Fingerprint Matching based on Global Minutia Cylinder Code

Yuxuan Luo, Jianjiang Feng, Jie Zhou
Tsinghua National Laboratory for Information Science and Technology
Department of Automation, Tsinghua University, Beijing, China
luoyx12@mails.tsinghua.edu.cn, jfeng@tsinghua.edu.cn, jzhou@tsinghua.edu.cn

Abstract

Although minutia set based fingerprint matching algorithms have achieved good matching accuracy, developing a fingerprint recognition system that satisfies accuracy, efficiency and privacy requirements simultaneously remains a challenging problem. Fixed-length binary vector like IrisCode is considered to be an ideal representation to meet these requirements. However, existing fixed-length vector representations of fingerprints suffered from either low distinctiveness or misalignment problem. In this paper, we propose a discriminative fixed-length binary representation of fingerprints based on an extension of Minutia Cylinder Code. A machine learning based algorithm is proposed to mine reliable reference points to overcome the misalignment problem. Experimental results on public domain plain and rolled fingerprint databases demonstrate the effectiveness of the proposed approach.

1. Introduction

Fingerprint is one of the most widely used biometric traits because of its high matching accuracy and low cost and compact size of fingerprint sensors. Much researches have been done to improve both matching accuracy and speed of fingerprint recognition systems in the last four decades [14]. To achieve high accuracy, most state-of-the-art fingerprint matching algorithms are based on minutia set (i.e. ridge endings and bifurcations) [18] [7]. Because of the unordered and variable size nature, the correspondence between two minutia sets is built using local attributes and global geometry relations as the first step of these algorithms before a global matching score can be calculated. Such minutiae matching algorithms are difficult to adapt to retrieve fingerprints in large databases.

Minutia set representation not only makes fast matching complicated, but also makes template protection challenging. With the diffusion of biometric application, the privacy problem arises in recent years. The most dangerous attack on a biometric system is against the template stored in the database because the stolen template can be used to attack all biometric systems where the same biometric is enrolled [10]. A popular biometric template protection scheme is biometric cryptosystem, including fixed-length binary vector based scheme, like fuzzy commitment [12] and the point set based scheme, such as fuzzy vault [16]. Although minutia set is a more natural and popular representation of fingerprint, fuzzy vault scheme has high time and space complexity. Therefore fixed-length representation of a fingerprint gains increasing interests in the fingerprint community because it is suitable for fast matching and template protection at the same time [8], [5], [6], [21], [20], [19], [15].

There have been many researches on fixed-length feature vector representation of fingerprint. Jain et al. proposed FingerCode that uses filterbank response to describe the ridge features of fingerprint [11]. Cappelli created a 110 dimensional feature vector using ridge orientation and frequency [6]. Without using any minutiae information, the distinctiveness of these representations is limited. Fixed-length feature vectors extracted from minutia sets are also studied. Farooq et al. suggested the histogram of minutiae triplets [8]. Bringer et al. proposed a pure local descriptor based on minutiae vicinities and converted it to fixed-length form by measuring descriptor's similarity with representative descriptors [5]. Unfortunately, the accuracy of these methods is still unsatisfactory. Spectral minutiae representation method proposed in [21] and its improved and binarized version [20][19] use Fourier transform under polar coordinates to convert a minutia set to a spectral representation which is invariant to translation. Nandakumar used phase spectrum instead of amplitude [15]. To sum up, current fixed-length representations of fingerprint can be classified into two types: alignment based and alignment free and Table 1 lists them for comparison. Alignment free methods tend to be less accuracy since they discard important geometric information. In alignment based methods, algorithms using ridge orientation and frequency features do not have enough discriminative ability and the ones using minutiae information suffer from misalignment.

In this paper, we propose a new fixed-length feature vec-

Algorithm	Feature	Alignment Method	Experiment
Jain et al. [11]	Filterbank response	Loop	Verification
Cappelli [6]	Ridge orientation and frequency	Singular point, focal point	Retrieval
Xu et al. [21][20][19]	Amplitude spectrum of minutiae	Singular point	Verification
Nandakumar [15]	Phase spectrum of minutiae	Focal point from High Curvature Points	Verification
Farooq et al. [8]	Histogram of minutiae triplets	Alignment Free	Verification
Bringer et al. [5]	Bag of words	Alignment Free	Verification

Table 1. Representative fixed-length fingerprint representations proposed in the literature

tor representation for fingerprint. To overcome the problems above, a state-of-the-art minutiae descriptor, Minutia Cylinder Code (MCC) [7], is extended to capture discriminative information in fingerprints; misalignment is handled by the fusion of traditional reference points and a new type of reference points mined by a learning algorithm.

The rest of this paper is organized as follows: Section 2 reviews the main idea of MCC and introduces its global version. Section 3 introduces traditional reference points (RPs) we adopt and proposes our algorithm to mine additional reliable RPs. In Section 4, experiments are done both in verification and retrieval to demonstrate our algorithm's effectiveness. Finally, Section 5 concludes our work and puts foreword some future work that can be done.

2. Global Minutia Cylinder Code (GMCC)

Minutia Cylinder Code [7] is a well-known local descriptor that has shown state-of-the-art performance [9]. Here we show it can also be used as a global descriptor of fingerprints if combined with reliable reference points. Section 2.1 reviews the basic idea of MCC for completeness. Section 2.2 presents the detail of the proposed GMCC. The algorithm that uses multiple RPs for better performance is discussed in Section 2.3.

2.1. Minutia Cylinder Code (MCC)

The Minutia Cylinder Code (MCC) of a minutiae (referred to as central minutia) records spatial and directional relationships between the central minutia and its neighbors in the form of a cylinder, whose base and height are related to spatial and directional information. The cylinder is divided into sections along height and sections are further split in to cells. Each *valid* cell is assigned a value which reflects the density of minutiae at that location and direction. Please refer to [7] for more details.

The floating point MCC representation can be easily converted to a fixed-length binary vector consists of a mask of *valid* cells and the binarized cell values by comparing to a threshold.

Parameter(s)	Rolled	Plain
R	**200**	**140**
σ_s	**12**	$\frac{20}{3}$
N_s	16	16
N_d	6	6
Ω	50	50
$\min_{VC}$	**0.25**	**0.25**
$\min_{M}$	2	2
$\min_{ME}$	**0.2**	**0.2**
μ_p	**0.007**	**0.007**
τ_p	**500**	**500**

Table 2. Parameter values for GMCC (numbers in bold indicate parameters different from original MCC). See [7] for the meaning of each parameter.

2.2. Global Minutia Cylinder Code (GMCC)

The fixed length of MCC makes computing similarity between two minutiae very efficient. However, a fingerprint is represented as an unordered set of MCCs [11]. Computing the similarity between two fingerprints represented as sets of MCCs is still complicated. In this paper, we further exploit its global ability. Given a location with direction in fingerprint image as RP (just like the central minutia), a GMCC that covers much larger region than the original one can be used to encode the entire minutia set (see Fig.1). Similarity of two fingerprints can be computed exactly as the original MCC [7] without any additional global consolidation step. Some parameters are optimized because GMCC should tolerate larger distortion than MCC. New parameter values are reported in Table 2 and the changed ones are highlighted in bold. To make this strategy feasible, a reference point (RP) represented as $\{x, y, \theta, type\}$ that can be stably detected in fingerprints should be defined, where x, y, θ denote the location and direction of the RP and $type$ is defined for situation where multiple types of reference points are used. Next, we will introduce the matching strategy under multiple RPs condition in Section 2.3.The RP detection algorithm will be described in Section 3.

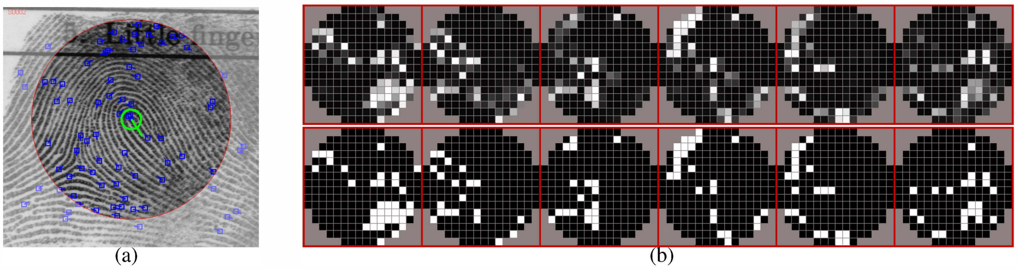

Figure 1. (a) A fingerprint from NIST4. The green circle indicates one of its reference points and the red circle indicates the region of interest. (b) The corresponding floating-point GMCC (first row) and binary GMCC (second row).

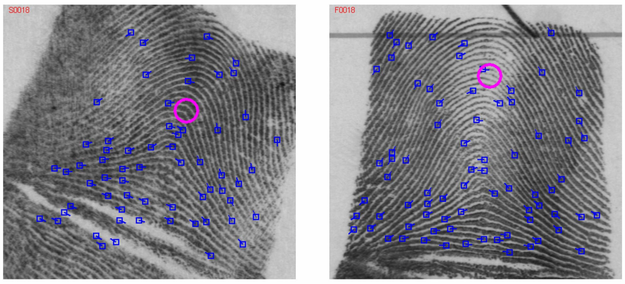

Figure 2. A pair of corresponding fingerprints for which neither singular points nor focal point can be used as good RP. No singular points are present and global skin distortion makes the detected focal points (marked as circle) very inconsistent.

2.3. Matching with GMCC

Suppose we have two fingerprints S and F to be matched. Two sets of reference points $R^S = \{R_1^S, R_2^S, \cdots\}$ and $R^F = \{R_1^F, R_2^F, \cdots\}$ have been detected in the two fingerprints. For each R_i^S, nine locations around it are sampled for robustness and then GCS_i^S that contains 9 GMCCs are extracted (interval is set to 16 pixels along each direction in experiments); for each R_j^F one GMCC is extracted as the only element of GCS_i^F. The difference between the number of descriptors in search and file side is for low memory cost in searching large databases. Denote the maximum matching score between GCS_i^S and GCS_j^F as s_{ij}, which represents the similarity of two GMCCs based on $R_i^S$and R_j^F, the final score between two fingerprints is calculated as:

$$s = \max_{\text{Matchable } R_i^S, R_j^F} s_{ij}.$$

The attribute *type* (delta, loop, focal point, or specific type of template point, see Section 3) of a RP is used here to decide whether R_i^S and R_j^F can be matched. Two RPs are said to be matchable if and only if they have the same type and the rotation between them is below 60 degrees.

3. Reference Point

Reference point detection is a critical step in alignment based fingerprint matching algorithm and any error in this step can cause matching failure. Therefore, to improve matching performance, three kinds of reference points are used in this paper. We first review two kinds of traditional reference points in section 3.1 and propose a new reference point called template point mining and detection algorithm in section 3.2. Section 3.3 describes how to estimate the direction of focal point.

3.1. Singular Point and Focal Point

Singular point (SP) and focal point (FP) [17] are two kinds of stable RPs and achieve good performance in the fingerprint retrieval algorithm described in [6]. The Poincaré index based approach is used to detect loop and delta type singular points [13] and the approach in [17] is used to detect the focal point of a fingerprint.The singular point can be accurately localized in good quality fingerprints if present, but cannot be detected in fingerprints of plain arch type, and cannot be reliably detected in poor quality fingerprints. The focal point is relatively robust to noise but sensitive to skin distortion (see Fig. 2). To overcome these issues, we propose a learning based algorithm to mine additional good orientation templates for fingerprint registration (Section 3.2). The direction of a singular point is estimated using the approach in [4] and the direction of a focal point is estimated using orientation templates learned by the proposed algorithm (Section 3.3).

3.2. Template Point

Template matching is a classical technique for object detection in images. Given a template T and a detection image D, it locates the center and direction under which T and D are most similar. For detecting reliable reference points in fingerprints, ridge orientation field (in the range of $[0, \pi)$) is a better feature representation than the original grayscale image because of its tolerance to low image quality and fingerprint distortion. Thus the following discussion is based on the orientation field. Given two orientation patches A and B, each consisting of $h \times h$ orientation elements, equation (1) gives the definition of similarity between them:

$$\mathrm{OrientSimi}(A, B) = \frac{1}{h^2} \sum_i \cos(2A(i) - 2B(i)). \quad (1)$$

Double angle are adopted because θ and $\theta + \pi$ have exactly the same meaning in orientation representation. When detecting RP using template matching technique, we first compute the similarity of T and D on each possible location and direction and the detection result is:

$$\{x^*, y^*, \theta^*\} = \arg\max_{x,y,\theta} \mathrm{OrientSimi}(T_\theta, D_{x,y}),$$

where T_θ means the rotation version of T, and $D_{x,y}$ means local orientation patch centralized at location (x, y). In this paper, we use a 15 by 15 orientation field template with a sample interval of 8 pixels. Fig. 3 depicts the detection procedure.

Instead of empirically specifying orientation templates, we choose to mine good templates using training fingerprints. A proper template should have the following two characteristics:

(1) It should be stable across difference images from the same finger.

(2) It should be complementary to singular point and focal point.

Following these intuitions, we formulate a supervised learning procedure. Given a set of search fingerprints $\boldsymbol{S} = \{S_1, S_2, \cdots, S_m\}$, a set of file fingerprints $\boldsymbol{F} = \{F_1, F_2, \cdots, F_n\}$ and their correspondence information[1], the template set $\boldsymbol{TS}$ satisfying our requirements is then generated automatically.

Because spatial transformation parameters between matching fingerprints are not available, the first objective cannot be measured by evaluating the displacement of reference points in matching fingerprints. Instead, we use fingerprint retrieval performance as the indirect objective function. Given a candidate template p and the Training Set $\{\boldsymbol{S}, \boldsymbol{F}\}$, the score of p is defined as the number of fingerprints from $\boldsymbol{S}$ that can be solved by using it. A search fingerprint is called *solved* if its corresponding file fingerprint can be retrieved at high rank (within top 1% of the whole file fingerprint set). The best template is the one that has the highest score over all candidate templates $\boldsymbol{P}$, which consists of sampled ridge orientation patches extracted from all the fingerprints in $\boldsymbol{S}$. To accelerate the training process, the patches that have a coherence higher than a predefined threshold (0.9 in all experiments) are ignored because of their obviously low locating ability. The coherence of an orientation patch p is defined as [4]:

$$\mathrm{coherence}(p) = \frac{|\langle \sum_i \cos(2p(i)), \sum_i \sin(2p(i)) \rangle|}{\sum_i |\langle \cos(2p(i)), \sin(2p(i)) \rangle|},$$

where $p(i)$ denotes the ith orientation element of patch p. The patches that have a high similarity above 0.9 measured by equation (1) with any existing candidate are also skipped. To learn multiple templates, we remove the solved fingerprints from $\boldsymbol{S}$ and select the next one until no more fingerprint can be solved or all fingerprints have been solved. The complete procedure Template Mining is outlined in Algorithm 1.

Algorithm 1 Template Mining

```
Input:
    Search Fingerprint Set $\boldsymbol{S} = \{S_1, S_2, \cdots, S_m\}$
    File Fingerprint Set $\boldsymbol{F} = \{F_1, F_2, \cdots, F_n\}$
Output: Selected Template Set $\boldsymbol{TS}$
Generate candidate templates $\boldsymbol{P} = \{p_1, p_2, \cdots\}$
$\boldsymbol{TS} \leftarrow \emptyset$
repeat
    for $i = 0$ to $\#\boldsymbol{P}$ do
        Detect RPs in $\boldsymbol{S}$ and $\boldsymbol{F}$
        Generate GMCCs for $\boldsymbol{S}$ and $\boldsymbol{F}$ using RPs
        Retrieve using RPs and GMCCs
        $score(p_i) \leftarrow$ # solved fingerprints from $\boldsymbol{S}$
        $E_i \leftarrow$ solved fingerprints from $\boldsymbol{S}$
    end for
    $i^* = \arg\max_i score(p_i)$
    $\boldsymbol{TS} \leftarrow \boldsymbol{TS} \cup p_{i^*}$
    $\boldsymbol{P} \leftarrow \boldsymbol{P} \backslash p_{i^*}$
    $\boldsymbol{S} \leftarrow \boldsymbol{S} \backslash E_{i^*}$
until no more fingerprint can be solved
```

The second objective is needed for the system's overall performance and can be achieved by elaborately selecting of the training set, which will be discussed in detail in Section 4.1.

3.3. Refine Focal Point's Direction

The direction of focal point is estimated by comparing with a handcrafted orientation template in [3]. In this paper, we adopt the strategy described in Section 3.2 for mining such templates. The procedure is almost the same as Algorithm 1 except that:

(1) Instead of generating candidates from all positions in fingerprints, we use patches that are centralized at detected focal point using the method in [17].

[1] We assume (1) no any two search fingerprints are from the same finger; (2) each search fingerprint has one and only one corresponding file fingerprint; (3) the file fingerprint set F may contain background fingerprints which have no corresponding search fingerprints (namely, $n > m$). This is the case of NIST4 and NIST14 databases.

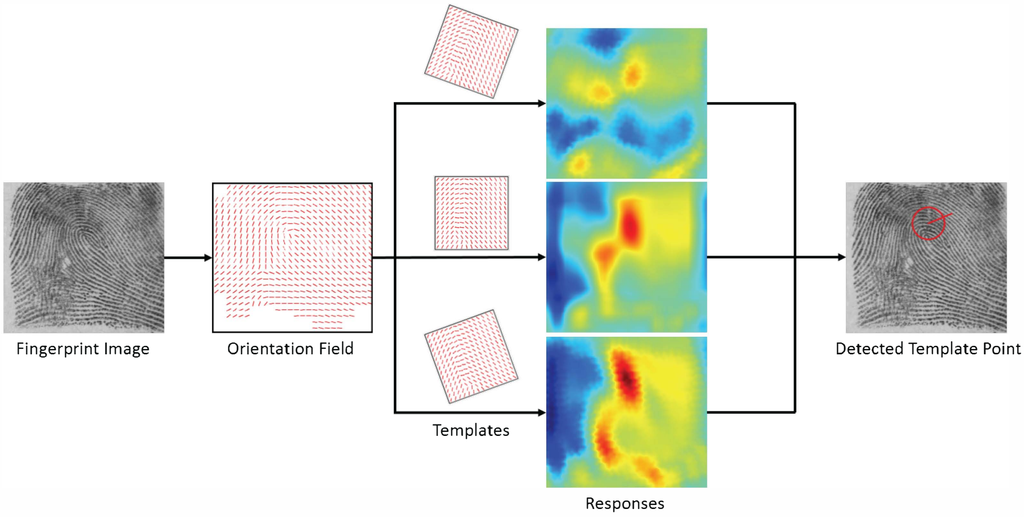

Figure 3. Procedure of estimating the location and direction of template point by template matching. A fingerprint is firstly converted to orientation representation. Then the response images of the orientation field for multiple rotated versions of orientation template are calculated in a sliding window manner. The forth column shows the responses of each template on orientation field (dark red indicates high response). The red circle with direction in the resulting image shows the location and direction under which the response is maximum.

(2) Reference point detection results using different candidate templates share the same position but have different directions.

4. Experiments

To demonstrate the effectiveness of our algorithm, we do both verification experiment on FVC2002 and retrieval experiments on NIST4 and NIST14. Compared with state-of-the-art fixed-length representation methods for verification and retrieval, our algorithm achieves better results.

4.1. Implementation Details

We use the first 1000 pairs of fingerprints from NIST14 as training set. Since template point is designed to be complementary with singular point and focal point, we set $\boldsymbol{S}$ as the fingerprints that no singular point is detected on either search fingerprint or corresponding file fingerprint. All the 1000 file fingerprints compose set $\boldsymbol{F}$. For mining templates for estimating direction of focal point, all the 1000 pairs of fingerprints are used as $\boldsymbol{S}$ and $\boldsymbol{F}$ respectively. The candidate template set $\boldsymbol{P}$ has a size of about 1500 and 1000 when mining the templates for the template point and for focal point respectively. The first two best templates for template point and focal point are shown in Fig. 4.

Considering the time consumption in detection phase and generalization performance, not all the mined templates are used. We reserve only the first two templates because they have solved most fingerprints with reasonable quality.

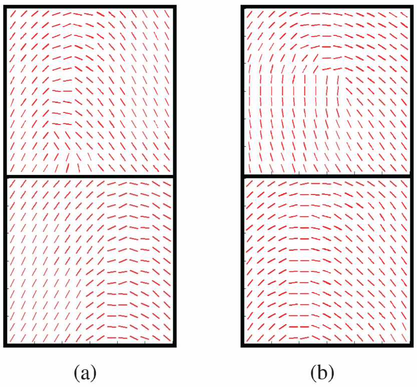

Figure 4. Learned orientation templates for (a) Template point detection and (b) Focal point direction estimation.

4.2. Verification

The proposed algorithm has been evaluated on FVC2002 DB2a fingerprint database. Minutiae are extracted using VeriFinger SDK 6.2 [2] and delta point is not used because it is usually not available in plain fingerprints. The state-of-the-art performance of fixed-length binary feature vector representation on this database is reported in [19]. To make a fair comparison with the method proposed in [19], four samples (samples 1, 2, 7, 8) of each finger are used in this experiment. We follow the evaluation protocol of FVC to generate all 600 genuine scores and 4950 imposter scores and the performance is shown in Fig. 5. We can observe that the performance of the proposed algorithm is better than binary spectral minutiae scheme [19]. Note that the differ-

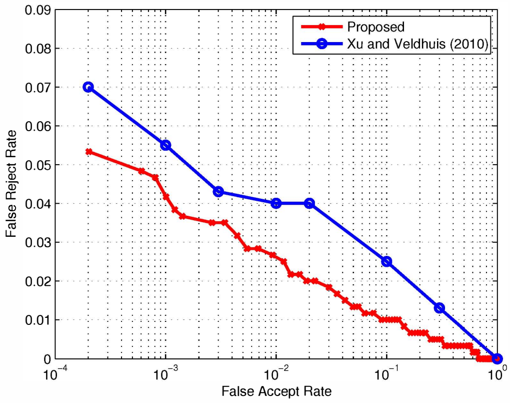

Figure 5. Verification performances of the proposed algorithm and the state of the art algorithm [19] on FVC2002 DB2a.

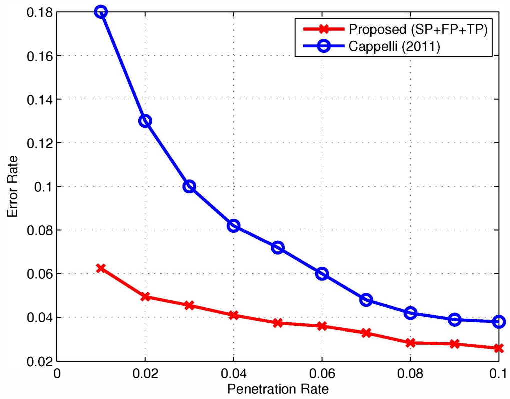

Figure 6. Retrieval performances of the proposed algorithm and the state of the art algorithm [6] on NIST4.

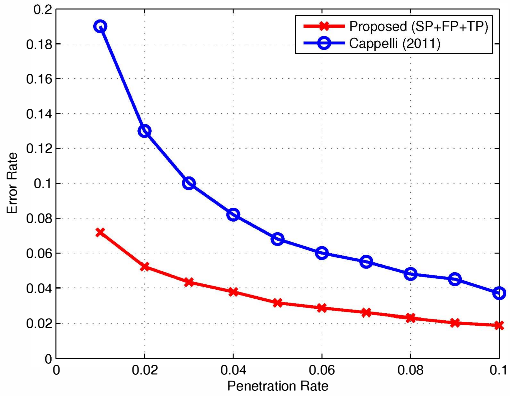

Figure 7. Retrieval performances of the proposed algorithm and the state of the art algorithm [6] on NIST14.

ence in matching accuracy cannot be solely attributed to the matching algorithm, since feature extraction algorithms for minutiae and singular points may also have a large impact on the matching performance.

4.3. Retrieval

Major advantage of fixed-length binary representation of a fingerprint is it's convenient to be scaled to large dataset. Fingerprint retrieval experiments are done on the last 2,700 pairs (to be consistent with previous research [6]) of fingerprints from NIST14 and all 2,000 pairs of fingerprints from NIST4. Fig. 6 and Fig. 7 show the tradeoff between penetration and error rate on the two datasets. The state-of-the-art performance of fixed-length feature vector based fingerprint retrieval algorithm [6] is cited for comparison. We can observe that the error rate of the proposed algorithm is 50% lower than that of the algorithm in [6] on both datasets. The superiority of the proposed algorithm can be attributed to more distinctive representation (we use minutiae information while [6] used ridge orientation and frequency information) and more robust reference point detection.

Fig. 8 shows the detection results of different kinds of RPs and Table 3 gives the error rates of different combinations of RPs under 1% penetration rate. Using only one kind of RPs, singular points achieve the lowest error rate on NIST14 and are less accurate than template points on NIST4, which contains uniform numbers of fingerprints of four fingerprint pattern types. Combination of singular points and template points outperforms other combinations that consist of two kinds of RPs. Attribute to the complementarity of different RPs, combining them together further improves performance on both datasets. Image quality has large impact on the performance. The error rate of the 1929 out of 2700 pairs of fingerprints from NIST14 and of the 1634 out of 2000 pairs of fingerprints from NIST4 with quality given by NBIS4.1.0 [1] better than 4 is also shown in Table 3. Fig. 9 shows the statistics of the distribution of the number percentages of different kinds of RPs based on which GMCC pair has the maximum score, it reveals that template points present better performance on plain arch and tented arch type fingerprints. An example where the template point provides more accurate registration between corresponding fingerprints than singular point does is shown in Fig. 10.

4.4. Computational Cost

The speed of the proposed matching algorithm is measured on NIST4. Implemented in C++, our matching algorithm can do about 100,000 matches per second on a standard PC (2.5 GHz CPU, single core), which is much faster than typical fingerprint matching algorithms. The speed of feature extraction is not measured because it is implemented as several different modules and in different languages.

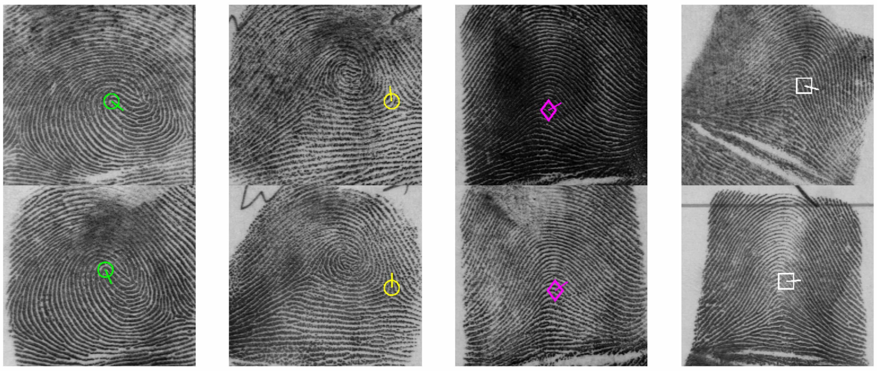

Figure 8. Examples of detected reference points. Circle indicates singular point, diamond indicates focal point and rectangle indicates template point. The reference point pairs with highest matching scores are shown.

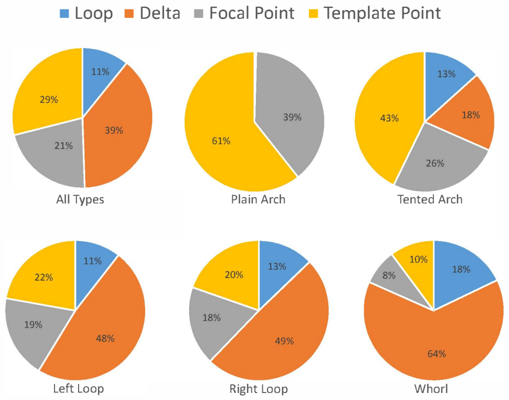

Figure 9. Distribution of the number percentages of four types of RPs based on which GMCC pair has maximum similarity. Statistics are obtained based on all 2,000 genuine pairs in NIST4 and six pie charts show the statistics over entire dataset and five types of fingerprints respectively.

5. Conclusions

Fixed-length binary vector representation for fingerprints is desired for indexing large databases and implementing template protection. With such representation, hamming distance can be used to measure the distance between fingerprints and biometric cryptosystem can be implemented conveniently. In this paper, we extend Minutia Cylinder Code, which was designed for describing local fingerprint region, to encode the whole minutia set of a fingerprint as a single binary vector. To address the challenging registration problem, we proposed a learning algorithm to mine good reference points, which are complementary with singular point and focal point for fingerprint registration purpose. Verification and retrieval experiments on public domain databases demonstrated the effectiveness of our algorithm. Noticing the large performance margin between fixed-length and general methods, how to combine image quality information with our framework to further improve the accuracy will be researched in the future.

Reference Points	Error Rate on NIST4	Error Rate on NIST14
SP	0.372	0.125
FP	0.224	0.544
TP	0.143	0.227
SP+FP	0.132	0.094
SP+TP	0.075	0.077
FP+TP	0.081	0.176
SP+FP+TP	0.063	0.072
(SP+FP+TP)*	0.026	0.046

Table 3. Retrieval performance of different combinations of RPs (SP: singular point, FP: focal point, TP: template point). The second and third columns show the error rates at 1% penetration rate. The last row indicates the error rate of the pairs of fingerprints whose quality given by NBIS4.1.0 [1] are both better than 4.

6. Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 61225008, 61373074 and 61020106004, the National Basic Research Program of China under Grant 2014CB349304, the Ministry of Education of China under Grant 20120002110033, and the Tsinghua University Initiative Scientific Research Program.

References

[1] National Institute of Standards and Technology, NBIS, http://www.nist.gov/itl/iad/ig/nbis.cfm.

[2] Neurotechnology Inc. VeriFinger SDK. http://www.neurotechnology.com/verifinger.html.

[3] V. Areekul, K. Suppasriwasuseth, and S. Jirachawang. The new focal point localization algorithm for fingerprint regis-

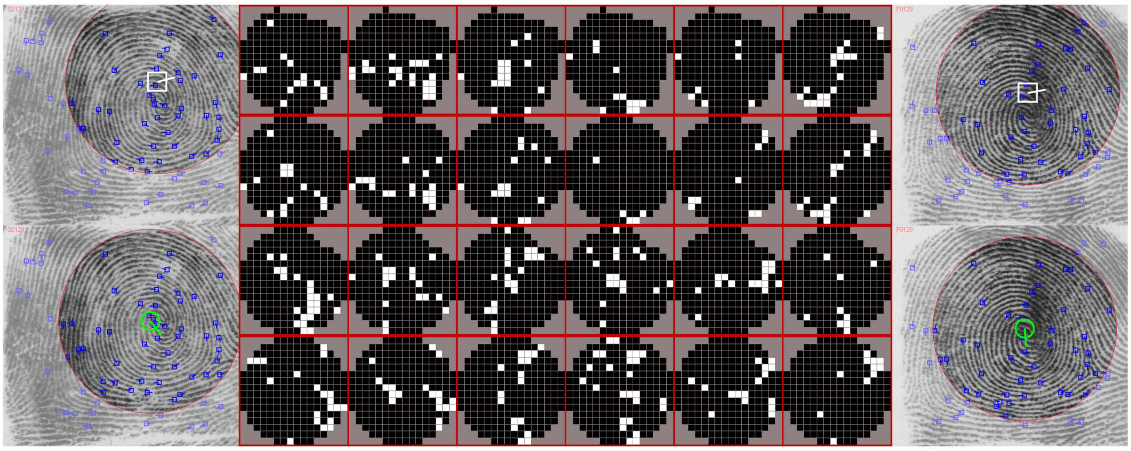

Figure 10. A pair of corresponding fingerprints in NIST4 and their GMCCs based on two types of RP. Region of interest is highlighted in each fingerprint image. The first and second row are GMCCs of the search and the file fingerprint based on corresponding template points. The third and fourth row are GMCCs based on corresponding singular points. Although the quality of both fingerprints is good, the inconsistent direction of singular points causes misalignment of GMCC. The matching score of GMCCs based on template point is 0.61, while the matching score of GMCCs based on singular point is only 0.14.

tration. In *Proceedings of the 18th International Conference on Pattern Recognition*, volume 4, pages 497–500, 2006.

[4] A. M. Bazen and S. H. Gerez. Systematic methods for the computation of the directional fields and singular points of fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):905–919, 2002.

[5] J. Bringer and V. Despiegel. Binary feature vector fingerprint representation from minutiae vicinities. In *Proceedings of the 4th IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS)*, pages 1–6, 2010.

[6] R. Cappelli. Fast and accurate fingerprint indexing based on ridge orientation and frequency. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 41(6):1511–1521, 2011.

[7] R. Cappelli, M. Ferrara, and D. Maltoni. Minutia cylinder-code: A new representation and matching technique for fingerprint recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2128–2141, 2010.

[8] F. Farooq, R. M. Bolle, T.-Y. Jea, and N. Ratha. Anonymous and revocable fingerprint recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2007.

[9] J. Feng and J. Zhou. A performance evaluation of fingerprint minutia descriptors. In *Proceedings of the IEEE International Conference on Hand-Based Biometrics (ICHB)*, pages 1–6, 2011.

[10] A. K. Jain, K. Nandakumar, and A. Nagar. Biometric template security. *EURASIP Journal on Advances in Signal Processing*, 2008.

[11] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti. Fingercode: a filterbank for fingerprint representation and matching. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, 1999.

[12] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *Proceedings of the 6th ACM conference on Computer and communications security*, pages 28–36, 1999.

[13] K. Karu and A. K. Jain. Fingerprint classification. *Pattern Recognition*, 29(3):389–404, 1996.

[14] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. *Handbook of fingerprint recognition (2nd edition)*. Springer, 2009.

[15] K. Nandakumar. A fingerprint cryptosystem based on minutiae phase spectrum. In *Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2010.

[16] K. Nandakumar, A. K. Jain, and S. Pankanti. Fingerprint-based fuzzy vault: Implementation and performance. *IEEE Transactions on Information Forensics and Security*, 2(4):744–757, 2007.

[17] K. Rerkrai and V. Areekul. A new reference point for fingerprint recognition. In *Proceedings of the International Conference on Image Processing*, volume 2, pages 499–502, 2000.

[18] M. Tico and P. Kuosmanen. Fingerprint matching using an orientation-based minutia descriptor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):1009–1014, 2003.

[19] H. Xu and R. N. Veldhuis. Binary representations of fingerprint spectral minutiae features. In *Proceedings of the 20th International Conference on Pattern Recognition*, pages 1212–1216, 2010.

[20] H. Xu and R. N. Veldhuis. Complex spectral minutiae representation for fingerprint recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop on Biometrics*, pages 1–8, 2010.

[21] H. Xu, R. N. Veldhuis, A. M. Bazen, T. A. Kevenaar, T. A. Akkermans, and B. Gokberk. Fingerprint verification using spectral minutiae representations. *IEEE Transactions on Information Forensics and Security*, 4(3):397–409, 2009.