IMPROVED FINGERCODE FOR FILTERBANK-BASED FINGERPRINT MATCHING

Lifeng Sha, Feng Zhao, and Xiaoou Tang

Department of Information Engineering
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
{lfsha1, fzhao0, xtang}@ie.cuhk.edu.hk

ABSTRACT

FingerCode has been shown to be an effective representation to capture both the local and global information in a fingerprint. However, the performance of FingerCode is influenced by the reference point detection process, and the AAD features cannot fully extract the discriminating information in fingerprints. In this paper, we first propose a new rotation-invariant reference point location method, and then combine the direction features with the AAD features to form an oriented FingerCode. Experiments conducted on a large fingerprint database (NIST-4) show that the proposed method produces a much improved matching performance.

1. INTRODUCTION

The fingerprint matching algorithms can be broadly classified into two categories: (*i*) minutiae-based, and (*ii*) filterbank-based. The minutiae-based matching algorithms first extract the local minutiae such as ridge endings and ridge bifurcations from the thinning image [2][7] or the grayscale image [6], and then match their relative placement in a given fingerprint with the stored template. A number of matching techniques are available in the literature including point-based matching [2], graph-based matching [1], and string-based matching [7]. Although the minutiae-based matching is widely used in fingerprint verification, it has problems in efficiently matching two fingerprint images containing different number of unregistered minutiae points. Further, it does not utilize a significant portion of the rich discriminatory information available in the fingerprints. The filterbank-based matching algorithm [3][4][5] uses a bank of Gabor filters to capture both local and global information in a fingerprint as a compact fixed-length FingerCode, which is suitable for matching and storage. The fingerprint matching is based on the Euclidean distance between two corresponding FingerCodes. Thus, it overcomes some of the problems with the minutiae-based matching algorithms.

However, the original feature extraction algorithm [5] assumes that the fingerprints are captured in a vertical position. While the reference point location method works well for vertically oriented fingerprints, it fails to precisely locate the reference point after the original fingerprint is rotated. In this paper, we propose a new rotation-invariant reference point location method. Furthermore, the AAD feature in the previous work [5] does not utilize the direction information that characterizes the oriented flow pattern of ridges and valleys in a fingerprint. In order to improve the overall matching performance, we combine the AAD features with the corresponding direction features to generate an oriented FingerCode. Experimental results obtained from a standard fingerprint database (NIST-4) confirm the effectiveness of the proposed method.

2. ORIGINAL FINGERCODE

The original FingerCode generation and matching process [5] can be summarized in the following steps:

1. Locate the reference point and determine the region of interest for the fingerprint image;
2. Tessellate the region of interest centered at the reference point. The region of interest is divided into a series of B concentric bands and each band is sub-divided into k sectors (B = 5 or 7, depending on the image size; k = 16);
3. Normalize the region of interest. The gray values in every sector are normalized to a specified constant mean M_0 and variance V_0, thus to remove the effects of sensor noise and gray level deformation due to the finger pressure differences;
4. Filter the region of interest in eight different directions using a bank of Gabor filters, thus produce a set of eight filtered images. The tuned Gabor filters enhance the ridges and valleys that are oriented at the same angle as the filter and suppress the ridges and valleys oriented at other angles;

5. Compute the average absolute deviation from the mean (AAD) of gray values in each of the 80 sectors for every filtered image to form the FingerCode;
6. Rotate the features in the FingerCode cyclically to generate five templates corresponding to five rotations (±45°, ±22.5°, 0°) of the original fingerprint image, thus to approximate the rotation-invariance;
7. Rotate the original fingerprint image by an angle of 11.25° and generate its FingerCode. Another five templates corresponding to five rotations are generated in the same way as Step 6;
8. Match the FingerCode of the input fingerprint with each of the ten templates stored in the database to obtain ten matching scores. The final matching score is the minimum of the ten matching scores, which corresponds to the best matching of the two fingerprints.

3. REFERENCE POINT LOCATION

The reference point is the center of the region of interest. Locating the reference point is an essential step that can influence the matching accuracy. The reference point detection algorithm developed in [5] has been shown to be more effective than most previous methods. However, we find that it is sensitive to fingerprint rotation. In this work, we develop a new reference point location algorithm that is rotation invariant. Our reference point location algorithm includes the following steps:

1. Compute the directional image using the same algorithm as in [5]. Let D be the smoothed directional image with $D(i, j)$ representing the local ridge direction at pixel (i, j);
2. Initialize a label image L with the same size of D. L will be used to indicate the reference point;
3. As shown in Fig. 1, for each pixel (i_c, j_c) in D, define a local region S centered around (i_c, j_c). Assign the corresponding pixel in L the value of the following summation:

$$L(i_c, j_c) = \sum_{(i,j)\in S} \left|\cos(\theta_t(i,j) - \theta_d(i,j))\right|$$

   where $\theta_d(i, j)$ represents the local ridge direction of a pixel (i, j) in S, $\theta_t(i, j)$ is the direction perpendicular to the line linking (i, j) and (i_c, j_c). If (i_c, j_c) is the center of the ridge curves, θ_d and θ_t will be similar at most points within S, thus will produce a high value in L;
4. Search for the maximum value M_1 and the secondary maximum value M_2 in L. If M_2 is over 0.95 times of M_1, there may be a double loop in the fingerprint image, so we assign the coordinates of the midpoint of the line M_{12} as the reference point; otherwise, we assign the coordinates of M_1 as the reference point.

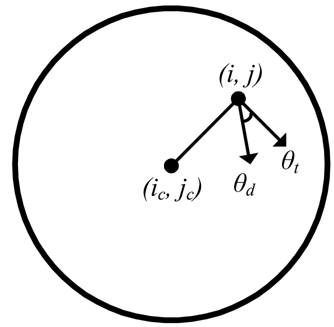


Figure 1. Region S is centered around pixel (i_c, j_c). Pixel (i, j) is within the region S. θ_t and θ_d represent the tangent direction and the local ridge direction of pixel *(i,j)*, respectively.

Figure 2 and 3 show some results of our reference point location algorithm. The new algorithm can locate the reference points for fingerprint images of different classes. However, the algorithm may fail to locate the reference point for the fingerprint with very poor quality as shown in Fig. 3 (the last image). The examples shown in Fig. 4 illustrate the rotation-invariance of our algorithm compared with the algorithm in [5].

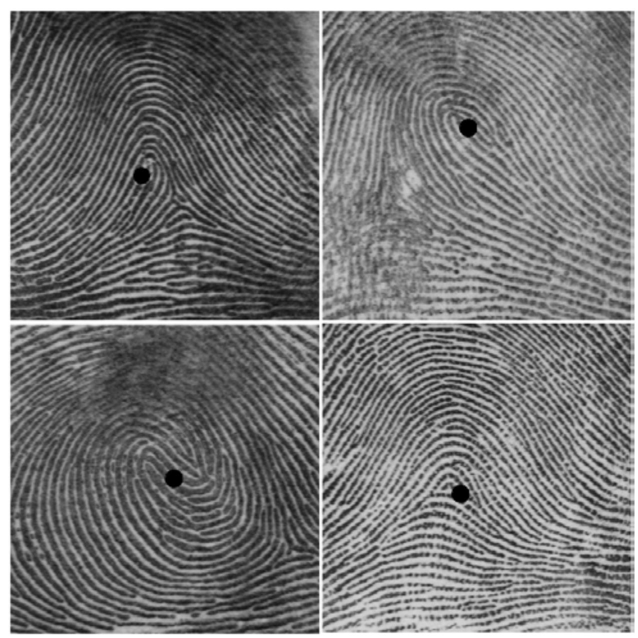


Figure 2. Examples of the reference point location results for different fingerprint classes.

4. ORIENTED FINGERCODE

The normalized region of interest is filtered using a bank of Gabor filters. The typically used Gabor filter [3][4][5] is an even symmetric real-valued Gabor filter. It has the following general form in the spatial domain:

$$G(x, y; f, \theta) = \exp\left\{-\frac{1}{2}\left[\frac{x'^2}{\delta_{x'}^2} + \frac{y'^2}{\delta_{y'}^2}\right]\right\}\cos(2\pi f x') \qquad (1)$$

$$x' = x\sin\theta + y\cos\theta \qquad (2)$$

$$y' = x\cos\theta - y\sin\theta \qquad (3)$$

where f is the frequency of the sinusoidal plane wave along the direction θ with respect to the x-axis, and $\delta_{x'}$ and $\delta_{y'}$ are the standard deviations of the Gaussian envelope along x'- and y'-axis, respectively. The filterbank consists of eight Gabor filters, each differing in orientation θ only ($f = 0.1$, reverse of the average inter-ridge distance; $\delta_{x'}$, $\delta_{y'}$ = 4.0, about half of the average inter-ridge distance.). The eight values of θ are 0^o, 22.5^o, 45^o, 67.5^o, 90^o, 112.5^o, 135^o, and 157.5^o. Thus, the filtering process produces a set of eight filtered images.

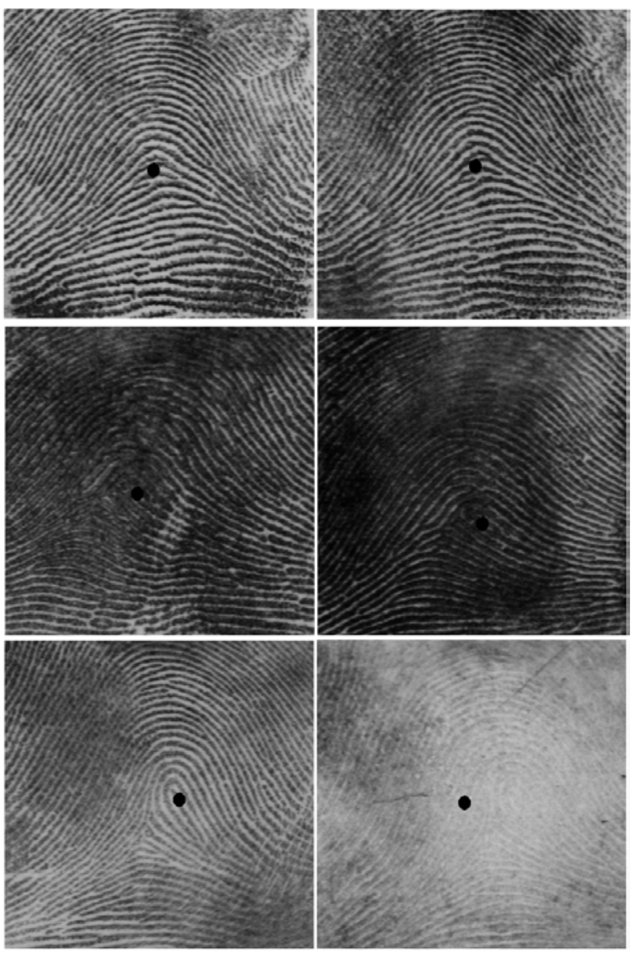


Figure 3. Reference point location results of the fingerprint pairs of different quality. (Top row: fingerprint pairs of high quality; middle row: fingerprint pairs of moderate quality; bottom row: fingerprint pairs of poor quality.)

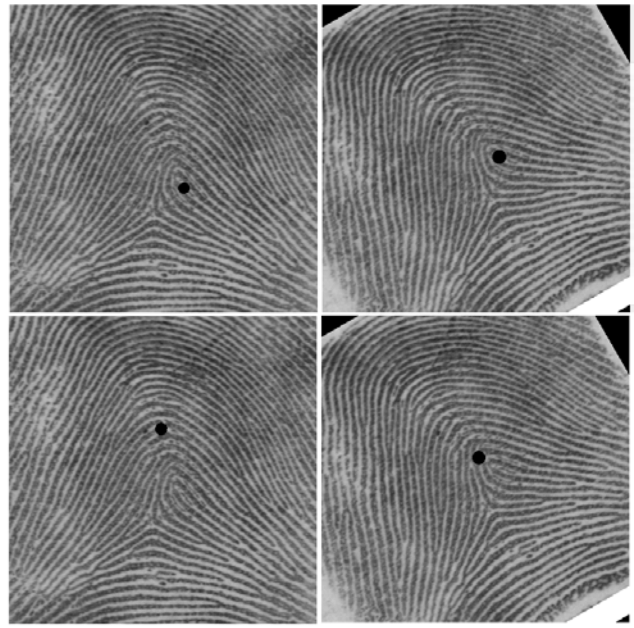


Figure 4. Examples to illustrate the rotation-invariance of our algorithm (top row) compared with the algorithm in [5] (bottom row).

Let F_m (m = 0, 1, … , 7) be the m_{th} filtered image. We define a new 2-D feature vector for each sector s_i (i = 0, 1, …, 79) in F_m as V_{im}. The magnitude value of V_{im} is the average absolute deviation from the mean (AAD) as defined in [5]; and the angle of V_{im} is the dominant local ridge direction in sector s_i. The features in all the sectors constitute a sub-FingerCode for the filtered image F_m. The collection of all the eight sub-FingerCodes is defined as the oriented FingerCode. Similar to [5], to approximate the rotation invariance, we cyclically rotate the features and the image to generate ten templates. The matching is based on the Euclidean distance between two corresponding FingerCodes. Let V_{im} and V_{jm} represent the vector features for the two corresponding sectors in the m_{th} sub-FingerCode. Then, the Euclidean distance between two FingerCodes is defined as follows:

$$ED = \sum_{m=0}^{7}\sum_{i,j=0}^{79}\left|V_{im} - V_{jm}\right|$$

Note that there is an ambiguity of π in the local ridge direction, i.e., local ridges oriented at π/4 and 5π/4 cannot be differentiated from each other. If the intersecting angle between V_{im} and V_{jm} is larger than ninety degrees, we reverse one of them to obtain a smaller distance. Experimental results show that using the oriented FingerCode gives a better performance than using AAD features only. Since the local ridge direction information is computed at the reference point location stage, the addition of the local ridge direction as a new feature will have almost no effect on the matching time.

5. EXPERIMENTS

Experiments are conducted on a standard fingerprint database NIST-4, which contains a set of 2000 fingerprint image pairs (512x512, 256 gray levels, 500 dpi). Each fingerprint pair has two different rolled impressions of the same finger. We remove some fingerprints if their reference points are too close to the edge of the image.

To test the performance of the oriented FingerCode, we first implement a filterbank-based fingerprint matching system using the original approach [5]. Here, we replace the reference point location algorithm with the new algorithm discussed in Section 3. The dataset is then processed using the implemented system.

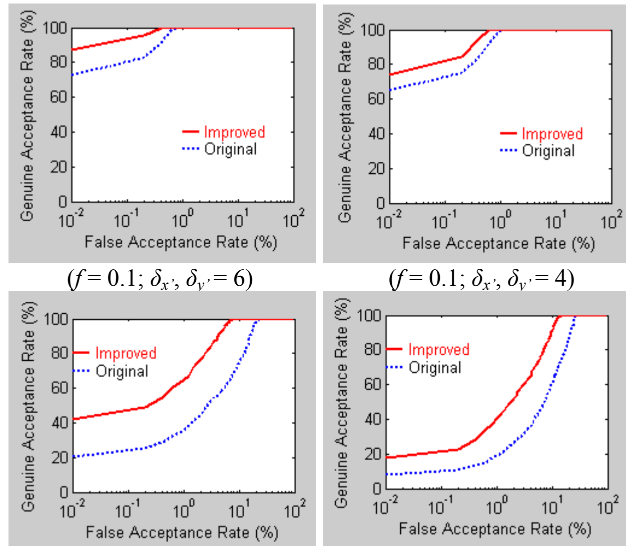
We then replace the FingerCode generation module with the new algorithm to compute the oriented FingerCode. The same dataset is again processed. Each test fingerprint image is matched with all the other fingerprints in the database. If the Euclidean distance between two FingerCodes is below a threshold, we make the decision that "the two fingerprint images are matched"; otherwise, we say, "the two fingerprint images come from different fingers". The matching is labeled correct if the matched pair is from the same finger and incorrect, otherwise. The genuine acceptance rate (GAR) and the false acceptance rate (FAR) corresponding to a distance threshold are computed according to these labeled matching results. The overall matching performance can be measured by a receiver operating characteristic (ROC) curve, which plots GAR against FAR at different operating points (distance thresholds). Each point on the curve corresponds to a special distance threshold. In the experiments, we use the Gabor filters with various f and $\delta_{x'}$, $\delta_{y'}$. Figure 5 illustrates the performance improvement of the new approach. The dashed line represents the performance of the original approach [5], while the solid line shows the performance of the new approach. From the results, we can see that the new approach outperforms the original approach over a wide range of FAR values, especially at low FAR values.

6. CONCLUSION

In this paper, we have presented an improved method for filterbank-based fingerprint matching, which utilize both the AAD features and the direction features available in the fingerprints. Experimental results obtained from a large fingerprint database (NIST-4) show that the addition of the direction features leads to a substantial improvement in the overall matching performance. Moreover, the proposed reference point location method is robust and rotation-invariant for fingerprint images.

7. ACKNOWLEDGMENT

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong SAR (Project no. CUHK 4190/01E).



($f = 0.1$; $\delta_{x'}$, $\delta_{y'} = 6$) ($f = 0.1$; $\delta_{x'}$, $\delta_{y'} = 4$)

($f = 0.05$; $\delta_{x'}$, $\delta_{y'} = 4$) ($f = 0.2$; $\delta_{x'}$, $\delta_{y'} = 4$)

Figure 5. ROC curves comparing the performance of the new FingerCode with the original FingerCode on the NIST-4 database.

8. REFERENCES

[1] A. K. Hrechak and J. A. Mchugh, "Automated fingerprint recognition using structural matching," *Pattern Recognition,* vol. 23, no. 8, pp. 893-904, 1990.

[2] A.K. Jain, L. Hong, S. Pankanti, and R. Bolle, " An identity authentication system using fingerprints", *Proc. IEEE*, Vol. 85, No. 9, pp. 1365-1388, 1997.

[3] A. K. Jain, S. Prabhakar, and L. Hong, " A multichannel approach to fingerprint classification," *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 21, no. 4, pp. 348-359, 1999.

[4] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "FingerCode: a filterbank for fingerprint representation and matching," *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition(CVPR),* vol. 2, pp. 187-193, 1999.

[5] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Trans. Image Processing*, vol. 9, no. 5, pp. 846-859, 2000.

[6] D. Maio and D. Maltoni, "Direct gray-scale minutiae detection in fingerprints," *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 19 no. 1 , pp. 27-40, 1997.

[7] N. Ratha, S. Chen, K. Karu, and A.K. Jain, " A real-time matching system for large fingerprint databases", *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 18, no. 8, pp. 799-813, 1996.