

An efficient algorithm for fingerprint matching based on convex hulls

Chengming Wen

School of Mathematical Sciences
Graduate University of Chinese Academy of Sciences
Beijing, China
wenchengming@163.com

Tiande Guo

School of Mathematical Sciences
Graduate University of Chinese Academy of Sciences
Beijing, China
tdguo@gucas.ac.cn

Abstract—The purpose of fingerprint matching is to compare two fingerprint images and return a similarity score that represents to the probability of match between the two fingerprints. The performance of an automatic fingerprint identification system is greatly determined by its fingerprint matching algorithm. In this paper, a full fingerprint matching algorithm is presented. The main contribution of this paper is that a novel algorithm based on convex hulls for eliminating spurious matching in fingerprint matching is proposed. It is very important for improving the performance of fingerprint matching algorithm to reducing spurious matching. The algorithm is tested on a public database FVC2002 DB1_A. Experimental results show that the proposed algorithm is performed well compared with other ones.

Keywords—fingerprint; recognition; minutiae; convex hull; matching

I. INTRODUCTION

Fingerprint recognition, as a popular application in pattern recognition, is very important in identity authentication due to the uniqueness and invariability of fingerprint feature. As a key issue for automatic fingerprint identification systems (AFIS), fingerprint matching highly influences the performances of AFIS. Usually, fingerprint matching can be classified into three categories: correlation-based matching, minutiae-based matching and ridge feature-based matching [1]. Among them, minutiae-based matching is the most widely used method in AFIS.

The common minutiae models adopt two kinds of minutiae feature in the fingerprint image: ridge ending and ridge bifurcation. The minutiae matching problem is generally addressed as a point matching problem in pattern recognition. Each minutia may be described by a number of attributes such as its position (x,y) , its orientation θ . Therefore the extracted fingerprint features can be seen as vectors. And fingerprint matching problem can be seen as the problem of matching two different vector point data sets. It means that given two point sets $A=\{a_1, a_2, \dots, a_m\}$ and $B=\{b_1, b_2, \dots, b_n\}$ where a_i and b_j are vectors (x, y, θ) , how to find a suitable way to match these points in the two sets.

For minutia-based algorithms, fingerprint matching is to establish the number of corresponding minutia pairs to compute the final matching score. A successful method is proposed in [2], which is based on graph matching principles. It defines a representation called K-plet which is also used here to encode the local neighborhood of each minutia. A minutia-based partial fingerprint recognition system is designed in [3]. A graph model of minimum cost

flow is used to build one-to-one match. We also draw on the idea of dealing with point matching problem using the graph theory model. But the graph model used here is different from network flow model. A fingerprint matching method which clusters locally matched minutiae and warps the fingerprint surface using minutiae clusters is presented in [4]. The above-mentioned methods all carry out local invariant structure matching firstly. But local structure matching may produce spurious matching because it is processed locally without reference to global information. A novel and efficient algorithm for eliminating spurious matches which are produced by local structure matching is proposed here.

In this paper, a new fingerprint matching algorithm based on convex hull matching principles is presented. Section 2 discusses the matching algorithm in detail. First of all, the query fingerprint is matched to the template fingerprint using the local structures. And matching minutiae pairs are achieved using global matching strategy with maximum weight bipartite matching model. Then eliminate spurious matching by checking convex-hull similarity. Finally, for the score computation, an overlapped area between two convex hulls of template and query fingerprint minutiae is considered in addition to a summation of similarities from matched minutiae pairs. Experimental results are shown in section III and conclusions are presented in section IV.

II. THE MATCHING ALGORITHM

A. Local Structure Matching

The local feature of a minutia describes the characteristics of the minutia in its neighborhood. Each local neighborhood is associated with structural properties that are invariant under translation and rotation. Therefore, local matching algorithms are more robust to non-linear distortion and partial overlaps. K-plet in [2] is used to describe the local structure.

The procedure of constructing the local structure for each minutia as follows: (1) Adjust the coordinates, making the minutia as the origin and the minutia orientation as the X-axis direction. (2) K neighboring minutiae are gained such that K/4 nearest neighbor is chosen in each of the four quadrants sequentially. For the K-plet, the invariant angles and distance from the reference minutia and the neighboring minutiae are defined as θ_i , Φ_i and Dis_i respectively [4]. TMF is defined as the two minutiae feature vector $(\theta_i, \Phi_i, Dis_i)$. It is clear that TMF can be gained for any two minutiae in one fingerprint. Fig. 1 (b) shows the parameters of local feature vector. We empirically set $k=8$ for K-plet in the experiment. A 8-plet

example is displayed in Fig. 1 (a), where square (blue), round (red) and cross (green) headed line represent a reference minutia, two minutiae selected for each quadrant and the remaining neighboring minutiae which have not been selected.

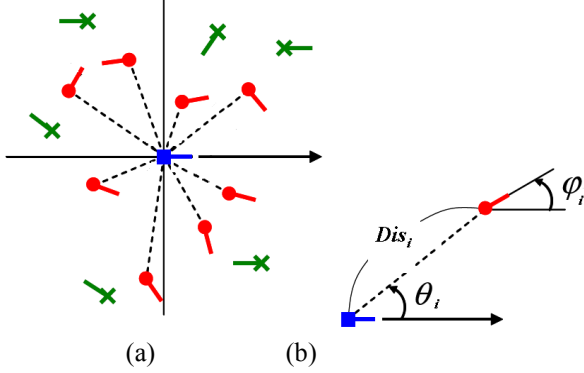


Figure 1. (a) An example of 8-plet; (b) Parameters of feature vector between two minutiae

W_{ij} represents the similarity score of the two K-plets between the minutia i from query image and the minutia j from template image. Set W_{ij} ($i = 1..m, j = 1..n$) as zero originally. If the local structure matching score is greater than some threshold, W_{ij} is set as the local structure matching score. The dynamic programming method which was presented in [2] is used to obtain W_{ij} in local structure matching. A local pattern is shown in Fig. 2. From that, it is easily observed that the method of local structure construction here can maintain high connectivity between different parts of the fingerprint.

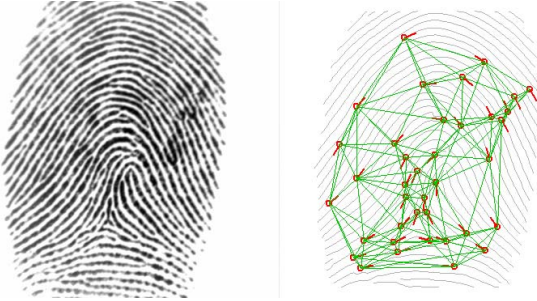


Figure 2. Local structure construction of a fingerprint

B. Maximum Weight Bipartite Matching

Local structure matching produce the initial score which describes the matching probability of any minutiae pair by examining the similarity of the local structures of minutiae. The purpose of global matching is to guarantee one minutia from input image can match to at most one minutia from template image. However, it is hard to find the optimal matching pairing of the feature points. Here, we build a maximum weight bipartite matching model illustrated in Fig. 3. Ambiguities in minutiae pairings are solved by employing maximum weight bipartite matching. The best pairing is the configuration that can maximize the final total score of all matched minutiae pairs. Our method balances the tradeoffs between maximizing the number of matches and minimizing total feature distance between query and template fingerprints.

A bipartite weight graph $G = (L, R)$ to match the minutiae pairs, where L denotes the set of n minutiae in the query image and R denote the set of m minutiae in template image. If m is lower than n , add virtual node in R to make sure $m \geq n$. For minutiae i in L and minutiae j in R , an edge is built between them and the edge weight value equals the local matching score W_{ij} . The weights of the edges relating virtual nodes are all set as 0. The model is illustrated in Fig.3 where LMS means local matching score. The traditional K.M. algorithm [8] for maximum weight bipartite matching is adopted for getting the scheme with maximum total score. After that the edge weight of each matching is checked. If the edge weight is greater than threshold value, the matching minutiae pair is recorded as candidate matching.

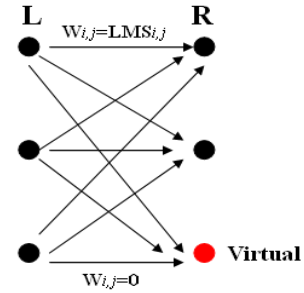


Figure 3. Maximum weight bipartite matching model

C. Check Convex Hull Structure for Eliminating Spurious Matching

Local structure matching does not sufficiently capture the global structural relationships making false matches very common. After that a convex hull matching method is used to reduce false matching result globally. The convex hull of a set Q of points is the smallest convex polygon P for which each point in Q is either on the boundary of P or in its interior. Intuitively, we can think of each point in Q as being a nail sticking out from a board. The convex hull is then the shape formed by a tight rubber band that surrounds all the nails [9].

A convex hull C is firstly built in the minutiae set which contains all the matching feature points in query fingerprint. If the points on the convex hull correspond to the right matching points in template fingerprint, the corresponding matching points in template fingerprint also construct a convex hull S which is similar with the convex hull in query fingerprint. Two similar successful matching convex hulls are illustrated in Fig.4 (a). Red points are the minutiae in query fingerprint while blue points are the minutiae in template fingerprint. The blue geometry shape is the similar with the red convex hull basically. If spurious matching pairs exist, the corresponding matching points in template fingerprint may not form a similar convex-hull as C , and even not form a convex-hull. An example is given in Fig.4 (b). The blue geometry shape is different from the red convex hull completely. So spurious matching can be eliminated by checking convex-hulls' similarity.

Two kinds of features can be used to evaluate the similarity of the convex hulls between query fingerprint and template fingerprint. One is the similarity of geometry structure of two convex hulls. For example, the length of edges and the angles which are constructed by the

adjoining edges in the convex hull can be used to evaluate the similarity of geometry structure. The other is the two minutiae feature vector. One edge of the convex hull corresponds to two minutiae. So TMF can be extracted from one edge. If the edges of the convex hulls are similar between query fingerprint and template fingerprint, the TMFs should be matched. In conclusion, the geometry structure information and the minutiae feature vector can both be used to make sure the similarity of the two convex hulls. By checking the similarity of convex hulls, the right matching pairs can be reserved and the spurious matching pairs can be found and eliminated. Once make sure the matching pairs reliability, the final matching score can be calculated exactly.

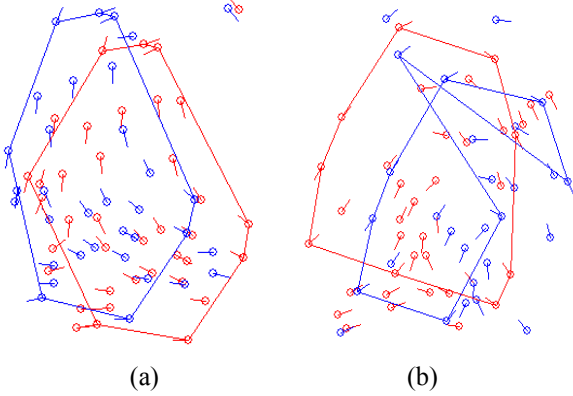


Figure 4. Examples of convex hulls of point sets

The algorithm for eliminating spurious matching is described as follows:

Step 1: The matching points in query fingerprint are recorded as set A. Sort the points in A by local matching score.

Step 2: Divide set A into small parts. The first M points with maximum local score are grouped to the first part G₁. The second M points are as the second group G₂. The rest may be deduced by analogy.

Step 3: For each group G_i, construct a convex hull C using Graham algorithm [9]. The geometry shape S is constructed by the corresponding matching points of C in template fingerprint. If all matching pairs in C are right, S should be similar with C.

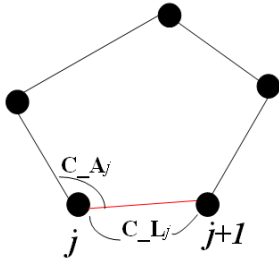


Figure 5. Illustration of the angle and edge length of an convex-hull

Step 4: Check the similarity between C and S.

Geometry similarity: For each vertex j in C, compare the angle C_Aj, edge lengths C_Lj with the matching point in S. Fig 5 illustrates C_Aj and C_Lj.

Feature vector matching: For each edge in C or S, the two vertexes are two minutiae. So a feature vector TMF can be calculated for each edge. For every edge in C and

corresponding edge in S, check the similarity between the two TMFs.

Step 5: If find spurious matching pair, eliminate it from matching results. For checked points C, separate them from G_i. Go back to Step 3 until G_i is empty.

Step 6: Go back to Step 3 to check the next Group G_{i+1} until G_{i+1} does not exist.

The flow chart of the algorithm for eliminating spurious matching pairs is shown in Fig.6. This algorithm is explicit and easy to be executed. There are many spurious matching pairs in the result of local matching. Spurious matching can be found by checking the convex hull similarity. As showed in Fig. 4(b), spurious matching pairs can be easily found. If the matching pairs are correct, the shape constructed by the corresponding matching points in template fingerprint should be similar with the convex hull in query fingerprint. The idea of checking convex hull similarity is feasible and effective for ensuring the correctness of matching result.

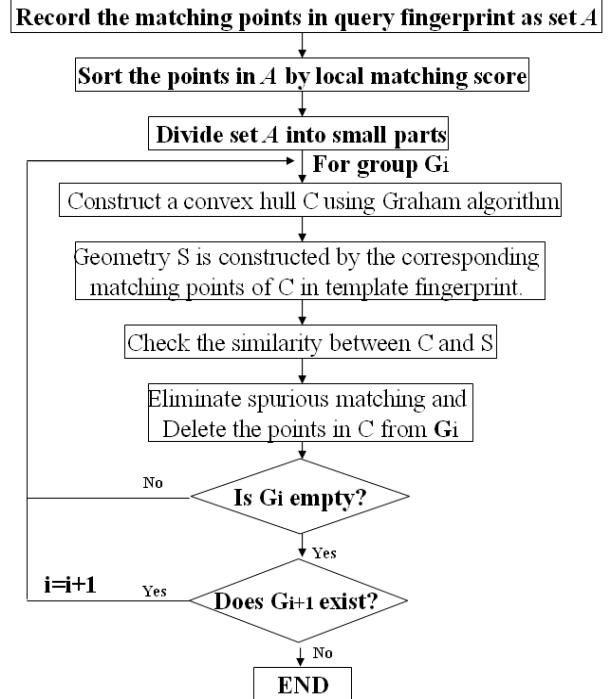


Figure 6. Flow chart of eliminating spurious matching

D. Scoring

The calculation of final matching score between query fingerprint and template fingerprint is crucial for fingerprint matching. To make the matching score function more discriminatory, a new score computation method is proposed which considers the number of successful matched TMF and the overlapped area of template and query fingerprints. The following equation is used here for score computation:

$$s(Q, T) = \frac{M}{C_v + (O_Q + Q_T - N)} \quad (1)$$

where

$$M = \sum_{i=1}^N \sum_{j=i+1}^N sim(FQ_{ij}, FT_{ij}), \quad (2)$$

$$C_v = \frac{N \times (N-1)}{2} - V_r. \quad (3)$$

N is the number of matched minutiae pairs. For any two matched minutiae pairs i and j , FQ_{ij} is the TMF of the two minutiae of query fingerprint while FT_{ij} is the TMF of the two minutiae of template fingerprint in matched pairs i and j . The token sim represents the similarity score between them. If pair i and pair j are both true matching, $\text{sim}(FQ_{ij}, FT_{ij})$ should correspond to a high score. V_r means the number of the matched TMF. V_r is set as zero originally. If $\text{sim}(FQ_{ij}, FT_{ij})$ is greater than some threshold, increase V_r by 1. OQ and OT represent the numbers of minutiae in the estimated overlapped region in query fingerprint and in template fingerprint respectively. The method for estimating OQ and OT is described in [3].

III. EXPERIMENTS

FVC2002 [7] DB1_A database is used to evaluate the performance of the proposed matching algorithm. The database consists of 800 fingerprint images with 100 distinct fingers and 8 impressions per finger. A total of 2,800 and 4,950 comparisons for genuine and imposter matching respectively are tested. In order to demonstrate the efficiency of eliminating spurious matching algorithm, the matching algorithm which removes the eliminating spurious matching phase from the proposed algorithm is also tested. The comparative results are presented in Table 1. ROC curve (Receiver Operation Characteristic) of tested algorithms is given in Figure 7. From the ROC performance result, it is clear that the blue algorithm (the full matching algorithm in this paper) shows a great advantage while maintaining the same FMR with lower FNMR. The red algorithm is the algorithm without executing the algorithm for eliminating spurious matching by checking convex hull similarity. From Table 1 and Fig. 7, it is safely to conclude that the proposed method for eliminating spurious matching improves the performance of AFIS efficiently. The EER of the proposed algorithm is 0.536%. Comparing with the results of the algorithms [2]-[6] which are tested FVC2002_DB1_A, it is demonstrated that the performance of the propose algorithm is very well.

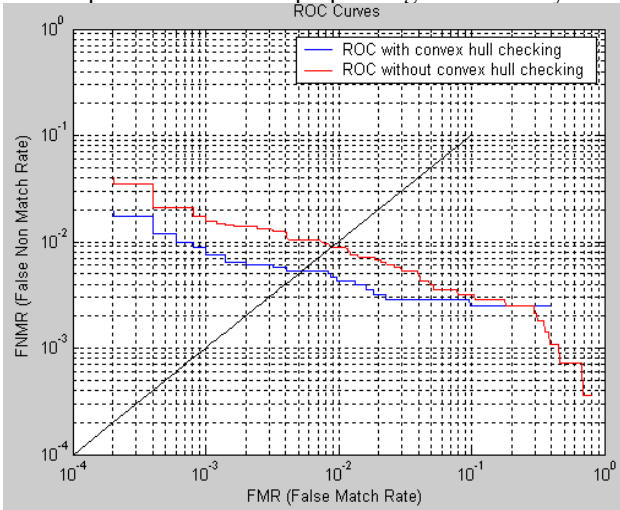


Figure 7. A comparison of ROC curves for FVC2002 DB1_A

TABLE I. A SUMMARY OF THE COMPARATIVE RESULTS ON FVC2002_DB1_A

Algorithm	EER	FMR100	FMR1000
without convex hull checking	0.893%	0.964%	1.786%
The proposed algorithm	0.536%	0.464%	0.929%

IV. CONCLUSIONS

An efficient and robust matching algorithm is crucial for determining the performance of AFIS. Local invariant structure matching is applied in many minutiae-based fingerprint matching algorithms. But a lot of spurious matching pairs are produced after local structure matching. It is a valuable problem about how to eliminating spurious matching completely and accurately for enhancing the accuracy of fingerprint recognition. This paper presents a novel fingerprint matching algorithm based on convex hull matching principles which is an improvement of traditional minutiae-based fingerprint matching method. The method of checking convex hull similarity is introduced into the whole matching process for reducing spurious matching. Experiment results on FVC database show that the method for eliminating spurious matching by checking convex hull similarity makes the EER reduced from 0.893% to 0.536%. Comparing with the results of other papers on the same database, the proposed full matching algorithm has a better performance than previous common minutiae point matching methods.

ACKNOWLEDGMENT

This work is supported by National Scientific Support Program of China, under Grants GA 2007A002 and GA 2007B002, also supported by CAS Innovation Program, under Grants kjcx-yw-s7.

REFERENCES

- [1] D. Maltoni, D. Maio, A. K. Jain, S. Prabhakar, "Handbook of Fingerprint Recognition", Springer-Verlag, June 2003.
- [2] Sharat Chikkerur, Alexander N. Cartwright, and Venu Govindaraju, "K-plet and Coupled BFS: A Graph Based Fingerprint Representation and Matching Algorithm", ICB 2006, LNCS 3832, pp. 309-315, 2005.
- [3] T.Y. Jea and V. Govindaraju, "Aminutia-based partial fingerprint recognition system", Pattern Recognition 38 (2005) 1672 - 1684.
- [4] D. Kwon, I. D. Yun, D. H. Kim and S. U. Lee "Fingerprint Matching Method Using Minutiae Clustering and Warping", the 18th International Conference on Pattern Recognition (ICPR'06).
- [5] W. Sheng, G. Howells, M. Fairhurst, and F. Deravi, "A Memetic Fingerprint Matching Algorithm", IEEE Trans. on information forensics and security, Vol. 2, No. 3, Sep. 2007.
- [6] Li Tian, Liang Chen, and Sei-ichiro Kamata, "Fingerprint Matching Using Dual Hilbert Scans", Third International IEEE Conference on Signal-Image Technologies and Internet-Based System.
- [7] <http://bias.csr.unibo.it/fvc2002/>
- [8] Douglas B. West, "Introduction to Graph Theory", second edition, ISBN 0-13-014400-2, 2001.
- [9] T.H. Cormen, C.E. Leiserson, R.L. Rivest, "Introduction to Algorithms", McGraw-Hill, New York, 1998.