Robust Reference Point Detection Using Gradient of Fingerprint Direction and Feature Extraction Method

Junbum Park and Hanseok Ko

Department of Electronics Engineering, Korea University
5Ka-1, Anam-dong, Sungbuk-ku, Seoul, 136-701, Korea.

jbpark@ispl.korea.ac.kr
hsko@korea.ac.kr

Abstract. A novel reference point detection method is proposed by exploiting the GPM(Gradient Probabilistic Model) that captures the curvature information of fingerprint texture. The detection of reference point is accomplished through searching and locating the points of occurrence of the most evenly distributed gradient in probabilistic sense. We also propose a novel filterbank method to improve shortcoming of existing filterbank method in verification part. Existing filterbank method can lose the discerning attributes because the sectors of the outer band from the reference point are larger in size than those of the inner bands. Such shortcomings of the filterbank method are resolved by maintaining the attribute regions to equal size.

1 Introduction

Recent surge in the interest of personal identification by means of biometrics is gaining even more attention due to the needs of reliable security system that's both affordable in cost while enabling user convenience. Biometric solution is particularly attractive since the traditional hand-carrying ID card has the risk of being lost or stolen and that an imposter can easily forge their way to get access. Among the many possible biometric schemes, the fingerprint is one of the most reliable methods for the identification of individuals because there are no two persons having the same fingerprint and for each fingerprint, it remains unchanged over a lifetime and is easy to acquire. For the fingerprint verification, the most technically critical elements are the detection of a reliable reference point and the extraction of key features based on the reference point detected. Fig. 1 depicts the usual flow of data processing required to perform the fingerprint verification. Poincare index has recently been by far the most widely used approach in detecting the target reference point (e.g core point or delta point)[1][2][3]. It essentially looks for the position of occurrence of high gradient components in the texture field. However, because the method is conditioned on detecting the variations in gradient components, it is prone to errors induced by cuts in a flow of finger print ridge contour or foreign materials coated on the finger surface. As a result, the procedure is sensitive to the quality of captured image and thus a detailed preprocessing is required. Thus, while it is well suited for detecting the reference point in fingerprints of loop type, double-loop type or whole type, it

performs poorly on the arch types having vague gradient. To remedy the weakness over the arch type textures, we employ the gradient direction information (0,45,90,135) in the form of gradient probability distribution. In particular, we determine the gradient probability distribution over the surface texture of the fingerprint and then search the point of occurrence of equal probability in the four principal directions. Through finding the spot of uniform density, we attain the robustness in the verification performance on various fingerprint types including those of the arch type, while suppressing the performance sensitivity to the image quality. This paper is composed as the following sections, we will present the details of our reference point detection algorithm(GPM) and feature extraction algorithm using new filterbank method. Section 2 presents our reference point detection algorithm. In Section 3, feature extraction algorithm is presented. Moreover, In Section 4, we present our experimental results obtained by the employed semiconductor based sensor that show the improvements in terms of accuracy and computational load under various noisy scenarios. The conclusions and future research directions are presented in Section 5.

Fig. 1. Typical fingerprint verification procedure

2 Reference Point Detection Using GPM(Gradient Probabilistic Method)

In this section, we propose GPM that uses the directional information and to expand the coverage of the fingerprint types that includes the arch-type. The directional information can be attained by extracting 4 directional data (0, 45, 90, 135) in fingerprint by taking a gradient over the entire image and by computing the directional probability at each of the directional angles in 50-by-50 raster scanning windows. We then search for the region where the direction probability distribution is most flat (uniform density). In addition, because the method uses the directional distribution information, no detailed preprocessing is required as it is not as sensitive to precise image events. The Poincare index method requires costly and detailed

preprocessing such as connecting ridge lines when the lines are cut by noise. The procedure of the proposed GPM is shown below Fig. 2.

Fig. 2. Reference Point Detection Algorithm using GPM

2.1 Extraction of Fingerprint Region

Fingerprint region is obtained by dilation/erosion operation(DE). To obtain the fingerprint region(FR)(Fig. 3(c)), we compute mean value of input image(Fig. 3(a)). After binary image is obtained by (Eq.1), we extract fingerprint region using dilation and erosion operation.

$$\text{binary image } (i, j) = \begin{cases} 1, & \text{input image } (i, j) > \text{threshold } (\text{mean} + \alpha) \\ 0, & \text{Otherwise} \end{cases} \quad (1)$$

(a) Input image (b) DE image (c) FR image

Fig. 3. Fingerprint region extraction image

2.2 Image Distinguishing Ridge from Valley

To obtain the binary image, we divide the method into using global threshold and local threshold. The fingerprint image used can be either bright or dark because of the variations in illumination from the sensor. Thus, to blindly set the global threshold is not appropriate. We, therefore, use the local threshold to distinguish the ridge from

valley. The Local threshold takes the average image of fingerprint by convolving a circle of 17×17 size to the captured image. The binary image (Fig. 4(b)) distinguishing ridge from valley is then obtained by following Eq. 2.

$$\text{binary image}(i, j) = \begin{cases} 1, & \text{FR image}(i, j) > \text{average image}(i, j) \\ 0, & \text{Otherwise} \end{cases} \quad (2)$$

$$\begin{aligned} \text{average image}(i, j) &= \text{FR image}(i, j) * \text{unit circle filter} \\ \text{unit circle filter} &= \{(x - i)^2 + (y - j)^2 = r^2\} \end{aligned}$$

(a) Average image (b) Binary image

Fig. 4. Average image and distinguishing ridge from valley

2.3 Direction Component Extraction of Fingerprint

Direction information of a fingerprint is divided into four components. Namely, direction components of 0,45,90,135 degrees are extracted Fig. 5 and the size of direction filter is set at 5×5 masking window.

(a) 0° filter (b) 45° filter (c) 135° filter (d) 90° filter

Fig. 5. Direction filter of 5×5 size

Moreover, the images in Fig. 6 are obtained by operating convolution between direction filters of Fig. 5 and the binary image of Fig. 4(b).

(a) 0° (b) 45° (c) 135° (d) 90°

Fig. 6. Extracted binary direction components

2.4 Reference Point Detection

In Fig. 6, we search for the regions where all four direction probabilistic distributions are the most equal using a Parzen window[4]. The point of uniform density can be extracted by counting the sampling numbers of each direction component in a fixed window. Parzen window is a good approximating approach that establishes the probability distribution at each direction components in the window prescribed by the following procedure.

- 1) Set Parzen window of fixed size $N \times N$.
- 2) Perform convolutions of each direction components shown in Fig. 6 and the Parzen window. Count the number of samples of each direction component as the window is raster scanned across the entire image.
- 3) Select the point of the most equal probability in all direction. In step 1, the size of window is set at 50×50 by experiment. The Parzen window contains 2005 pixels. Computation of the probability distribution in step 2 is denoted in Eq. 3.

$$P(\theta_i | I(x, y)) \Rightarrow \sum_i P(\theta_i | I_i(x, y)) = 1, \quad P(\theta_i) = \frac{\text{count}(I_i(x + \Delta x, y + \Delta y))}{S} \quad (3)$$

where, $I_i(x, y)$ is coordinate (x, y) of all pixels in image, $P(\theta_i)$ is probability of each directional component in coordinate (x, y) , S is an area of Parzen window and $i = 0^\circ, 45^\circ, 90^\circ, 135^\circ$. After direction probability of all pixels in image is computed, we use a relative entropy concept to find the point of the most equal probability. If A, B are distribution, a relative entropy (H) defined by

$$H(A(x + \Delta x, y + \Delta y), B(x + \Delta x, y + \Delta y)) = \left| \sum_i a_i(x + \Delta x, y + \Delta y) \log \frac{a_i(x + \Delta x, y + \Delta y)}{b_i(x + \Delta x, y + \Delta y)} \right| \quad (4)$$

where, $H(A, B)$ is a relative entropy between two distribution A and B , $a_i(x + \Delta x, y + \Delta y)$ is PMF(Probability Mass Function) really computed in all position $(x + \Delta x, y + \Delta y)$ of i -direction and $b_i(x + \Delta x, y + \Delta y)$ is PMF(threshold) in all position $(x + \Delta x, y + \Delta y)$ of i -direction. Namely, Eq. 4 is to measure the distance between A and B . Therefore, the more similar A and B , the smaller the relative entropy (H). In here, threshold PMF of each direction set 0.2 by experiment. Finally, we select a reference point having the most minimum value by Eq. 5.

$$\arg \min \left(\sum_i H_i(x + \Delta x, y + \Delta y) \right) \quad (5)$$

where, H_i is entropy of PMF really computed by Parzen window, $(x + \Delta x, y + \Delta y)$ is coordinate. Fig. 7 shows the distance between A and B using relative entropy. In here, a solid arrow denotes a standard PMF for all pixels of each direction and a dotted arrow denotes PMF computed by Parzen window at each coordinate. Moreover, d_i denote distance between two distribution by relative entropy in i -direction. We

select point of the most equal probability by relative entropy in all direction. Therefore, the point has the most equal direction components.

Fig. 7. Distance between two distribution by relative entropy

Fig. 8 shows reference point selected using entropy.

(a) Image by relative entropy (b) Reference point

Fig. 8. Reference point detection image by relative entropy

3 Feature Extraction Using Filterbank Method

Once a reference point is established, next step is to extract unique and characterizing features of the fingerprint to be able to distinguish it from others. Existing filterbank method can lose the discerning attributes because the sectors of the outer band from the reference point are larger in size than those of the inner bands as shown in Fig. 9(a). Moreover, the nearest sectors from the reference point were found to be sensitive to the specific position of reference point extracted. Such shortcomings are avoided by maintaining the attribute regions to equal size. A novel filterbank method employed here extracts identifying features from each region as listed in the following procedure:

- 1) Extract 8 direction components (between 0 and 157.5, 22.5 interval) using Gabor-filter[5][6][7] over the region.
- 2) Proceed with a normalization procedure in which the mean and variance of the entire image is adjusted because Gabor-filtered image is influenced by brightness.

In this way, eighty features (mean, variance) with respect to the reference point were determined by computing the parameters over each normalized image. In step 1, Gabor-filter is defined by

Fig. 9. Segmentation of regions for feature extraction

$$g(x, y, \theta_k, \sigma_x, \sigma_y) = \exp \left[-\frac{1}{2} \left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} \right) \right] \times \exp(i2\pi f x \theta_k), \quad (6)$$

$$x\theta_k = x \sin \theta_k + y \cos \theta_k, \quad y\theta_k = x \cos \theta_k - y \sin \theta_k,$$

Note that f is the frequency of the sinusoidal plane wave, that is, the number of pixels between band and band. Also, θ_k is the orientation of the Gabor-filter, while σ_x and σ_y are the standard deviations of the Gaussian envelop along the x and y axes, respectively. Fig. 10(b) shows normalized image of eight-direction by Gabor-filter using Eq. 6. The model to compute mean and variance can be expressed by

$$D(\text{deg}, x, y) = \sum_{w=-10}^{10} \sum_{v=-10}^{10} g_f(w, v) \cdot g_{\text{deg}}(x+w, y+v) \cdot g_{\text{deg}}(x+w, y+v) - \mu_g^2(x, y), \quad (7)$$

$$\mu_g(x, y) = \sum_{w=-10}^{10} \sum_{v=-10}^{10} g_f(w, v) \cdot g_{\text{deg}}(x+w, y+v),$$

g_{deg} is a feature value of the eighty feature points (x, y) from the filtered image. Moreover, w and v reflect the filter size from feature point (x, y) . If filter size is too large, an overlapping can occur among nearby feature points and thereby may lose the essential attributes. If filter size is too small, there may not be enough features to be captured. We determined the optimum filter size to be 11×11 by experiment. Note that $g_f(w, v)$ denotes a convolution of mean-filter and Gaussian-filter. We can reduce computational load by designing a novel filter as $g_f(w, v)$. Eq. 8 denotes mean-filter having mean values in all coordinates.

$$M = \frac{1}{N} \sum_{i=-S_w}^{S_w} \sum_{j=-S_v}^{S_v} P(i, j) \quad (8)$$

In here, M is the mean value of the neighboring region for each pixel, S_w and S_v are the filter size along the x -axis and y -axis, $P(i, j)$ is the value in pixel (i, j) , and N is the total number of pixels in the neighboring region. Moreover, the Gaussian filter (Eq. 9) is to weight for each feature point as ridges and to reduce weight for part as valley. Namely, we can obtain more accurate feature point by weighting for feature point.

$$G = e^{-\frac{r^2}{\sigma^2}}, \quad r = \sqrt{(x - \text{size})^2 + (y - \text{size})^2} \quad (9)$$

where, G is a Gaussian filter, (x,y) is the coordinate, size is the size of Gaussian filter. The Gaussian filter size was set to 18 by experiments. The σ value denotes the width of the Gaussian filter. If σ is too large, the filter would be more robust to noise, but would more likely smooth the image to the extent that the ridge and valley details in the fingerprint may get lost. If σ is too small, the filter would not be effective in removing the noise. The value was determined empirically and was set at 36. Note that $g_f(w,v)$ is defined as a convolution between the mean filter and the Gaussian filter.

$$g_f(w,v) = G * M \tag{10}$$

Note also that μ_g denotes convolution between Gabor-filtered image and $g_f(w,v)$ filter.

Fig. 10. Gabor-filtered normalized image of region of interest

4 Experimental Result

To evaluate the performance of our verification system, we made experiment as the following. The first series of experiments is to evaluate accuracy of reference point detection. We obtained each 6 images from the 20 same persons. So, a total of 120 fingerprint images are used. In here, we decided reference point position of one image having the best quality to the database. Then, we computed distance of the others and the reference point position. Table 1 shows average of distance for each subject. Particularly, Poincare has a defect for arch type. Fig. 11 shows this example.

Table 1. Average of distance for each subject between GPM and Poincare index method

	Loop	Double loop	Whorl	Arch
GPM	5.12(pixel)	6.39(pixel)	6.14(pixel)	6.08(pixel)
Poincare	7.08(pixel)	2.78(pixel)	5.39(pixel)	More than 20(pixel)

Fig. 11. Reference point detection example of GPM and Poincare index for arch type

The second series of experiments is to test the verification rate of our system. We took several fingerprint images from each subject and registered one image having the best quality to the database. This process was repeated for all 23 test subjects, and in the process a total of 211 fingerprint images were taken. In here, we made experiment under general condition and two separate noise conditions(noise due to variation in brightness and salt and pepper noise). Table. 2-4 shows these examples.

The third series of experiments is to evaluate the performance of GPM method as a function of the Parzen window size. Fig. 12 shows performance of the GPM method using 4 different sizes of the Parzen window. From this comparison, 50x50 window was adopted as the most optimum size for the verification process.

Table 2. Verification rate of a novel method(above) and Poincare index(below) under general environment

GPM	Loop	Double loop	Whorl	Arch	Total	Verification rate
FRR(%)	5.8	10.8	15.2	13.7	10.9	99.53
FAR(%)	0	0	0	0	0	
Poincare	Loop	Double loop	Whorl	Arch	Total	Verification rate
FRR(%)	14.7	7.8	8.7	62.7	23.5	98.95
FAR(%)	0	0	0	0	0	

Table 3. Verification rate of a novel method(above) and Poincare index (below) under brightness variation noise

GPM	Loop	Double loop	Whorl	Arch	Total	Verification rate
FRR(%)	10.3	23.9	19.6	27.5	19.4	99.16
FAR(%)	0	0	0	0	0	
Poincare	Loop	Double loop	Whorl	Arch	Total	Verification rate
FRR(%)	19.1	10.9	15.2	66.7	28.0	98.8
FAR(%)	0	0	0	0	0	

Table 4. Verification rate of a novel method(above) and Poincare index (below) under salt and pepper noise

GPM	Loop	Double loop	Whorl	Arch	Total	Verification rate
FRR(%)	14.7	32.6	19.6	41.2	26.1	98.87
FAR(%)	0	0	0	0	0	
Poincare	Loop	Double loop	Whorl	Arch	Total	Verification rate
FRR(%)	19.1	13.0	15.2	56.9	26.1	98.87
FAR(%)	0	0	0	0	0	

5 Conclusion and Future Works

We developed an on-line fingerprint verification system which operates in three steps: reference point detection, feature extraction, and matching. From the preliminary results, it has been shown that the proposed GPM method yielded superior verification performance compared to other conventional verification methods such as the Poincare algorithm. It has shown improved performance over the conventional method in a low noise environment, and it demonstrated more robust performance in a high noise environment such as the one caused by variations in image brightness. It was also shown that the new method requires less verification time compared to the conventional method. These are due to the fact that the method uses directional probability distribution information only, thus requiring no detailed preprocessing and

Fig. 12. Verification rate of the GPM method using 4 Parzen window sizes

making it less sensitive to the precise image events. Moreover, the most significant performance improvement attributed to the method is in the case of verifications against the arch type fingerprint images. Large reductions in FRR were noted consistently on all three cases of the comparisons against the Poincare algorithm. This improvement is due mainly to the GPM's ability to locate a more repeatable reference point from fingerprint images of the same subject. Our future work will be focused on the enhancement of the verification performance by improving the present algorithm under various noise environments. Because the preprocessing procedure is important in researching a more robust algorithm under noise environments, we will direct our future research towards local processing rather than global processing about filters used for preprocessing.

References

1. A. K. Jain, L. Hong, and R. Bolle, "On-Line Fingerprint Verification," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 302–313, April 1997.

2. Lin Hong, Yifei Wan, Anil Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation" in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 777–789, August, 1998.
3. Kalle Karu and Anil K. Jain, "Fingerprint Classification," in *Pattern Recognition*, vol. 29, no. 3, pp. 389–404, 1996.
4. Richard O. Duda, Peter E. Hart, David G. Stork, *Pattern Classification*, John Wiley & Sons, 2001.
5. Y. Hamamoto, S. Uchimura, M. Watanabe, T. Yasuda, Y. Mitani and S. Tomita, "A Gabor filter-based method for recognizing handwritten numerals," in *Pattern Recognition*, Vol. 31, No. 4, pp. 395–400, 1998
6. Anil. K. Jain, Salil Prabhakar, Lin Hong, and Sharath Pankanti, "Filterbank-based Fingerprint Matching," *IEEE Transactions on Image Processing*, vol. 9, no. 5, pp. 846–859, May 2000
7. Anil K. Jain, Salil Prabhakar, and Lin Hong, "A Multichannel Approach to Fingerprint Classification," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 348–359, April 1999
8. Anil K. Jain, Lin Hong, Sharath Pankanti, and Ruud Bolle, "An Identity-Authentication System Using Fingerprints," in *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1365–1204, September 1997.