Improving Efficiency of Fingerprint Matching by Minutiae Indexing

Yangyang Zhang, Jie Tian*, Kai Cao, Peng Li, and Xin Yang
Institute of Automation, Chinese Academy of Sciences, Beijing 100080 China
{*zhangyy, tian*, caokai, lipeng, yx*}*@fingerpass.net.cn*

Abstract

This paper proposes a novel minutiae indexing method to speed up fingerprint matching, which narrows down the searching space of minutiae to reduce the expense of computation. An orderly sequence of features are extracted to describe each minutia and the indexing score is defined to select minutiae candidates from the query fingerprint for each minutia in the input fingerprint. The proposed method can be applied in both minutiae structure-based verification and fingerprint identification. Experiments are performed on a large-distorted fingerprint database (FVC2004 DB1) to approve the validity of the proposed method.

1. Introduction

The most popular features for fingerprint matching are minutiae, which generally outperform other fingerprint features in terms of the storage size and the discriminability. However, it is challenging to pair two sets of minutiae accurately and efficiently since there may exist severe interferential factors between them. First, there are several kinds of transformation between two impressions from the same sensor including linear transformation (translation, rotation) and non-linear distortion. Second, the quality of raw fingerprints is influenced by residual on the surface of sensor, skin condition and the capture environment, so the extraction of minutiae is likely to generate spurious minutiae or miss genuine ones.

The early minutiae-based algorithms [6] consider the single minutia described by its location, orientation and type as the element for matching. These methods usually compare any pair of minutiae from the input and query templates to determine the alignment parameters and similarity score with the computational complexity O(n^2)(n is the number of minutiae in a fingerprint). However, the single minutia is so sensitive to the noise that the performance of matching is greatly degraded for low-quality fingerprints. The subsequent researches constructed the relative minutiae structures (e.g. minutiae simplex [2], minutiae triplet [3]) to improve the accuracy. These local structures reinforced the relationship between neighboring minutiae by introducing rich relative features. Experimental results approved they have more reliable and robust performance in fingerprint alignment and minutiae paring. However, the improvement of accuracy inevitably brings out expensive expense of computation.

This paper proposes a novel minutiae indexing approach to speed up fingerprint matching. This method narrows down the searching space of minutiae to reduce the computing expense of minutiae structure-based verification. An orderly sequence of features are extracted to describe each minutia and the indexing score is defined to estimate represents the possibility that two minutiae are correspondent. Then the query minutiae for each input one is reduced to a limited number based on their indexing scores and transferred to the sequent matching, which greatly improves the efficiency of minutiae triplet-based verification.

Furthermore, we apply this method in fingerprint identification on large databases. Different from previous methods based on global features [5], [4], we extract two synthetical indexing variants from all the minutiae indexing results as the searching features, avoiding difficult alignment and performing better for large-distorted fingerprints.

The rest of the paper is organized as follows: Section 2 proposes the procedure of minutiae indexing with the local minutiae descriptor. Section 3 presents the application of minutiae indexing into minutiae triplet-based verification and fingerprint identification. Experiments are performed on FVC2004 DB1 and the results are displayed to approve its validity in section 4. Section 5 summarizes our researches.

2. Minutiae Indexing

Suppose $\{m_i = (x_i, y_i, o_i)\}_{i=1}^{n}$ are the collection of minutiae extracted from fingerprint skeleton, including their positions and orientations.

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

2.1. Indexing Features

The following features are extracted to describe each minutia m_i in offline stage of minutiae indexing. These features are selected according to three rules: 1) Capable of discriminating one minutiae; 2) Efficient to compare; 3) Robust to fingerprint noise.

- **Direction** Dr Compared to orientation, the direction of minutiae is extended to the range $[0, 2\pi)$. We determine Dr as the direction of the attached ridge or valley, which have this minutia as its ending point, as displayed in Fig. 1(a). This variant can be adjusted by its corresponding orientation since they approximately satisfy the constraint $Dr = o_i$ or $o_i + \pi$. The direction represents the geometric character of minutiae so it can be used to reduce the number of false correspondences.

- **Coherence** Ch The coherence indicates the distribution of orientation in a neighborhood block B around the minutia. It is evaluated by the covariance matrix J and the normalized coherence measure $Ch \in [0, 1]$ [1]. Let $g_s = (g_s^x, g_s^y)$ denote the gradient intensity at each point s in B, then,

$$J = \frac{1}{n^2} \sum_{s \in B} g_s g_s^T = \begin{bmatrix} j_{11} & j_{12} \\ j_{21} & j_{22} \end{bmatrix} \quad (1)$$

$$Ch = \frac{j_{11}^2 - j_{22}^2 + 4j_{12}^2}{(j_{11} + j_{22})^2} \quad (2)$$

- **Curvature** Cv The curvature is usually calculated by approaching the attached ridge and measuring the rate of change of the tangent at the minutia. In this method, both the associated ridge and valley are sampled once in a constant interval, extracting two neighborhood sampling points. Then the curvature can be estimated for termination and bifurcation in a common form. Let $Z_1 = x_1 + jy_1$ and $Z_2 = x_2 + jy_2$ be two vectors respectively connecting the minutia and two sampling points as shown in Fig. 1(b). If the ridge is approached using a cubic B-Spline, the curvature at the minutia is estimated as [7].

$$Cv = \frac{8(x_1 y_2 - x_2 y_1)}{[(x_1 - x_2)^2 + (y_1 - y_2)^2]^{\frac{3}{2}}} \quad (3)$$

- **Orientation Vector** $\overrightarrow{O}$ We also characterize each minutia with a vector that comprises information in neighborhood orientation field. We employ the block orientation for minutiae description instead of point orientation to decrease the effect of noise.

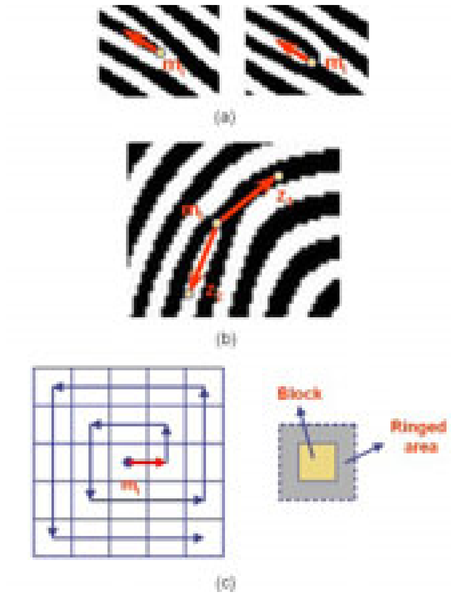

Figure 1. (a)Direction of termination and bifurcation; (b)Curvature; (c)Orientation vector, the red arrow denotes the direction of minutia, $K = 25$**.**

The region of interest centered at the minutia is partitioned into a lattice of blocks. In order to avoid the rotation and translation alignment, one edge of blocks is parallel with the direction of the minutia. The size of block is set as twice of the average fingerprint ridge width to obtain the best tradeoff between decreasing storage and preserving information. The vector $\overrightarrow{O} = \{o_k\}_{k=1}^{K}$ (o_k is the orientation of the k^{th} block, K is the number of the blocks) consists of the average orientation value of all these blocks in an order of tracing the arrows in Fig. 1(c). The novelty is that each o_k is attached with two binary bits b_k^1, b_k^2 to record if its corresponding block or its ringed area is minutia marked, this is, there exist minutiae in it. The two bits mark the distribution of minutiae among these blocks, which improves the discriminability of the feature. If both one block and the compared block (or its ringed area) are minutia-marked, they are considered minutia-corresponding. If one block is minutia-marked, but the compared block (and its ringed area) contain no minutiae, they are considered minutia-opposite. We introduce the ringed area to tolerate the displacement of corresponding minutiae due to distortion.

2.2. Indexing Score

According to Sec. 2.1, each extracted minutia consists of four fixed-length features, suitable for efficient comparison and similarity calculation. In order to minimize the average time of indexing, we construct a hierarchical structure for feature comparison. The more discriminating and efficient the feature is, the higher layer it is located on. Based on the experiments in Sec.4.1, a pair of minutiae are orderly compared by their direction, coherence, curvature and orientation vectors.

Let $P = \{M_i^p\}$ and $Q = \{M_j^q\}$ denote the minutiae collections extracted from the input and query fingerprints. The similarity between any pair of minutiae M_i^p and M_j^q is calculated by the following formula.

$$S_{i,j} = \begin{cases} 0 & \text{if } h(M_i^p, M_j^q) = 1 \\ exp(-\frac{D_{i,j}}{Thr_o}) & otherwise \end{cases} \quad (4)$$

$$D_o = \sqrt{[(\sum_k (w_k * sin\theta_k))^2 + (\sum_k (w_k * cos\theta_k))^2]/K'} \quad (5)$$

If and only if all the differences of their direction, curvature and coherence locate within the preset tolerances, $h(M_i^p, M_j^q) = 1$; $\theta_k = 2 * (o_{i,k}^p - o_{j,k}^q)$ if both of two block are in the fingerprint foreground, otherwise θ_k is invalid. K' denotes the number of valid θ_k. $D_{i,j}$ indicates the degree of correlation of two orientation vectors. If the orientation fields of such two vectors are very similar just with a rotation, all of θ_k tend to concentrate around a constant and the value of $D_{i,j}$ approaches 1, otherwise $D_{i,j}$ decreases. $Thr_o \in [0, 1]$ is determined by the distribution of D_o in training. If two blocks are minutiae-corresponding, $w_k = 1 + \varepsilon$ and if they are minutiae-opposite, $w_k = 1 - \varepsilon$, otherwise $w_k = 1$; $\varepsilon \in [0, 1]$.

The indexing score represents the possibility that two minutiae are correspondent based on not only their mutual similarity but the similarities to other minutiae in the opposing fingerprints. Formally, it is estimated as following:

$$I_{i,j} = S_{i,j}^2 / (\sum_{k \neq i} S_{k,j} + \sum_{k \neq j} S_{i,k}) \quad (6)$$

The score determines the priority of each minutia in the query list. For each input minutia, we sort the query minutiae in term of decreasing indexing score and hypothesize that the top M ones are valid in the sequent matching stage.

3. Application

Our minutiae indexing method is employed to not only speed up the minutiae structure-based 1:1 verification but to accelerate fingerprint searching in 1:N identification.

3.1. Minutiae Triplet-based verification

Minutiae triplet have been used for many matching algorithms since the relationship between the vertexes of triangle are proved effective to improve the matching accuracy. In these methods any pairs of triplets from the input and query fingerprints are compared in turn, the complexity of matching reaches O(n^6)(n is the number of minutiae) if the triplets are randomly constructed(C_n^3 triplets). In this application, minutiae indexing is employed to separate a subset of the query triplets as the candidates in sequent matching. The online procedure of minutiae triplet-based algorithm is modified with minutiae indexing to complete triplet pairing.

- For each minutia M_i^p, generate an indexing array to record the serial numbers of the top M query minutiae in the decreasing order of indexing score.
- For each minutiae triplet T_k^p, respectively select three different minutiae from three corresponding indexing arrays of its vertexes and construct a set of candidate triplets.
- Compare T_k^p with each of its candidate triplets. The reliability of their similarity is weighted by the sum of three indexing scores.

Without loss of generality, we take no account of the constraint during constructing of triplets. Based on minutiae indexing, the number of query triplets decreases from C_n^3 to C_M^3.

3.2. Fingerprint identification

Fingerprint indexing technique divides fingerprint database into a number of subsets to reduce the searching space. In this application, two indexing variants $Inum$ and $Iscore$ are derived from all the minutiae indexing results between the input and query fingerprints to indicate the posterior possibility of genuine matching. Suppose P is the input fingerprint and Q_i the query ones, $Inum$ is calculated as the sum of the number of the indexing minutiae pairs between P and Q_i while $Iscore$ is the average of the first indexing score for all the input minutiae. A linear fisher classifier is employed to determine whether Q_i should be retrieved into the sequent fine matching process.

4. Experiments

FVC2004 DB1 is selected for algorithm evaluation. This database consists of 800 fingerprints (100 fingers, 8 impressions per finger) captured at a resolution of about 500dpi. It emphasizes on distortion, dry and wet

Table 1. Performance of minutiae indexing with different features, $M = 10$.

	EER	FMR100	FMR1000	AvgT (s)
none	4.35	7.00	14.90	0.25
Dr	3.93	6.93	17.86	0.14
Ch	4.25	7.18	25.15	0.15
Cv	4.41	6.97	13.79	0.17
$\overrightarrow{O}$	3.63	6.83	24.54	0.09
all	3.69	7.04	24.72	0.034

fingerprints, some of which have no singularities in the valid capture areas. All the experiments are conducted on PC Pentinum-4 3.2 GHZ.

4.1. Effectiveness in Verification

To approve the efficacy of the hierarchical structure of indexing features, we evaluate the performance of minutiae triplet-based verification [3] in three situations: without minutiae indexing, with each single feature and with all combined features. The results in Tab. 1 confirm that minutiae indexing based on combined features can significantly improve the efficiency of minutiae structure-based matching. Minutiae indexing also meliorates the accuracy of verification since it prefers to compare two minutiae triplets constructed with the most similar M minutiae pairs. The parameter M (the limited number of candidates for each input minutia) is set 10.

4.2. Validity in Identification

The first impression of each finger is used as the query fingerprint in database while other 7 impressions are used as the input fingerprints. Totally 700 searching are performed. The performance is evaluated by the dependence of retrieval accuracy (the rate of retrieving the correct fingerprint) on penetration rate (the average portion of database retrieved) at all thresholds. The global features proposed by Liu [4] is implemented for comparison including the average ridge width and orientation field around singularities. Furthermore, we combine the indexing variants derived from both the global features and local minutiae descriptors with a weighted sum rule for fingerprint indexing. As shown in Fig. 2, the retrieval accuracy of minutiae-based fingerprint indexing is higher than Liu's features because the large distortion in FVC2004 DB1 results in far more degradation on the global features than on local minutiae descriptors. The results by fused features is better than any of them because two kinds of features are independent and compensate each other in discriminability.

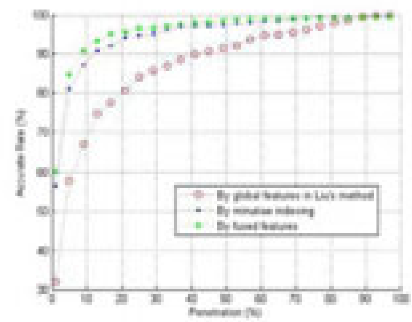

Figure 2. Performance of three fingerprint indexing methods on FVC2004 DB1.

5. Conclusion

A minutiae indexing method is proposed to speed up minutiae-based fingerprint matching. The proposed method has been applied to one minutiae triplet-based verification to approve its effectiveness. We also derive indexing variants from all the minutiae indexing results for fingerprint identification on the database. All the experiments are performed on the public domain database FVC2004 DB1. Compared to the fingerprint indexing based on the global features, our method is based on the local minutiae descriptors and performs more robustly for the large-distorted fingerprints.

Acknowledgments.

This paper is supported the Key Project of National Natural Science Foundation of China (No. 60575007).

References

[1] Y. Chen, S. Dass, and A. Jain. Fingerprint quality indices for predicting authentication performance. *AVBPA2005*, 3546:160–170, 2005.

[2] Y. He, J. Tian, L. Li, and X. Yang. Fingerprint matching based on global comprehensive similarity. *IEEE Trans. on PAMI*, 28(6):169–177, 2006.

[3] Z. M. Kovcs-Vajna. A fingerprint verification system based on triangular matching and dynamic time warping. *IEEE Trans. on PAMI*, 22(11):1266–1276, 2000.

[4] M. Liu, X. Jiang, and A. C. Kot. Efficient fingerprint search based on database clustering. *Pattern Recognition*, 40(6):1793–1803, June 2007.

[5] T. Liu, G. Zhu, C. Zhang, and P. Hao. Fingerprint indexing based on singular point correlation. *ICIP2005*, 3:293–296, 2005.

[6] X. Luo, J. Tian, and Y. Wu. A minutia matching algorithm in fingerprint verificatione. *ICPR2000*, 4:833–836, 2000.

[7] G. Medioni and Y. Yasumoto. Corner detection and curve representation using cubic b-splines. *Computer Vision, Graphics and Image Processing*, 39:267–278, 1987.