

A Multiple Substructure Matching Algorithm for Fingerprint Verification

Mabel Iglesias Ham, Yilian Bazán Pereira, and Edel B. García Reyes

Adv. Tech.Application Center 7a No. 21812, Siboney, Miramar, 12200, Habana, Cuba
{miglesias,ybazan,egarcia}@cenatav.co.cu

Abstract. In this paper, we represent a fingerprint image with a Delaunay graph formed by minutiae as nodes. The graph has attributes which contributes to the final similarity measure and are invariant under rotation and translation. We design an algorithm for the comparison of these graphs based on the similarities of multiple common substructures. We use different heuristics to tackle the problems of noise, deformation and partial matching found in fingerprint recognition. We match star structures and extend it by edges maintaining the local structural compatibility. Finally, we consolidate the global similarity taking into account the size of the common substructures and the accumulated similarity of all stars involved. We use a simple greedy algorithm obtaining a very efficient performance. We use our proposed method in some experiments with fingerprint images in databases from FVC2002. It shows better results compared to other known algorithms as K-plet, and several others recently published.

Keywords: graph matching, fingerprint recognition, Delaunay triangulation.

1 Introduction

Fingerprints have been widely used in personal identification, access control, financial security, and verification of driver license applicants [1]. However, fingerprint matching is a difficult problem that has not been solved yet for a generic application. The problem of comparing fingerprint impressions appears in two different forms, verification and identification. The verification is applied in systems of access control, where it is needed to determine if a pair of impressions belongs to the same finger or not. Identification, used in forensic applications, refers to the process of establishing the identity of a person from its impression, which entails the comparison of this impression with the one stored in a database returning a list of candidates where it must appear the corresponding one in positive case. Finally the resulting list is examined by the specialist who is the one in charge of giving a final decision.

Minutiae pattern is the most widely used method for representing a fingerprint. Each minutia has some characteristics as coordinates, type and direction. Extraction of minutiae features generally needs a series of processes, including orientation computation, image segmentation, image enhancement, ridge extraction and thinning,

minutiae extraction and filtering. This paper only discusses the process of fingerprint matching.

Fingerprint matching based on minutia features is a well researched problem. Yet, several problems remain as challenges nowadays as tough situations of non-linear distortions, low-quality images and partial prints. A minutia based fingerprint matching system usually returns the number of matched minutiae from query and reference fingerprint and uses it to generate a similarity. Generally, more matched minutiae yield higher similarity scores. We can confidently distinguish a genuine or an imposter fingerprint using the number of matched minutiae when the number of minutiae on both fingerprints is large. However, it is not reasonable to use an absolute number of matched minutiae alone in case of partial fingerprint. A traditional way to calculate the similarity score for a minutiae-based system is $n^2 / (n_1 * n_2)$. Where n_1 and n_2 represent the number of minutiae on query and reference fingerprint, and n is the number of matched minutiae on both prints. This method is unreliable, especially when we are matching fingerprints of different sizes. For this reason we change the similarity score measurement. Kovacs-Vajna [2] has shown that the geometric deformations on local areas can be more easily controlled than global deformation. That is why we pose the problem as a search of multiple common substructures simultaneously on both graph representations of fingerprints and use an accumulative process of evidence about the similarity.

Usually, the point pattern matching methods first align two sets of minutiae and then identify the corresponding ones using an adaptive bounding box to handle the spatial distortion. For that reason we propose a scheme based on graphs, to construct a translation and rotation invariant feature avoiding the alignment step for the local matching. To overcome the effect of missing or fake minutiae in a local area we adopted a Delaunay triangulation graph representing the proximity relations among minutiae.

It is widely believed that if two fingerprints are from the same finger they would share the same local spatial distribution of minutiae. Human experts, in practice, pay special attention to the relative positions among some candidate minutiae pairs. Some approaches in automatic fingerprint matching imitate this observation looking for a global compatibility of the local structures using a graph that represents these local relations [3, 4, 5]. Our idea is similar to the algorithm proposed by Sharat et al., at the ICB2006 [6]. We also, starting in a minutia pair, expand the substructure by the neighbourhood maintaining the local structural compatibility. Nevertheless, we use for the expansion a modification of DFS (depth-first search) strategy instead of BFS (breadth-first search) as Sharat et al. does. The space complexity of DFS is much lower than BFS (breadth-first search). It also lends itself much better to heuristic methods of choosing a likely-looking branch as in our case of study. They used K-plet to represent the topological relationships of the feature set while we use a Delaunay triangulation with the same purpose. In the local matching they employ a dynamic programming approach based on string alignment algorithm and converting the unordered neighbours of each K-plet into an ordered sequence by arranging them in the increasing order of the radial distance. This order is affected because of the fact that distances vary with the translation of minutiae produced by ridge deformations in the process of capturing the image even by different skin conditions or pressure. Meanwhile we use a greedy algorithm for the local matching which is a simpler and

faster heuristic. We obtain a more elaborated consolidation taking into account the similarities of several common substructures representing different areas with low levels of noise. Meanwhile, K-plet uses only the maximum substructure found to measure the global similarity. The K-plet graph does not ensure a connected representation restricting the expansion capacity of the method, not in the case of the Delaunay graph of our proposal.

In section 2, we discuss some issues about the representation of the topological relationships of minutiae. Our heuristic for multiple substructures matching using attributed Delaunay minutiae graphs is given in section 3. The results of the experiments are presented in section 4 using some standard indexes of FVC competition protocol. Finally in section 5, conclusions and future work are presented.

2 Representation

Graphs provide a powerful representation technique in many areas of computer vision including Biometrics. In particular, the structural relationships among minutiae have a good discrimination power. Here the basic idea is to recognise fingerprints by comparing the graph-structure. To the best of our knowledge, four different kinds of representations have been used to capture the local structural information in fingerprints based on minutiae: Minutiae Adjacency Graph (MAG) [7], K-nearest neighbour (K-NN)[4], K-plet[6] and Delaunay triangulation[3, 8, 9].

MAG, K-NN and K-plet do not ensure a connection between all parts of the fingerprint. In MAG two nodes of a graph are considered neighbours if the Euclidean distance between them is less than or equal to a threshold. In K-NN representation, the Euclidean distances between a minutia and its neighbours are calculated and arranged in an increasing order selecting the k-nearest neighbours to form a star. In K-plet to maintain high connectivity at least a neighbouring minutia is chosen in each of the four quadrants sequentially leading to directed edges. The Delaunay triangulation connects two nodes if they have adjacent Voronoi neighbourhoods. Voronoi neighbourhood of a point is the set of points that are nearer to the reference point than to any other point [10, 11]. So in that way points connected by Delaunay triangulation are intuitively neighbours. The Voronoi diagrams and Delaunay triangulation approaches were originally introduced to study proximity relationship among objects in space. Delaunay triangulation ensures connectivity using approximately 2.6 edges by node, which is a very compact and effective representation to propagate matches globally, whilst K-plet needs at least 4 edges by node without warranty of a connected graph.

In this paper, we represent a fingerprint feature set as a Delaunay triangulation. The Delaunay triangulation of a non-degenerate set of points is unique. One problem is that it is sensitive to noise and distortions (e.g, introduced by missing or spurious minutiae points). However, the insertion of a new point in a Delaunay triangulation affects only the nearby triangles. As a result, noise affects the Delaunay triangulation only locally. This means that correct identification will be possible if some regions of the fingerprint have not been seriously affected. In [12] the robustness of various closest point graphs to corruption was investigated. The conclusion reached was that

the Delaunay graph is the most resistant to disruption, a result of considerable significance to the matching of noised fingerprints.

We store translation and rotation invariant attributes as the minutiae type in the nodes; in the edges, the distance between the connected minutiae (D) and the angles between the minutiae directions and the edge in the clock direction (α_1, α_2), as shown in the Figure1.

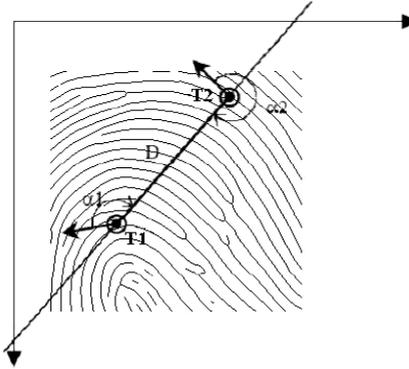


Fig. 1. Minutiae and edge attributes in the graph. T1 and T2 are the types of minutiae; D is the edge length or distance between minutiae and α_1, α_2 the angles between minutiae directions and the edge in the clock direction.

3 Multiple Substructures Matching: MSM

The heuristic comes from the fact that two fingerprints represented as Delaunay minutiae graphs share common substructures belonging to a set of nodes that remains in both prints with low deformation errors. An example of a common substructure in two fingerprints from the same finger, where the local neighbourhood of two matching nodes looks similar, is shown in the Figure 2.

The method compares each node in the reference graph with each node in the model graph finding a common substructure starting from them simultaneously and propagating the match to other nodes in the neighbourhood eventually.

3.1 Local Substructure Matching (Star Matching)

There are two main processes for the comparison of the local substructures. The first is the measurement of the similarity between the adjacent edges to the central vertex (S_a), and the second is the measurement of the similarity between the adjacent vertexes to the central vertex (S_v). If both of the previous similarities are greater than zero then they are combined to emit a final similarity score between both stars, giving a predominant weight to the similarity of the adjacent edges, as shown in the following equation (1).

$$S_t = 0.35 * S_v + 0.65 * S_a . \quad (1)$$

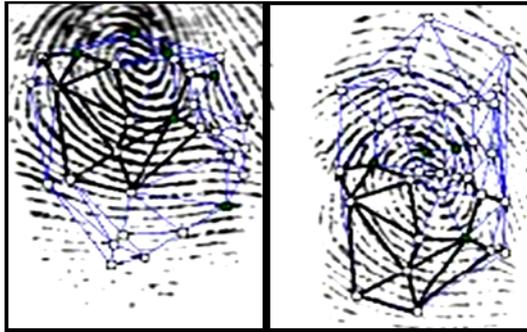


Fig. 2. Example of a common substructure in Delaunay minutiae graphs in two fingerprint images of DB1A FVC2002

3.1.1 Measurement of the Similarity Between the Adjacent Edges to the Central Vertex

A greedy algorithm is used to associate the edges of the first star to the edges of the second local structure. The previously associated edges are marked so they are not associated again in further steps. The similarity of the associated edges are stored and counted. The inferiority in the number of vertexes in the first star is guaranteed by the method. The similarity between two edges is measured as follows.

Two edges ($a1$ and $a2$) are considered similar if the angles $\alpha1$ and $\alpha2$ are similar by the formula (2) using the difference between angles ($dang1, dang2$).

$$\begin{aligned}
 dang1 &= (|a1.\alpha1 - a2.\alpha1| > 180) ? (360 - |a1.\alpha1 - a2.\alpha1|) : |a1.\alpha1 - a2.\alpha1| . & (2) \\
 dang2 &= (|a1.\alpha2 - a2.\alpha2| > 180) ? (360 - |a1.\alpha2 - a2.\alpha2|) : |a1.\alpha2 - a2.\alpha2| . \\
 & dang1 < AngleThreshold . \\
 & dang2 < AngleThreshold . ,
 \end{aligned}$$

and the distances (lengths of the edges) are similar by the formula (3) using the ratio of length (Rd).

$$\begin{aligned}
 Rd &= |a1.D - a2.D| / \text{Min}(a1.D, a2.D) . & (3) \\
 Rd &< DistanceThreshold .
 \end{aligned}$$

Finally if the constraints of distance and angles are satisfied the similarity (s) of a pair of edges is evaluated as formula 4 shows.

$$s = 1 - Rd . \tag{4}$$

In other cases the similarity is evaluated in zero and it is considered that the edges do not match. Once we have an association of adjacent edges by means of the greedy strategy, the final similarity (Sa) is evaluated as appears in the equation bellow (5).

$$Sa = \text{SumEdgesPairSim} / \text{NumEdgesPairs} . \tag{5}$$

Where $NumEdgesPairs$ is the number of associated adjacent edges and $SumEdgesPairSim$ is the sum of the respective similarities.

Our proposal makes a final correction of the similarity score taking into account the type of minutiae in the center of the stars. If there is a coincidence in the minutia

type the final score is increased in a 30% otherwise is decreased in the same percentage.

3.1.2 Measurement of the Similarity Between the Adjacent Vertexes to the Central Vertex

In the star, similarly to the measurement of the matching scores in the adjacent edges, a greedy algorithm is used to associate the vertexes of the first star to the vertexes of the second local structure. The previously associated vertexes are marked so they are not associated again in further steps. The similarity of the associated vertexes are stored and counted. The inferiority in the number of vertexes in the first star is guaranteed by the method. The similarity between two adjacent vertexes is measured as the similarity of its respective adjacent edges using the same procedure defined in section 3.1.1. The final similarity between the adjacent vertexes (S_v) is defined as in the equation stated below (6).

$$S_v = \text{SumVertexPairSim} / \text{MinNumVertexes} . \quad (6)$$

Where *MinNumVertexes* is the number of adjacent vertexes of the first vertex (degree) and *SumVertexPairSim* is the sum of the similarities of the associated vertexes.

3.2 Local Substructure Expansion

Given a vertex pair v_1 and v_2 not previously marked as belonging to the common substructure, the local star similarity is measured. If they are considered sufficiently similar based on the local similarity, the vertexes are accepted in the common substructure and marked. The expansion is attempted from the first pair of edges (selecting a reference edge adjacent to v_1 and comparing to the set of edges adjacent to v_2) with similarity greater than zero with the respective adjacent vertexes unmarked. The mentioned adjacent vertexes are then considered as the centers of a new pair of stars for performing a new local matching. If the expansion fails because the new vertexes are not similar enough, then the algorithm will attempt to associate the reference edge in v_1 to other consecutive edge in v_2 for a possible expansion. The similarities of the pairs of vertexes accepted in the common substructure measured as the star similarities are stored (*sumStarSim*) and counted (*subsSize*). The strategy mentioned is similar to the DFS with the restriction that the traversal is made simultaneously in both graphs and the vertexes are visited only if they have a similar star structure.

The final similarity measure between a common substructure (*Sim*) is evaluated by the expression 10, taking into account a weight in the size of the substructure (*sizeW*) regarding the number of vertexes in the shorter graph (*MinNumVertexes*) as shows the formula 7, and the average of the similarities of its pairs of associated vertexes (A_v) (8,9).

$$\text{sizeW} = \text{subsSize} / \text{MinNumVertexes} . \quad (7)$$

If the size of the common substructure is greater than 6 or greater than the 70% of the total of vertexes of the shorter graph then the A_v is evaluated as follows (8). In other cases formula 9 is used.

$$Av = \text{sumStarSim} / \text{subsSize} . \quad (8)$$

$$Av = \text{sumStarSim} / \text{MinNumVertexes} . \quad (9)$$

$$\text{Sim} = \text{sizeW} * Av . \quad (10)$$

3.3 Global Similarity Consolidation

The human expert performs a first visual comparison of all the minutiae in the reference fingerprint against all minutiae in the query fingerprint taking into account the nearby context. Starting by a pair of minutiae, both neighbourhoods are analyzed looking for matching and expanding the similar area as long as it is possible.

Our proposal reproduces this behavior. The first step is to obtain the graph with the lower number of vertexes (g1) to assume it belongs to the query fingerprint. The bigger graph (g2) represents the reference fingerprint. This is related to the search of a fragment of impression realized by an expert in a stored fingerprint.

The next step is to find the correspondences of vertexes of graph g1 in the set of vertexes of graph g2. Each vertex of g1 is associated with the vertex of graph g2 from which it is obtained the best similarity using a greedy strategy. The previously associated vertexes are marked so they are not used again in the subsequent associations. It is necessary a similarity of the common substructure starting by a pair of vertexes greater than zero to be considered in the association process. The search of the common substructure is made first looking for a local minutia matching (star matching) and finally expanding the common local structure by a minutia in a sequence of local matchings.

It is possible that starting from a pair of common minutiae there is not a connected substructure that contains all the coincidences of both prints even when they belong to the same finger. This is due to the graph deformations produced by missing minutiae or false minutiae occurrence in the feature extraction process, apart from the nonlinear geometric deformations that produce a bound in the possible expansion of the common substructure. This is the reason why, starting by different corresponding minutiae pairs it is possible to find different common substructures. Each of these common substructures contains relevant information for the global similarity of a pair of fingerprints.

There are three cases in the consolidation of the global similarity from the similarities of the multiple common substructures found by each vertex of the reference graph:

1. If the number of vertexes of the reference graph that found a similar substructure is greater than 5 then it is considered that there is a relevant number of representatives and the final similarity is the average of the similarities of the best substructures found by each vertex. This procedure is similar to the expert criterion of the necessity of at least 6 minutiae to emit a relevant similarity score.
2. If the mentioned number is lower than 6 then the ratio regarding the total of vertexes of g1 (the shorter graph) is evaluated. If this ratio is superior to a threshold then the final similarity is the average of the similarities of the best substructures found by each vertex. It is the case when the reference fingerprint is a segment of

impression with a low number of minutiae of which a considerable number reach a local correspondence at least.

3. If the number of associated vertexes is not representative regarding the total of vertexes of the shortest graph, the final global similarity is evaluated in zero.

4 Experimental Results

A series of experiments have been conducted on the test data set of FVC. We use standard indexes of comparison that appear published in the official site for the competition of verification FVC [13] as the EER and FMR. Each FVC databases contains 100 different fingers and 8 impressions for each one yielding a total of 800 fingerprints. For FMR (False Matching Rate) the total number of impostor test is $(100*99)/2 = 4950$. For FNMR (False Non-matching Rate) the total number of genuine test is $(8*7)/2*100 = 2800$.

We think it is important to compare our approach with previous methods using local structural similarity. First experiment compares our approach with the algorithm based on graph matching principles presented by Sharat *et al.* at ICB2006 as shown in Table 1.

Table 1. A summary of the comparative results w.r.t. Sharat *et al.* algorithm

Database	Sharat <i>et al.</i> Algorithm K-plet		Proposed Approach, MSM	
	EER	FMR100	EER	FMR100
FVC2002 DB1	1.5%	1.65%	0.56%	0.5%

We also compare our algorithm with a minutiae-based matching scheme proposed by Yansong *et al.* at ICPR2006 [14], which introduced a concept of compatibility to the minutiae triangle structures and adopted a relaxation process to adjust the similarity matrix of the minutiae triangle cells between the query and template images. This experiment was performed on the DB1_A of FVC2004. FVC2004 databases are markedly more difficult than FVC2002 and FVC2000 ones, due to the perturbations and geometrical deformations deliberately introduced. Table 2 describes our results compared to Yansong *et al.* algorithm.

Table 2. A summary of the comparative results w.r.t Yansong *et al.* algorithm

Database	Yansong <i>et al.</i> Algorithm	Proposed Approach, MSM
	EER	EER
FVC2004 DB1	9.4%	8.09%

We compare our approach with a minutiae based fingerprint verification algorithm using Delaunay triangulation presented by Deng and Huo [3], at the AVBPA 2005. This algorithm tries to find the best-matching edge pairs in the local structure instead of finding best-matching minutiae pairs. The global matching score between two fingerprints is calculated by using an aligned-edge-guided triangle procedure. Table 3 describes our results compared to Deng and Huo algorithm.

Table 3. A summary of the comparative results *w.r.t* Deng and Huo algorithm

Database	Deng and Huo AVBPA2005	Proposed Approach, MSM
	EER	EER
FVC2002 DB1	1.82%	0.56%
FVC2002 DB2	1.52%	0.71%

5 Conclusions and Future Work

In this paper, we apply a heuristic method to obtain a similarity measure between two fingerprints represented as attributed Delaunay graphs based on the comparison of multiple common substructures. This algorithm is simple, fast and does not need an explicit alignment between both fingerprints. We obtain a better global similarity measure taking into account the similarities of several common substructures representing correspondences of different areas of the impressions with low distortion errors. Another contribution is that we guarantee a connected graph in order to allow a total global substructure expansion unlike K-plet representation of Sharat *et al.* Also, the space complexity of DFS traversal is much lower than BFS (breadth-first search). It also lends itself much better to heuristic methods of choosing a likely-looking branch as in our case of study.

We compared our method with several algorithms recently published and we obtain better results without many efforts tuning the thresholds. Especially, using FVC2002 DB1 database we reach an EER of 0.56% and FMR100 of 0.5%. We do not tackle the particular problem of correcting the relevant non-linear deformations as included in FVC2004 databases in this contribution. However in a preliminary experiment with FVC2004 DB1A we obtain a better EER index compared with the Yansong *et al* proposal. Nevertheless, we notice that no linear deformation of fingerprint is an important issue to deal with. Also, we want to develop an automatic methodology to learn the thresholds.

References

1. Federal Bureau of Investigation: The Science of Fingerprints: Classification and Uses. Washington, D.C.: U.S. Government Printing Office (1984)
2. Kovacs-Vajna, Z.M.: A fingerprint verification system based on triangular matching and dynamic time warping. *IEEE Trans. PAMI* 22(11), 1266–1276 (2000)
3. Deng, H., Huo, O.: Minutiae Matching Based Fingerprint Verification Using Delaunay Triangulation and Aligned- Edge-Guided Triangle Matching. In: Kanade, T., Jain, A., Ratha, N.K. (eds.) AVBPA 2005. LNCS, vol. 3546, pp. 270–278. Springer, Heidelberg (2005)
4. Jiang, X., Yau, W.Y.: Fingerprint minutiae matching based on the local and global structures. In: *Proc. Internat. Conf. on Pattern Recognition*, pp. 1038–1104 (2000)
5. Isenor, D.K., Zaky, S.G.: Fingerprint Identification Using Graph Matching. *Pattern Recognition* 19(2), 113–122 (1986)
6. Chikkerur, S., Govindaraju, V.: K-plet and CBFS: A graph based fingerprint representation and matching algorithm. In: *International Conference on Biometrics* (2006)

7. Ratha, N., Bolle, R., Pandit, V., Vaish, V.: Robust fingerprint authentication using local structural similarity. In: 15th IEEE Workshop on Applications of Computer Vision, pp. 29–34 (2000)
8. Bebis, G., Deaconu, T., Georgiopoulos, M.: Fingerprint identification using Delaunay triangulation. In: Int. Conf. on Information Intelligence and Systems, pp. 452–459 (1999)
9. Parziale, G., Niel, A.: A Fingerprint Matching Using Minutiae Triangulation. In: Zhang, D., Jain, A.K. (eds.) ICBA 2004. LNCS, vol. 3072, pp. 241–248. Springer, Heidelberg (2004)
10. Guibas, L., Stolfi, J.: Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams. *ACT TOG* 4(2) (1985)
11. Sloan, S.W., Houlsby, G.T.: An Implementation of Watson’s Algorithm for Computing 2-D Delaunay Triangulations. *Advanced Engineering Software* 6(4) (1984)
12. Tuceryan, M., Chorzempa, T.: Relative Sensitivity of a Family of Closest-Point Graphs in Computer Vision Applications. *Pattern Recognition* 24(5), 361–373 (1991)
13. FVC2004. Fingerprint Verification Competition, url: <http://bias.csr.unibo.it/fvc2004/>
14. Feng, Y., Feng, J., Chen, X., Song, Z.: A novel fingerprint matching scheme based on local structure compatibility. In: Int. Conf. on Pattern Recognition (2006)