch, it is reasonable to combine them into one class, and if so, the classification accuracy rises to 95.3% with 11.8% of the images rejected.

As. / True	Left Loop	Right Loop	Whorl	Tented Arch	Arch
Left Loop	**746**	3	16	3	39
Right Loop	5	**644**	28	3	48
Whorl	12	13	**624**	1	3
Tented Arch	25	4	8	**220**	307
Arch	3	2	1	1	**889**

Table 2: Experiment results tested on NIST-4

The second largest misclassification happens between the Arch and some of the Loops as they have very similar ridge patterns (see the figures below).



a b

Fig. 7 Two examples of the misclassified images. (a) A Right Loop as an Arch, (b) A Left Loop as an Arch

6 Conclusions

The classification algorithm we propose also works well when false singular points exist or true singular points are missing. If good enhancement methods are applied to the images or scale-variant orientation field smoothing methods are used, the result should be improved further. Also, the detection of singularities provides a good reference point for image registration which is mostly used for verification or matching, or even classification as the one developed by Jain, A. et al(1999).

7 References

JAIN, L. C., HALICI, U., HAYASHI, I., LEE, S. B. and TSUTSUI, S. (1999): *Intelligent Biometric Techniques in Fingerprint and Face Recognition.* CRC Press.

Singh, A.(1991): *OPTICAL Flow Computation: A Unified Perspective*, IEEE Computer Society Press.

RAO, A. R.(1990): *A Taxonomy for Texture Description and Identification.* Springer-Verlag.

MARDIA, K. V.(1972): *Statistics of Directional Data*, Accademic Press.

GONZALEZ, R. C. and WOODS, R. E.(1993): *Digital Image Processing*, Addison-Wesley.

Ando, S.(2000): Consistent Gradient operators. *IEEE Trans. On Pattern Analysis and Machine Intelligence* 20(3): 252-265.

JAIN, A., PRABHAKAR, S. and HONG, L.(1999): A Multichannel Approach to Fingerprint Classification. *IEEE Trans. On Pattern Analysis and Machine Intelligence* 21(4): 348-359.

CAPPELLI, R., LUMINI, A., MAIO, D., and MALTONI, D.(1999): Fingerprint Classification by Directional Image Partitioning. *IEEE Trans. On Pattern Analysis and Machine Intelligence* 21(5): 402-421.

CHONG, M. M. S., NGEE, T. H., JUN, L., and GAY, R. K. L.(1997): Geometric Framework for Fingerprint Image Classification. *Pattern Recognition* 30(9): 1475-1488.

KARU, K. and JAIN, A.(1996): Fingerprint Classification. *Pattern Recognition* 29(3): 389-404.

JAIN, A., HONG, L. and BOLLE, R.(1997): On-Line Fingerprint Verification. *IEEE Trans. On Pattern Analysis and Machine Intelligence* 19(4): 302-314.

VIZCAYA, P. R., and GERHARDT, L. A.(1996): A Nonliear Orientation model for Global Description of Fingerprints. *Pattern Recognition* 29(7): 1221-1231.

HALICI, U., and ONGUN, G.(1996): Fingerprint Classification Through Self-Organizing Feature Maps Modified to Treat Uncertainties. *Proc. of the IEEE* 84(10).

CANDELA, G. T., GROTHER, P. J., WATSON, C. I., WILKINSON, R. A. and WILSON, C. L.(1995): PCASYS-A Pattern_Level Classification Automation System for Fingerprint, Technical Report NISTIR.

KAWAGOE, M. and TOJO, A.(1984): Fingerprint Pattern Classification. *Pattern Recognition* 17(3): 295-303.

KASS, M. and WITKIN, A.(1987): Analyzing Orientaed Patterns. *Computer Vision, Graphics, ans Image Processing* 37: 362-385.

SRINIVASAN, V. S. AND MURTHY, N. N.(1992): Detection of Singular Points in Fingerprint Images. *Pattern Recognition* 25(2): 139-153.