

RESEARCH

Open Access

# A framework of multitemplate ensemble for fingerprint verification

Yilong Yin\*, Yanbin Ning, Chunxiao Ren and Li Liu

## Abstract

How to improve performance of an automatic fingerprint verification system (AFVS) is always a big challenge in biometric verification field. Recently, it becomes popular to improve the performance of AFVS using ensemble learning approach to fuse related information of fingerprints. In this article, we propose a novel framework of fingerprint verification which is based on the multitemplate ensemble method. This framework is consisted of three stages. In the first stage, enrollment stage, we adopt an effective template selection method to select those fingerprints which best represent a finger, and then, a polyhedron is created by the matching results of multiple template fingerprints and a virtual centroid of the polyhedron is given. In the second stage, verification stage, we measure the distance between the centroid of the polyhedron and a query image. In the final stage, a fusion rule is used to choose a proper distance from a distance set. The experimental results on the FVC2004 database prove the improvement on the effectiveness of the new framework in fingerprint verification. With a minutiae-based matching method, the average EER of four databases in FVC2004 drops from 10.85 to 0.88, and with a ridge-based matching method, the average EER of these four databases also decreases from 14.58 to 2.51.

**Keywords:** fingerprint verification, multi-template ensemble, fusion rule, establish polyhedron

## 1. Introduction

Researchers never stop to improve the performance of a biometrics system pursuing the lower equal-error rate (EER). Major approaches of reducing the EER can be divided into the following two categories:

(1) Improving the performance of process steps of a biometrics system. These steps include segmentation [1], enhancement [2], extraction [3], matching [4], etc. However, there are some problems in this method. For example, the room of performance increasing is limited.

(2) Fusing multiple sources of biometrics to increase the overall performance of a biometrics system. These sources include multiple sensors, multiple features [5], multiple matchers [6], multiple fingers [7], multiple impressions of a same finger [8], etc. Recent research results show that the most effective method to improve the performance of a biometrics system is to fuse more biometric information using ensemble learning [9]. These ensemble approaches, particularly these ensemble approaches with multiple matching algorithms, need

more computing resources and more storage. Ensembles of multiple sensors and multiple biometric verifications also need various kinds of sensors. Furthermore, it is very inconvenient for users since those multiple biometric verification ensembles need to capture various feature information from users in enrollment stage and verification stage.

Currently, multiple templates' ensemble is widely used in biometrics systems. In practice, multiple fingerprint images are captured and stored in database for one finger. These fingerprint images are called multiple templates. In current multiple templates ensemble researches, there are two challenges: (1) how to choose the proper templates for ensemble; (2) how to use the multiple templates information effectively.

There are a few studies have been done to deal with the problem of template selection to solve the first challenge. Uludag et al. [10] proposed two typical methods for automatic template selection: the first one, DEND, employs a hierarchical clustering strategy to choose a template set that could be best represents the intra-class variations. The second method, MDIST, selects a template set which exhibits maximum similarity with the

\* Correspondence: [ylyin@sdu.edu.cn](mailto:ylyin@sdu.edu.cn)  
School of Computer Science and Technology, Shandong University, Jinan  
250101, China

other fingerprints. The MDIST achieves better performance comparing with DEND in Uludag et al.'s study [10]. Lumini and Nanni [11] presented another clustering method which automatically selected the number of clusters. This method could also save memory and computational cost for a verification task. Multiple fingerprint images of a finger are acquired in order to obtain images of different regions of the finger [9]. So, when we select templates, the "ideal" templates should have these advantages: (1) The difference of these templates is big enough; (2) These templates are partially overlapping images. "Ideal" templates are shown in Figure 1.

For the second challenge, there are two major methodologies to use multi-template ensemble in fingerprint field: Mosaicking and Score level fusion. With mosaic [12,13], a larger fingerprint image could be obtained from several small images. But, the major problem in creating a mosaicked image is that the alignment different impressions/pieces cannot be completely recovered. Meanwhile, with the score level fusion [9,14,15], a query fingerprint has some matching scores with the templates. So, the final score is to fuse these scores with different weights. However, these weights are difficult to be determined in practice.

In this article, a framework of multitemplate ensemble for fingerprint verification is proposed. As mentioned above, in the enrollment stage, some fingerprint images are chosen and stored in database as fingerprint templates. And then, a polyhedron is created by the matching results of multiple template fingerprints and a virtual centroid of the polyhedron is given. The matching scores are also stored in the database. During the verification stage, a distance is calculated from a query fingerprint to the centroid. We add the distance into the set which is constituted by the distance between the query and templates. Finally, the framework returns a proper distance from the set as the final score of the query image and the template fingerprints. The

experimental results in FVC2004 show the effectiveness and robustness of the novel framework.

This article is a significant extension from the conference version which is published in [16]. The rest of this article is organized as follows. Section 2 describes the flowchart of the framework in detail and introduces various parts of the framework detailed. Section 3 introduces two relative fingerprint matching algorithms which will be as the base matcher. Section 4 gives out the experimental results. Conclusion and future study are given in Section 5.

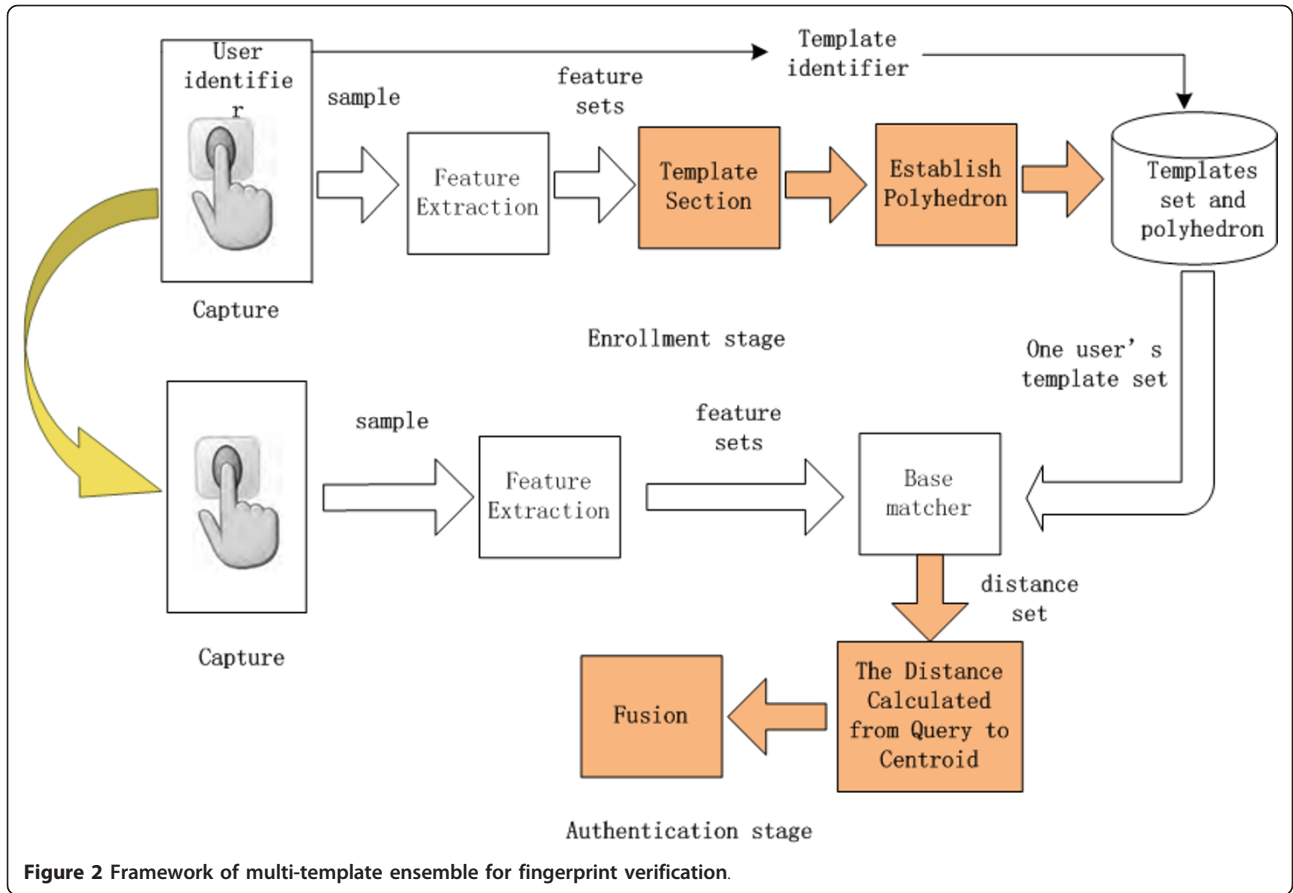
## 2. The proposed framework

A verification system includes enrollment and verification processes. The proposed framework of multitemplate ensemble also consists of the two processes. First, in the enrollment stage, some fingerprint images of the same finger are enrolled, and a template selection method is used to choose some fingerprints which are the best represent of this finger as the templates. Then, we will establish a polyhedron using the templates and get a virtual centroid of the polyhedron. The templates and the polyhedron will be stored in the database. Second, in the verification stage, a new polyhedron is established using the query and the templates fingerprint, and then a distance from the query to the centroid is calculated. Finally, a fusion rule will be used to choose a proper distance from a distance set which contains these distances between the query fingerprint and the templates and the distance between the query fingerprint and the centroid as the final score. The structure of the framework is shown in Figure 2.

As shown in Figure 2, the orange square is depicted in particular. In enrollment stage, when selecting templates, the number of templates is set beforehand. In this article, taking resources of computing and storing consideration, we prefer to set the number as 3. In database, we just store the feature sets of the templates and



Figure 1 "Ideal" templates.



the scores among the templates. The distance describes the similarity of two fingerprints, if the two fingerprints are more similar, then the distance is shorter. Otherwise, the distance is longer. The remaining will describe each part of the framework detailed.

### 2.1. Enrollment stage

In this section, the template selection and the polyhedron establishment will be introduced in detail. Most systems store multiple templates of the same finger in order to represent the finger better, but when the number of templates is larger, the resource of computing and storing is needed more. While, template selection is an effective method to reduce the number of fingerprint templates in database. And in order to reduce the computing time of verification, the matching scores among the templates are also preserved in the enrollment stage.

#### 2.1.1. Template selection

In enrollment stage, suppose the set of enrolled fingerprints of the same finger is represented as

$$E = \{F^i \mid i = 1, 2, 3, \dots, m\} \quad (1)$$

where  $m$  is the number of the enrolled fingerprints and  $F^i$  is the  $i$ th fingerprint.  $S(F^i, F^j)$  means similarity score of two enrolled fingerprints  $F^i$  and  $F^j$ . We will choose  $n$  ( $n < m$ ) fingerprints as the templates.

The template selection method is described as follows.

Step 1. For every enrolled fingerprint  $F^a$  from the same finger, we will get all the matching scores  $S(F^i, F^j)$  with other fingerprints  $F^j$  ( $j \neq i$ ). And then the average score will be calculated as

$$AVE_i(F^i) = \frac{1}{m-1} \sum_{j \neq i} S(F^i, F^j) \quad (2)$$

The  $a$ th fingerprint that the  $AVE_a(F^a)$  is the maximum will be chosen as the first template fingerprint.

Step 2. For the second template fingerprint, the fingerprint  $F^b$  that the  $S(F^a, F^b)$  is minimum will be chosen as the second template. In this step, we only calculate these scores between the  $a$ th fingerprint and the others.

Step 3. For the third template fingerprint, the fingerprint  $F^c$  which is farthest to the  $F^a$  and  $F^b$  will be chosen. The farthest is defined that  $\frac{1}{2}(S(F^a, F^c) + S(F^b, F^c))$

is the minimum. These matching scores  $S(F^a, F^c)$  and  $S(F^b, F^c)$  ( $c \neq a$  and  $c \neq b$ ) are accepted, and then we calculate the minimum value

$$\left\{ \frac{1}{2}(S(F^a, F^c) + S(F^b, F^c)) \mid c \neq a \&\&c \neq b \&\&c \in [1, m] \right\} \quad (3)$$

and the  $F^c$  is as the third template.

Step  $n$ . For the  $n$ th template fingerprint, the matching scores between the remaining fingerprints with the former  $n - 1$  template fingerprint are calculated. And then we get the minimum

$$\left\{ \frac{1}{n-1}(S(F^a, F^n) + S(F^b, F^n) + \dots + S(F^{n-1}, F^n)) \mid n \neq a \&\&n \neq b \dots n \neq n-1 \&\&n \in [1, m] \right\} \quad (4)$$

and the  $F^n$  is as the  $n$ th template.

### 2.1.2. Establish polyhedron

As shown in Figure 3, we take three templates as an example. In this case, the three templates selected from FVC2004DB4 are all chosen by using the template selection method.

$T_1, T_2, T_3$  indicate the three templates,  $L_{12}, L_{13}, L_{23}$  indicate the similarity distance among the three templates. Next, process of establish polyhedron is described in detail.

Template set is represented as

$$T = \{F^i \mid i = 1, 2, \dots, n\} \quad (5)$$

where  $n$  is the number of the template fingerprints. The set of similarities within templates is represented as

$$I = \{S(F^i, F^j) \mid F^i, F^j \in T\} \quad (6)$$

Suppose every template  $F^i$  is a point  $r_i$  in an  $n - 1$  dimensional space, it can be represented as

$$r_i(x_{i1}, x_{i2}, \dots, x_{i(n-1)}) \quad (7)$$

Considering the distance between  $F^i$  and  $F^j$  in an  $n - 1$ -dimensional space,

$$l_{ij} = \| r_i r_j \| = \sqrt{\sum_{k=1}^{n-1} (x_{ik} - x_{jk})^2} \quad (8)$$

Because the centroid of regular polyhedron is its geometrical center [17], the centroid of  $T$  in an  $n - 1$ -dimensional space is

$$r_c \left( \frac{1}{n} \sum_{i=1}^n x_{i1}, \frac{1}{n} \sum_{i=1}^n x_{i2}, \dots, \frac{1}{n} \sum_{i=1}^n x_{i(n-1)} \right) \quad (9)$$

### 2.2. Verification stage

In this section, a distance calculated from query to centroid and a fusion rule will be introduced in detail.

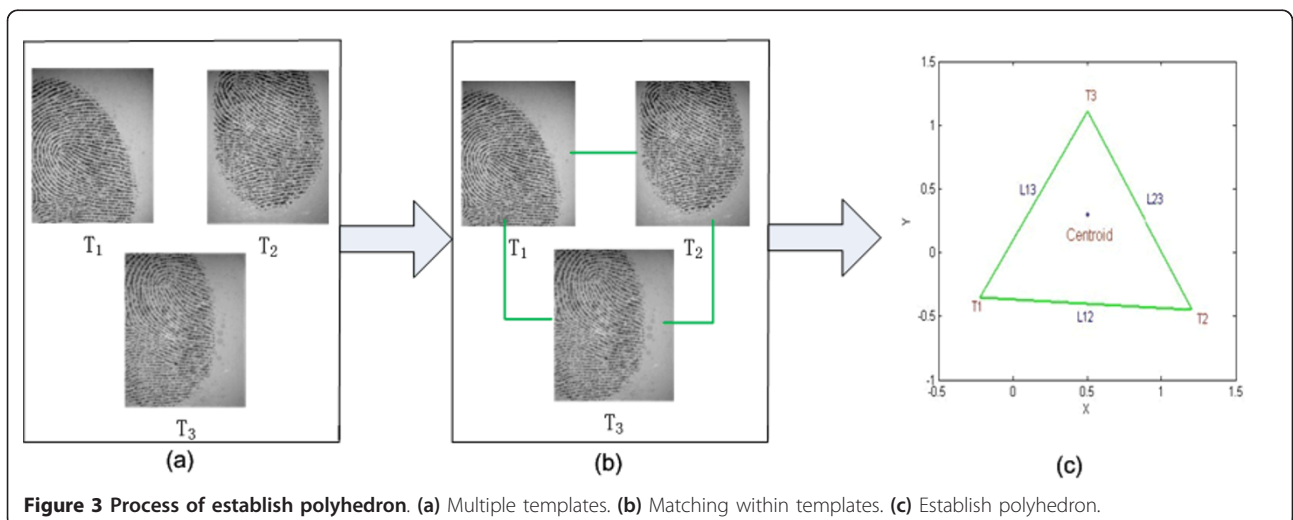
When a query image is presented, the matching proceeds as follows:

- The query image and each template of the same finger stored in database are matched to generate matching scores, and these scores are translated to distance using a proper distance expression;
- Computing the distance from query image to the centroid, and output the distance.
- Choosing a perfect distance and translating it to score using the inverse distance expression as the final score.

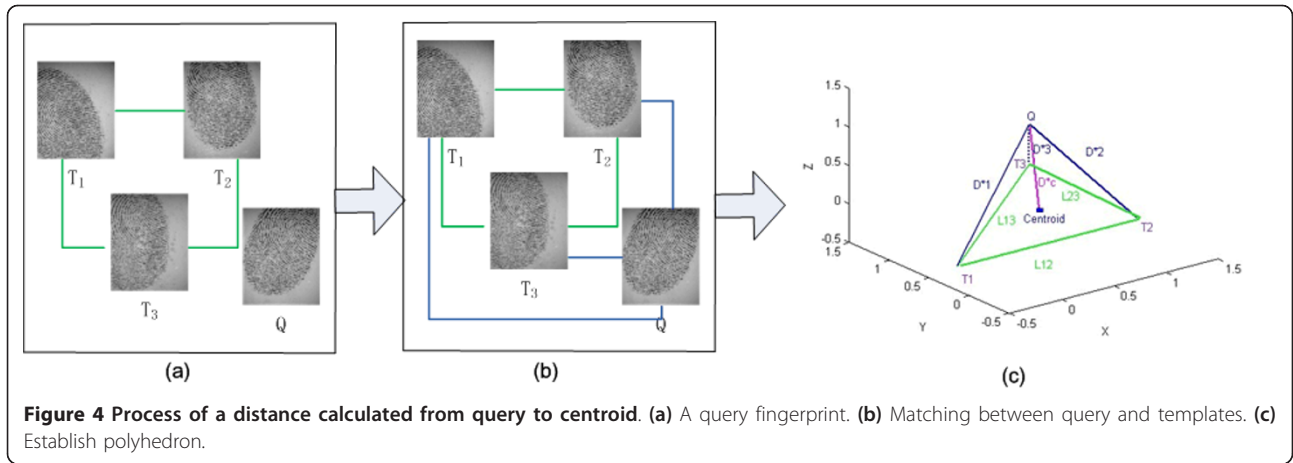
#### 2.2.1. A distance calculated from query to centroid

When a query fingerprint is coming, the process of a distance calculated from query to centroid is shown in Figure 4.

$Q$  indicates the query fingerprint,  $D_{*1}, D_{*2}, D_{*3}$  indicate the similarity distance between the query and templates,



**Figure 3** Process of establish polyhedron. (a) Multiple templates. (b) Matching within templates. (c) Establish polyhedron.



and  $D_{*c}$  indicates the distance between the query and the centroid.

In verification stage, a set of matching scores can be calculated between query image  $Q$  and every template fingerprint. The set is represented as

$$V = \{S(F^i, Q) \mid F^i \in T\} \quad (10)$$

Because the query becomes a member of the polyhedron, the dimension of the polyhedron should be plus one dimension. And the point  $r_i$  of template  $F^i$  be represented as

$$r_i(x_{i1}, x_{i2}, \dots, x_{i(n-1)}, 0) \quad (11)$$

The centroid of  $T$  in an  $n$ -dimensional space is

$$r_c \left( \frac{1}{n} \sum_{i=1}^n x_{i1}, \frac{1}{n} \sum_{i=1}^n x_{i2}, \dots, \frac{1}{n} \sum_{i=1}^n x_{i(n-1)}, 0 \right) \quad (12)$$

So, the query image  $F^*$  is  $r^*$  in an  $n$ -dimensional space, it can be represented as

$$r_*(x_{*1}, x_{*2}, \dots, x_{*(n-1)}, x_{*n}) \quad (13)$$

The distance from  $F^i$  to  $F^*$  will be

$$d_{*i} = \| r_* r_i \| = \sqrt{\sum_{k=1}^n (x_{*k} - x_{ik})^2} \quad (14)$$

The distance from  $F^*$  to the centroid will be

$$d_{*c} = \| r_* r_c \| = \sqrt{\sum_{k=1}^n \left( x_{*k} - \frac{1}{n} \sum_{i=1}^n x_{ik} \right)^2} \quad (15)$$

and,

$$\begin{aligned} d_{*c}^2 &= \sum_{k=1}^n \left( \frac{1}{n} (n x_{*k} - \sum_{i=1}^n x_{ik}) \right)^2 \\ &= \frac{1}{n^2} \sum_{k=1}^n ((x_{*k} - x_{1k}) + (x_{*k} - x_{2k}) + \dots + (x_{*k} - x_{nk}))^2 \\ &= \frac{1}{n^2} \sum_{k=1}^n ((x_{*k} - x_{1k})^2 + (x_{*k} - x_{2k})^2 + \dots + (x_{*k} - x_{nk})^2 + 2 \sum_{i=j+1}^n \sum_{j=1}^{n-1} (x_{*k} - x_{ik})(x_{*k} - x_{jk})) \\ &= \frac{1}{n^2} \left( \sum_{k=1}^n d_{*k}^2 + \sum_{k=1}^n \sum_{i=j+1}^n \sum_{j=1}^{n-1} ((x_{*k} - x_{ik})^2 + (x_{*k} - x_{jk})^2 - (x_{ik} - x_{jk})^2) \right) \\ &= \frac{1}{n^2} \left( \sum_{i=1}^n d_{*i}^2 + \sum_{i=j+1}^n \sum_{j=1}^{n-1} (d_{*i}^2 + d_{*j}^2 - l_{ij}^2) \right) \\ &= \frac{1}{n^2} \left( n \sum_{i=1}^n d_{*i}^2 - \sum_{i=j+1}^n \sum_{j=1}^{n-1} l_{ij}^2 \right) \end{aligned}$$

So,

$$d_{*c} = \frac{1}{n} \sqrt{n \sum_{i=1}^n d_{*i}^2 - \sum_{i=j+1}^n \sum_{j=1}^{n-1} l_{ij}^2} \quad (16)$$

Because  $n$  is const in an instance, the final result  $\propto$

$$n \sum_{i=1}^n d_{*i}^2 - \sum_{i=j+1}^n \sum_{j=1}^{n-1} l_{ij}^2 \quad (17)$$

The final matching result will be given if we decide the distance expression. For example, the inverse of similarity  $S(F^i, F^j)$  is a naïve choice of distance expression. In this article, we use the distance expression  $\left( \frac{1-s}{1+s} \right)$  to compute the final matching result. In [16], we have confirmed that the distance expression  $\left( \frac{1-s}{1+s} \right)$  is good.



### 2.2.2. Fusion rule

Now, we have got all the distance. For the geometric architecture which was built by the distance among the templates, there are two different statuses showing in Figure 5.

In Figure 4, the  $Q$ ,  $T_1$ ,  $T_2$ ,  $T_3$ ,  $C$  means query image, template 1, template 2, template 3, centroid, respectively. These red lines mean the distance from query image to the template fingerprints. The green line means the distance from query image to the centroid of this geometric architecture. In Figure 5a, the length of red line is similar, so the green line is shortest. But in Figure 5b, the query image is more similar with template 2, and the black line is shortest. We all know that the more short of the length, the more similar. So, in this stage, we will use the Min rule to get the minimum distance from all the distance. And in the geometric architecture, we will get the shortest line.

Sometimes, we could get the distance between the query and templates, however, the geometric architecture could not be built because the distance cannot meet the rule of polyhedron. So, the distance between the query and the centroid cannot exist. In this case, we get the minimum distance between the query fingerprint and the templates as the final result.

## 3. Relative fingerprint matching algorithm

In this section, two base matchers that include minutiae-based algorithm [18] and ridge-based algorithm [19] will be introduced briefly. And in the experiment, the results are given based on the two base matchers.

### 3.1. Minutiae-based fingerprint matching algorithm

We choose a typical minutiae-based matching algorithm, which matches the fingerprint images using both the

local and global structures of minutiae [18]. The process of the minutiae-based matching algorithm is shown in Figure 6. The local structure of a minutia is rotation and translation invariant because it consists of the direction and location relative to some other minutiae. It is used to find the correspondence of two minutiae sets and to increase the reliability of the global matching. Moreover, the local structure can tolerate some deformation because it is formed from only a small area of the fingerprint. So, the local structures can be directly used for matching and the best matched local structures will provide the correspondences for aligning the global structure of the minutiae. The global structure of minutiae reliably determines the uniqueness of fingerprint. The aligned global structure together with the result of the local structure matching finally determines whether the two fingerprints are acquired from the same finger. Therefore, the local and global structures of minutiae together provide a solid basis for reliable and robust minutiae matching.

### 3.2. Ridge-based fingerprint matching algorithm

The ridge-based algorithm [19] chosen in this article consists of three stages: preprocessing, alignment, and matching, whose process is shown in Figure 7. In the preprocessing stage, ridges are extracted by sampling equidistantly from the thinned image. The relations between ridges and minutiae are established. In the alignment stage, a set of  $N$  initial substructure pairs is found using a novel approach. In the matching stage, for each of the  $N$  initial substructure pairs, ridge matching is performed to produce a matching score. Finally, the maximum of the  $N$  scores is used as the final matching score of the two fingerprints. The alignment algorithm focuses on how to choose a reliable

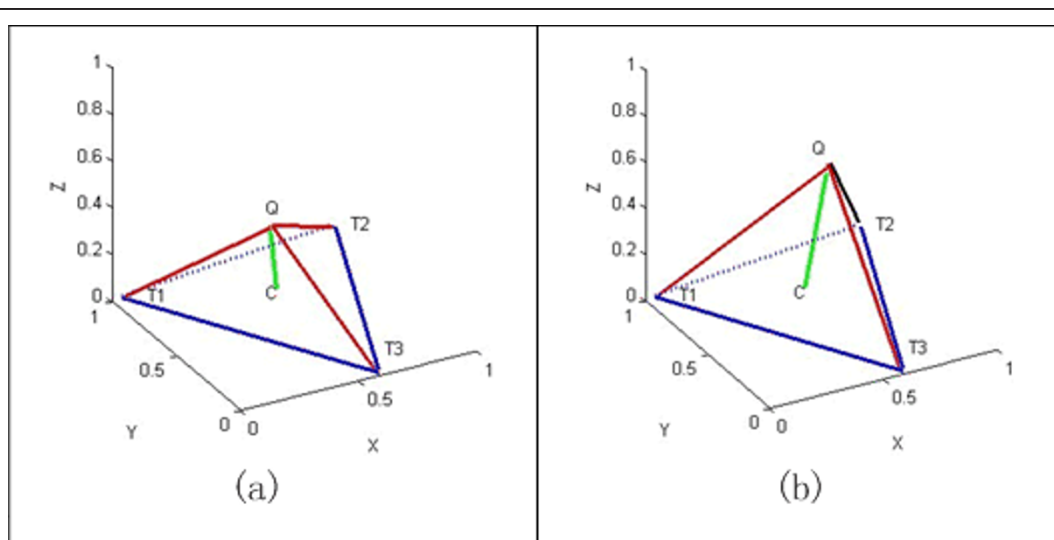
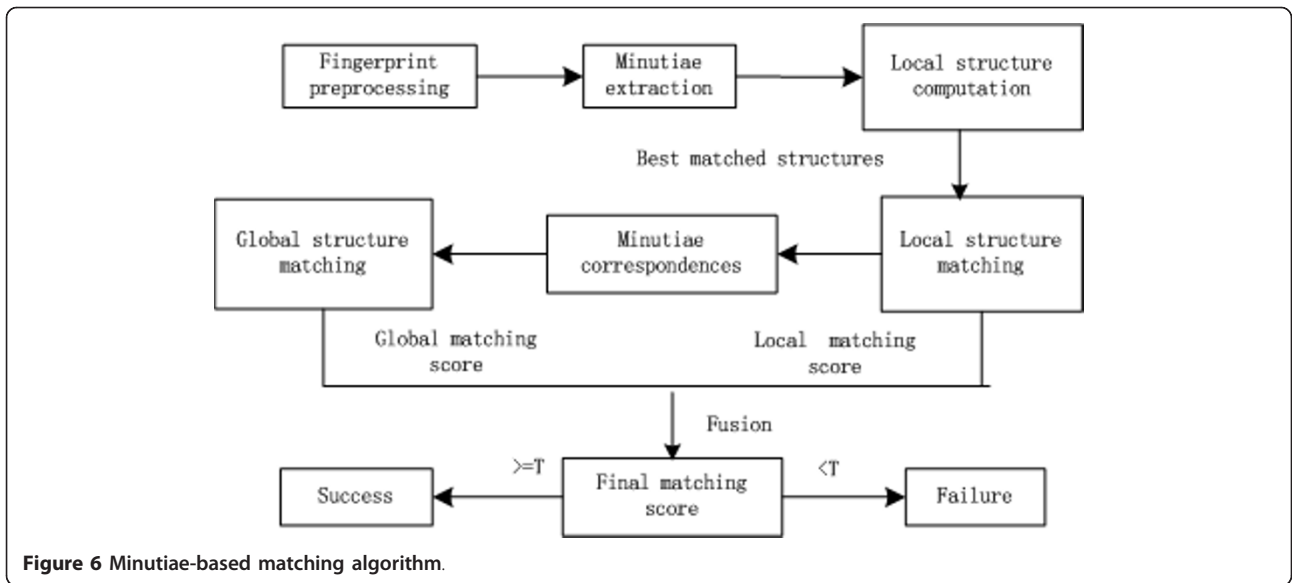
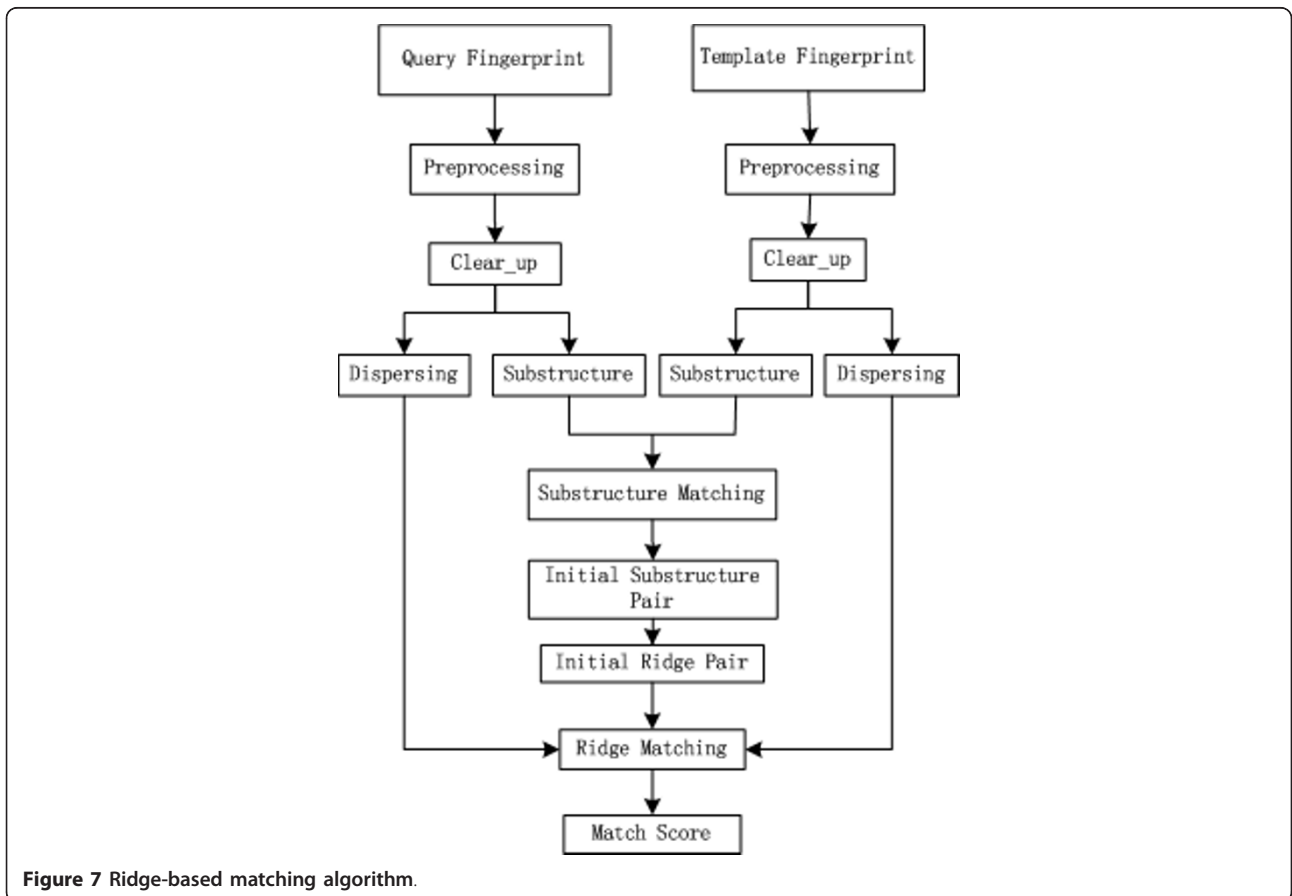


Figure 5 Two different statuses.



local feature pair as the datum mark of matching. This is accomplished first by defining a substructure that contains as much local information (one minutia and several ridges) as possible, and second by finding the

substructure pair which have the most consistent substructure pairs around. In the matching algorithm, during the process of ridge matching, minutiae are also paired, and the matching score is computed



**Table 1 EER of the two base matchers**

Database	DB1	DB2	DB3	DB4	Average
Minutiae-based	16.97	10.05	9.55	6.83	10.85
Ridge-based	18.11	14.99	13.69	11.53	14.58

according to both the matched minutiae and the matched ridges.

#### 4. Experimental results

In this section, we present results on fingerprint database FVC2004 database. This database has four sub-databases: DB1, DB2, DB3, and DB4. Each sub-database consists of fingerprint impressions obtained from 100 non-habituated, cooperative subjects, and every subject was asked to provide eight impressions of the same finger.

The performance of a biometric system is often measured in terms of False Acceptance Rate (FAR) and False Rejection Rate (FRR). FAR and FRR are defined as

$$FAR = p(D_1 | \omega_2) \quad (18)$$

and

$$FRR = p(D_2 | \omega_1) \quad (19)$$

where  $\omega_1$  and  $\omega_2$  represent the classes of true genuine matches and impostor matches, respectively,  $D_1$  and  $D_2$  denote the decisions of genuine matches and impostor matches, respectively. The EER is computed as the point where  $FAR(t) = FRR(t)$ , usually we use EER to evaluate the biometric system [20]. And the performance of the biometric system can also be shown as a receiver operating characteristic (ROC) curve that plots the FRR against the FAR at different thresholds on the matching score. In the experimental results, we will show out the performance of a fingerprint verification system by using the EER and ROC, respectively.

In these experiments, a minutiae- and a ridge-based matching method are used as the base matchers of the fingerprint verification system, and Table 1 lists EERs of the two base matchers.

#### 4.1. Template selection results

In this section, the proposed template selection is compared to MDIST [10] template selection. Uludag et al. [10] proposed two methods for template selection: DENT and MDIST, but MDIST method gets a better performance than DEND in their study. Lumini and Nanni [11] presented a novel clustering method for template selection, and this method is better than MDIST in their study. While this clustering method is depicted simply, we cannot reappear, so we select the MDIST as comparison.

When we carry out our experiments, each sub-database is divided into two subsets called template and query databases. Images selected by using template selection methods constitute the template database and the remaining images of the finger constitute the query database. A maximum matching score is chosen from all scores between a query fingerprint and templates as final score. We perform a comparison among the following methods for the same template selection:

Double-templates (DT): two images are selected as the templates, six images as the query images. And there will be 200 images in the template database, 600 images in the query database.

Three-templates (TT): three images are selected as the templates, five images as the query images. And there will be 300 images in the template database, 500 images in the query database.

Four-templates (FT): four images are selected as the templates, four images as the query images. And there will be 400 images in the template database, 400 images in the query database.

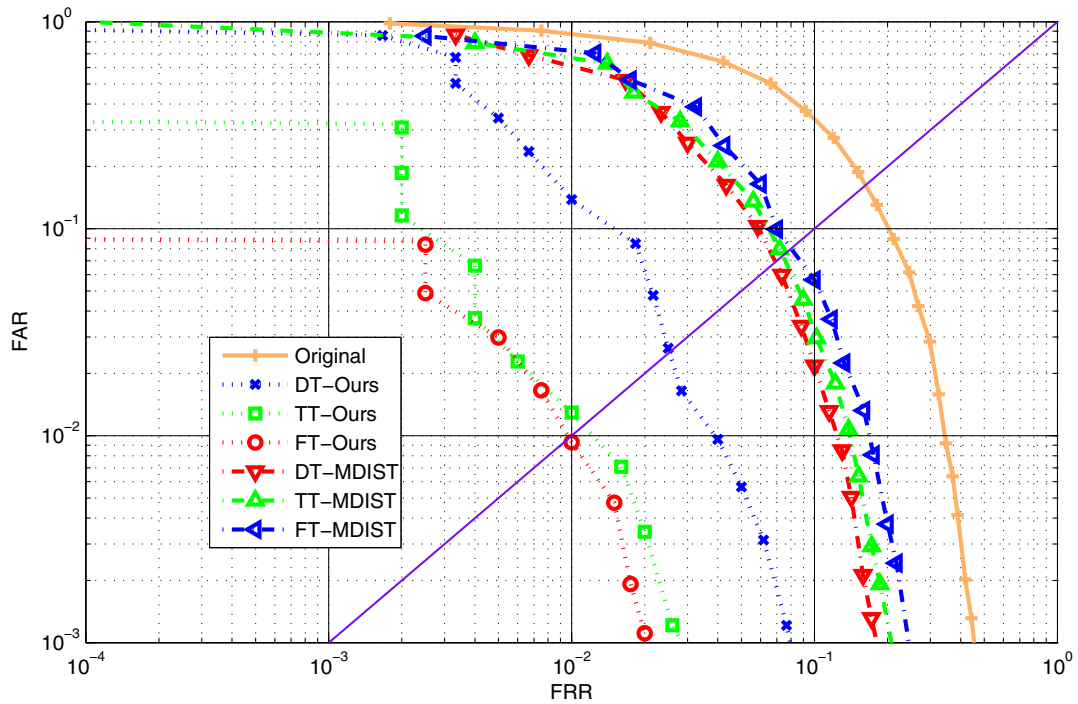
**Table 2 The EERs of template selection using minutiae-based method**

Database	DB1		DB2		DB3		DB4		Average	
	MDIST	Ours	MDIST	Ours	MDIST	Ours	MDIST	Ours	MDIST	Ours
DT	6.66	2.57	5.16	2.75	4.84	2.92	2.16	0.74	4.71	2.25
TT	7.59	1.15	6.13	2.33	5.36	1.40	2.42	0.68	5.38	1.39
FT	8.47	0.96	6.81	2.10	8.87	0.96	2.58	0.46	6.68	1.12

**Table 3 The EERs of template selection using ridge-based method**

Database	DB1		DB2		DB3		DB4		Average	
	MDIST	Ours	MDIST	Ours	MDIST	Ours	MDIST	Ours	MDIST	Ours
DT	10.77	6.17	6.36	3.74	7.82	5.48	4.30	3.51	7.31	4.73
TT	11.79	3.32	7.17	2.56	8.94	4.13	4.73	2.64	8.16	3.16
FT	12.89	1.98	7.18	2.70	10.67	3.15	5.28	2.21	9.01	2.51





**Figure 8** ROC curve of our proposed TS on FVC2004DB1.

In Table 2, we use minutiae-based method, for DT, TT, FT, our proposed template selection method outperforms the MDIST, and the average EERs is lower than original matcher. In our proposed template method, with the number increasing of templates, the EERs are more lower, but, in MDIST, the EERs are more higher. In Table 3, for ridge-based method, it shows the same characteristic. ROC curve on FVC2004DB1 is given in Figure 8, the base matcher is minutiae-based method and shows such an improvement in matching accuracy results by using our proposed template selection method.

Our proposed template selection method is better than MDIST. The reasons may be that the templates selected by our proposed method have a perfect complementary, while the templates selected by MDIST may be a high similarity and their differences are small.

#### 4.2. Verification results

In this section, we will show the EERs of our proposed framework using minutiae- and ridge-based methods.

DT-framework, TT-framework, FT-framework mean our proposed framework using double templates, three templates, and four templates, respectively.

In Table 4, we use minutiae-based method as the base matcher, compared to Table 2, for the same templates,

our proposed framework has a more performance than only using template selection method. And for ridge-based method, it shows the same characteristic in Table 5. From the ROC curve on FVC2004DB1 in Figure 9, we will show out the performance improving clearly.

Finally, although, with the number increasing of templates, the EERs are lower, but the resource of computing and storing is increasing. So, this is a trade-off between performance and resource. For guaranteeing

**Table 4** The EERs of our proposed framework using minutiae-based method

Database	DB1	DB2	DB3	DB4	Average
DT-framework	2.67	2.68	1.67	0.81	1.96
TT-framework	1.05	2.01	1.23	0.48	1.19
FT-framework	0.43	1.96	0.74	0.39	0.88

**Table 5** The EERs of our proposed framework using ridge-based method

Database	DB1	DB2	DB3	DB4	Average
DT-framework	5.85	3.54	4.19	3.10	4.17
TT-framework	2.36	2.47	2.97	2.45	2.56
FT-framework	1.71	2.58	2.49	1.75	2.13

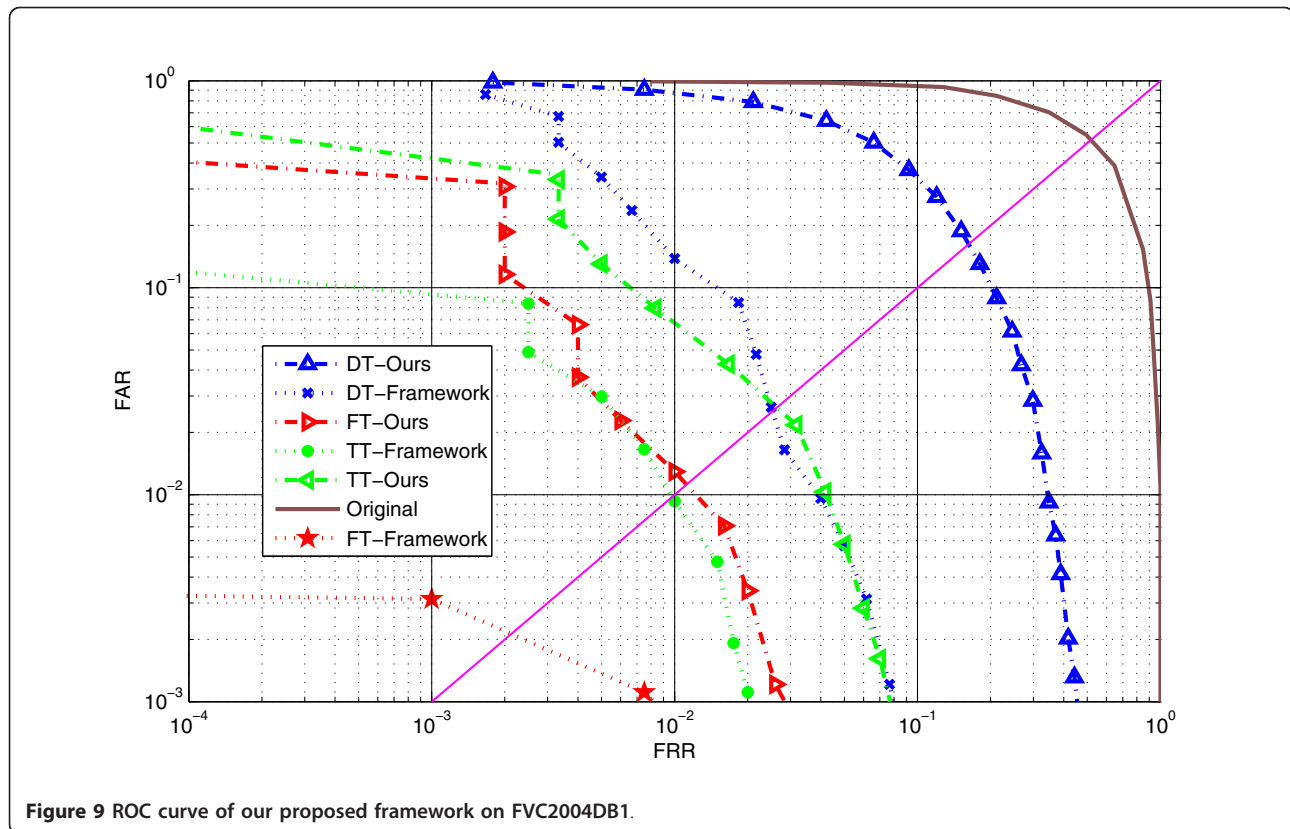


Figure 9 ROC curve of our proposed framework on FVC2004DB1.

the verification accuracy and resource saving, we recommend to use three templates.

## 5. Conclusions

The main contributions of this article to the fingerprint verification are (1) a template selection method is proposed, and this method is more robust and effective than the MDIST. (2) A polyhedron is established by using matching scores among templates, and gets a virtual centroid of the polyhedron. When a query image is inputted, a distance between the query image and the centroid is calculated, and then a distance is chosen from all the distances as the final score. (3) A complete framework for the fingerprint verification system is built based on these two steps.

The experiment of this framework is carried out on the FVC2004DB4 database. Due to the number of fingerprint images in the database the experiment results only represent the functionality of the framework. Future study is to be done on more samples to further verify the performance of both the new template selection method and the proposed framework.

## Acknowledgements

This study was partly supported by the National Natural Science Foundation of China under Grant No. 61070097, 61173069 and the Research Fund for the Doctoral Program of Higher Education under Grant No. 20100131110021.

## Competing interests

The authors declare that they have no competing interests.

Received: 5 July 2011 Accepted: 19 January 2012

Published: 19 January 2012

## References

1. TS Ong, TBJ Andrew, NCL David, YW Sek, Fingerprint images segmentation using two stages coarse to fine discrimination technique. *Ai* 2003. *Adv Artif Intell.* **2903**, 624–632 (2003)
2. L Hong, YF Wan, A Jain, Fingerprint image enhancement: algorithm and performance evaluation. *IEEE Trans Pattern Anal Mach Intell.* **20**(8), 777–789 (1998). doi:10.1109/34.709565
3. A Farina, ZM Kovacs-Vajna, A Leone, Fingerprint minutiae extraction from skeletonized binary images. *Pattern Recognit.* **32**(5), 877–889 (1999). doi:10.1016/S0031-3203(98)00107-1
4. M Tico, P Kuosmanen, Fingerprint matching using an orientation-based minutia descriptor. *IEEE Transact Pattern Anal Mach Intell.* **25**(8), 1009–1014 (2003). doi:10.1109/TPAMI.2003.1217604
5. AK Jain, A Ross, S Prabhakar, A hybrid fingerprint matching using minutiae and texture features, in *Proc Int Conf Image*, Thessaloniki, Greece, pp. 282–285 (2001)
6. A Ross, A Jain, J Reisman, A hybrid fingerprint matcher. *Pattern Recognit.* **36**(7), 1661–1673 (2003). doi:10.1016/S0031-3203(02)00349-7
7. AK Jain, S Prabhakar, A Ross, Fingerprint matching: data acquisition and performance evaluation. *MSU Technical Report TR99-14* (1999)
8. CY Yang, J Zhou, A comparative study of combining multiple enrolled samples for fingerprint verification. *Pattern Recognit.* **39**(11), 2115–2130 (2006). doi:10.1016/j.patcog.2006.05.008
9. D Maltoni, D Maio, AK Jain, S Prabhakar, *Handbook of Fingerprint Recognition*, (Springer, New York, 2003)
10. U Uludag, A Ross, A Jain, It biometric template selection: a case study in fingerprint. *Pattern Recognit.* **37**(7), 1533–1542 (2004). doi:10.1016/j.patcog.2003.11.012

11. A Lumini, L Nanni, A clustering method for automatic biometric template selection. *Pattern Recognit.* **39**(3), 495–497 (2006). doi:10.1016/j.patcog.2005.11.004
12. A Jain, A Ross, Fingerprint mosaicking, in *2002 IEEE Int. Conf. Acoustics, Speech, Signal Process.*, Orlando, FL, USA, pp. 4064–4067 (2002)
13. K Choi, HS Choi, S Lee, J Kim, Fingerprint image mosaicking by recursive ridge mapping. *IEEE Trans Syst Man Cyber Part B Cybern.* **37**(5), 1191–1203 (2007)
14. AK Jain, S Prabhakar, SY Chen, Combining multiple matchers for a high security fingerprint verification system. *Pattern Recognit Lett.* **20**(11-13), 1371–1379 (1999). doi:10.1016/S0167-8655(99)00108-7
15. H Fronthaler, K Kollreider, J Bigun, Fingerprint image-quality estimation and its application to multialgorithm verification. *IEEE Trans Inf Forensics Secur.* **3**(2), 331–338 (2008)
16. CX Ren, YL Yin, J Ma, GP Yang, A novel method of score level fusion using multiple impressions for fingerprint verification, in *2009 IEEE Conf Syst Man (Smc 2009)*, vol. 1-9. San Antonio, TX, pp. 5051–5056 (2009)
17. B Grünbaum, G Shephard, Rotation and winding numbers for planar polygons and curves. *Trans Am Math Soc.* **322**, 169–188 (1990). doi:10.2307/2001527
18. XD Jiang, WY Yau, Fingerprint minutiae matching based on the local and global structures, in *15th Conf Pattern Recognit*, vol. 2. Barcelona, Spain, pp. 1038–1041 (2000)
19. JJ Feng, Z Ouyang, AN Cai, Fingerprint matching using ridges. *Pattern Recognit.* **39**(11), 2131–2140 (2006). doi:10.1016/j.patcog.2006.05.001
20. JL Wayman, AK Jain, D Maltoni, D Maio, *Biometric Systems-Technology, Design and Performance Evaluation*, (Springer, New York, 2005)

doi:10.1186/1687-6180-2012-14

**Cite this article as:** Yin et al.: A framework of multitemplate ensemble for fingerprint verification. *EURASIP Journal on Advances in Signal Processing* 2012 **2012**:14.

**Submit your manuscript to a SpringerOpen® journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](http://springeropen.com)

---