# Fingerprint Matching Using a Geometric Subgraph Mining Approach

Alfredo Muñoz-Briseño(✉), Andrés Gago-Alonso, and José Hernández-Palancar

Advanced Technologies Application Center (CENATAV), Havana, Cuba
{amunoz,agago,jpalancar}@cenatav.co.cu

**Abstract.** In the present work, a new representation of fingerprints in form of geometric graph, is proposed. This representation is obtained by fusing two previously defined approaches found in the literature and proves to be very tolerant to occlusions and distortions in the minutiae. Also, a novel matching fingerprint algorithm that uses geometric graphs was introduced. The mentioned algorithm applies frequent geometric subgraph mining in order to match fingerprint representations for computing a final similarity score. The introduced proposal reports very promising accuracy values and it applies a new approach allowing many future improvements.

**Keywords:** Fingerprint matching · Graph mining · Delaunay triangulation · Geometric similarity

## 1 Introduction

Biometrics can be seen as the automatic use of physical or behavioural characteristics to identify or verify the identity of a person. One of the most commonly used techniques in biometric systems is the comparison of fingerprints. The ridge patterns found in fingers and other parts of the human body are unique, providing enough information to distinguish a specific person from the rest. Also, these patterns can be extracted in a simple manner, which makes the use of fingerprints a very reliable technique.The goal of fingerprint recognition is to determine if two impressions were generated by the same finger or not, and it is a very treated topic in literature. However, even when there are some very effective solutions, this problem can not be considered entirely solved. In fact, the design of more accurate and efficient algorithms is still a topic of interest.

Most fingerprint recognition algorithms use minutiae in order to represent characteristic information. Minutiae are singularities in the ridge patterns, which are classified as bifurcations and terminations. A bifurcation is a point in which a ridge bifurcates, while a termination represents the termination of a ridge. Another very used feature is the direction of minutiae. This characteristic is defined as the angle formed between the horizontal axis and the tangent of the ridge associated to the minutiae, in counter clock wise. Even when there are some features extraction methods that report good results [1], they can fail

© Springer International Publishing Switzerland 2015
A. Pardo and J. Kittler (Eds.): CIARP 2015, LNCS 9423, pp. 160–167, 2015.
DOI: 10.1007/978-3-319-25751-8_20

under occlusion conditions. In this work, minutiae and their directions are used in order to define a fingerprint representation as a geometric graph. Thus, the fusion of two previously defined representations is used. The introduced matching algorithm uses frequent geometric subgraphs between the representation of two impressions, and it states coherence geometric criterion for calculating a similarity score. This approach represents a new perspective for fingerprint matching presenting very promising results. The rest of this work is organized as follows. The Section 2, introduces some theoretical formulations necessary for the understanding of the work. Section 3 is dedicated to the definition of the fingerprints representation. Section 4 introduces the algorithm for finding matches between the representations of two impressions. Finally, the experimental results that validate our proposal are shown in Section 5 and the final conclusions are given in Section 6.

## 2 Introduction to Geometric Graphs

Undirected geometric graphs are used as base for modeling fingerprints in this work. This kind of graph and their properties are defined as follows:

**Definition 1 (Label domain).** *Let $L_V$ and $L_E$ be label sets, where $L_V$ is a set of vertex labels and $L_E$ represents a sets of edge labels, the label domain of every label is denoted by $L = L_V \cup L_E$.*

**Definition 2 (Geometric graph).** *A geometric graph in $L$ is a 5-tuple, $G = (V, E, I, J, K)$ where $V$ is a set of vertexes, $E \subseteq \{\{u, v\} \mid u, v \in V, u \neq v\}$ is a set of edges (the edge $\{u, v\}$ connects the vertexes $u$ and $v$), $I : V \to L_V$ is a function that assigns labels to vertexes, $J : E \to L_E$ is a function that assigns labels to edges, and finally $K : V \to \mathbb{R}^2$ is a function that assigns coordinates to vertexes, $\mathbb{R}$ represents the set of real numbers, and $K(u) \neq K(v)$ for each $u \neq v$.*

**Definition 3 (Topological isomorphism).** *Let $G_1 = (V_1, E_1, I_1, J_1, K_1)$ and $G_2 = (V_2, E_2, I_2, J_2, K_2)$ be two geometric graphs, $G_1$ is a topological subgraph of $G_2$ if $V_1 \subseteq V_2$, $E_1 \subseteq E_2$, $\forall u \in V_1, I_1(u) = I_2(u)$, and $\forall e \in E_1, J_1(e) = J_2(e)$. ALso, $f$ is a topological isomorphism between $G_1$ and $G_2$ if $f : V_1 \to V_2$ is a bijective function where $\forall u \in V_1, I_1(u) = I_2(f(u))$, and $\forall \{u, v\} \in E_1, \{f(u), f(v)\} \in E_2 \wedge J_1(\{u, v\}) = J_2(\{f(u), f(v)\})$.*

When a topological isomorphism exists between $G_1$ and $G_2$, we can say that $G_1$ and $G_2$ are topologically isomorphic.

In the geometric context, two graphs may be geometrically similar having different vertex coordinates. An example of this can be seen in Figure 1(a). This situation takes place when one of the graphs is rotated, moved or scaled with respect to the other. That is why it is necessary to take in consideration the best geometric transformation for matching the two involved graphs, obtaining a geometric isomorphism. Let $G_1 = (V_1, E_1, I_1, J_1, K_1)$ and $G_2 = (V_2, E_2, I_2, J_2, K_2)$ be two topologically isomorphic geometric graphs. Let $f$ be a topological isomorphism between $G_1$ and $G_2$. A geometric transformation in $\mathbb{R}^2$ is defined as a function $T : \mathbb{R}^2 \to \mathbb{R}^2$ which can be characterized by a scale factor $\lambda$, a rotation angle $\omega$, and a traslation $(t_x,\, t_y)$.

On the other hand, the error of a geometric transformation can be computed using the following expression:

$$\epsilon(T) = \frac{\sum_{v \in V_1} \|K_1(v) - T(K_2(f(v)))\|}{\|V_1\|}. \tag{1}$$

Let $G_1 = (V_1, E_1, I_1, J_1, K_1)$ and $G_2 = (V_2, E_2, I_2,\ J_2, K_2)$ be two topologically isomorphic geometric graphs with $n$ vertexes. Let $f$ be an isomorphism between $G_1$ and $G_2$. The geometric transformation $T$ associated to $G_1$ and $G_2$ is defined as the transformation that minimizes the error $\epsilon(T)$. Using this measure, the concept of geometric isomorphism can be defined.

**Definition 4 (Isomorphism $\tau$-tolerant).** *Let $T$ be the transformation associated to the topologically isomorphic geometric graphs $G_1$ and $G_2$. We can say that $G_1$ and $G_2$ are isomorphic $\tau$-tolerant if $\epsilon(T) < \tau$.*

In this case, the geometric similarity between two graphs $G_1$ and $G_2$ $\tau$-tolerant can be defined with the following expression:

$$\phi(G_1, G_2) = \frac{1}{n\tau} \sum_{v \in V_1} \|K_1(v) - T(K_2(f(v)))\|. \tag{2}$$

In this way, given two geometric graphs $G_1 = (V_1, E_1, I_1, J_1, K_1)$ and $G_2 = (V_2, E_2, I_2,\ J_2, K_2)$, $G_1$ is a geometric subgraph of $G_2$ if some topological subgraph of $G_2$ is isomorphic $\tau$-tolerant with $G_1$.

In Figure 1(b), an example of a geometric transformation $T$ associated to two graphs, can be seen. The distances between the corresponding vertexes are pointed with arrows. If the average value of these distances is higher than the tolerance threshold $\tau$, the graphs are not isomorphic $\tau$-tolerant. In the illustrated case, the first edge of each graph was used to compute $T$.

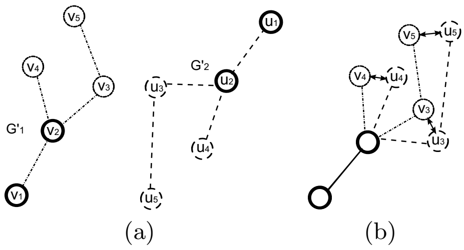



**Fig. 1.** Transformation applied to isomorphic $\tau$-tolerant geometric graphs.

**Definition 5 (Support set).** *Let $D = \{G_1, G_2, \ldots, G_{|D|}\}$ be a collection of geometric graphs and let $\delta$ be a predefined frequency threshold. The support set of a graph $g$ is defined as the set $G_i \in D$ composed by the graphs where $g$ is geometric subgraph. The notations $\Delta(g, D)$ and $\sigma(g, D) = |\Delta(g, D)|$ are used for referring the support set and the frequency of $g$ in $D$, respectively. A graph $g$ is frequent in a collection $D$ if $\sigma(g, D) \geq \delta$.*

**Definition 6 (Geometric occurrence).** *Let $g$ and $G$ be two geometric graphs, such that $g$ is a geometric subgraph of $G$, the topological subgraph $g'$ of $G$, isomorphic $\tau$-tolerant with $g$ is called geometric occurrence of $g$ in $G$. If there is more than one subgraph $g'$ fulfilling this condition, the geometric occurrence will be the one that maximizes the expression $\phi(g, g')$.*

**Definition 7 (Frequency set).** *Let $g$ be a frequent geometric subgraph in a collection $D$, and let $\Delta(g, D) = \{G_1, \ldots, G_N\}$ be its support set. The frequency set of $g$ is defined as $\Gamma(g, D) = \{g'_1, \ldots, g'_N\}$, where $g'_i$ is the geometric occurrence of $g$ in $G_i$.*

There are some algorithms finding every frequent geometric subgraph in a given collection [2,3]. Using these approaches, the construction of the frequency set of each occurrence is a simple process. One of the novelties presented in this work is the use of one of these algorithms in fingerprint matching.

## 3 Fingerprints Representation

In the present work, a fingerprint representation based on the fusion of two previously proposed approaches is introduced. Thus, there are some aspects that need to be mentioned. From a geometric point of view, a triangulation of a set of points $P = \{p_1, \cdots, p_n\}$ in $\mathbb{R}^2$ is a planar subdivision of the plane in triangles $\triangle p_i p_j p_k$. The vertexes of these triangles are made up by points of $P$. A triangulation is considered of Delaunay, and is denoted as $TD(P)$, if the circumcircle of every triangle contains no points of $P$ [4].

This triangulation is unique for a specific set $P$, if there is no circumcircle with more than three points of $P$ at its border. This characteristic makes $TD(P)$ very useful in the field of fingerprints recognition. There are some algorithms that use $TD(P)$ computed from the coordinates of the minutiae, for representing the impressions [5]. These representations have some problems since $TD(P)$ can suffer great structural changes when minutiae are slightly displaced or when some of them are missing. Both situations are common in the feature extraction step, mostly because of the skin elasticity or the occlusions of some fingerprints parts. In order to deal with the minutiae displacement, Muñoz-Briseño et al. [6] proposed a representation based on Delaunay triangulations of order $k$.

**Definition 8 (Delaunay triangulation of order $k$).** *Let $P = \{p_1, \cdots, p_n\}$ be a set of points in $\mathbb{R}^2$, for $p_l, p_m, p_n \in P$, $\triangle p_l p_m p_n$ is a Delaunay triangle of order $k$ is its circumcircle contains at most $k$ points of $P$. Subsequently, a triangulation of $P$ is of Delaunay of order $k$, and is denoted as $TD_k(P)$, if every one of its triangles is of order $k$.*

The representation proposed on the previously mentioned work is made up by the set of triangles:

$$ET_k(P) = TD_0(P) \cup TD_k(P) \tag{3}$$

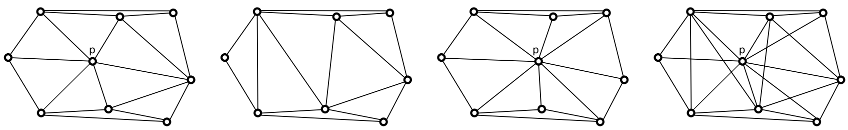
Similarly, Gago-Alonso et al. [7] proposed a representation, denoted as $R$, that avoids the problems found when a minutia is not detected. Using the Delaunay triangulation of order $k$, in this work we propose a generalized variant as follows.

**Definition 9 (Polygonal hull of order k).** *Let $P = \{p_1, \cdots, p_n\}$ be a set of points on $\mathbb{R}^2$, and let $TD_k(P)$ be the Delaunay triangulation of order $k$ of $P$. Let $p_i \in P$, the set $N_i$ denotes the points adjacent to $p_i$ in $TD_k(P)$. The polygonal hull of order $k$ is defined as the Delaunay triangulation of order $k$ of $N_i$, and is denoted as $Hk_i$.*

The final triangle set used in this work is given by the following expression:

$$R_k(P) = ET_k(P) \cup Hk_1(P) \cup \ldots \cup Hk_n(P) \tag{4}$$

Figure 2 shows that a Delaunay triangulation can suffer major structural changes when a point $p$ is not present 2(b) or when its slightly displaced 2(c). However, the proposed triangles set 2(d) preserves many edges in both situations. In this context, this implies that our approach can find correspondences between fingerprints of a same finger, even if some of them have some missing or displaced minutiae. In this way, the set $R_k(P)$ is robust to displacement and absence of minutiae. The defined representation also preserves the linearity of the amount of triangles with respect to the number of minutiae in the impressions. In order to use a geometric subgraph mining algorithm, the associated




(a) Delaunay triangulation (b) Without $p$ (c) With $p$ displaced (d) Proposed representation

**Fig. 2.** Triangles sets.

graph $G_R = (V_R, E_R, I_R, J_R, K_R)$ of $R_k(P)$ is defined, where $V_R$ represents the points of $P$, $E_R$ is composed by every edge contained in the triangles of $R_k(P)$ (every edge is represented only once, even if appears in more than one triangle), $I_R : V \to \{0, 1\}$ is the function that assigns labels to the vertexes (the values of these labels depend on the minutiae type that each vertex represents, bifurcation or termination) and $J_R : E \to \{0, \ldots, 360\}$ is the function that assigns labels to the edges (computed by subtracting the directions of the minutiae that define each edge). With the impressions represented as geometric graphs, the mining method that is the base of the matching algorithm can be applied.

## 4 Fingerprint Matching

Let $G_{R1} = (V_{R1}, E_{R1}, I_{R1}, J_{R1}, K_{R1})$ and $G_{R2} = (V_{R2}, E_{R2}, I_{R2}, J_{R2}, K_{R2})$ be the graphs that represent two fingerprints. The goal of applying a matching algorithm to $G_{R1}$ and $G_{R2}$ is to determine if the impressions belong to the same finger, by computing similarity score. For this, the proposal introduced uses a geometric graph mining algorithm found on the literature [3]. This algorithm will be referred as $FreqGeom(D, \delta)$. The idea is to extract correspondences between $G_{R1}$ and $G_{R2}$ using their common geometric subgraphs. The input of $FreqGeom(D, \delta)$ is set by a collection $D = \{G_{R1}, G_{R2}\}$ in which the mining is performed, and a support. In our case $\delta = 2$, since the matching operation involves only two graphs. The output of $FreqGeom(D, \delta)$ is a set of frequent geometric subgraphs $F = \{g_1, \ldots, g_n\}$, and their respective frequency sets $\{\Gamma_1, \ldots, \Gamma_n\}$. Using $F$, correspondences between the representations are found.

Let $g_i = (V_i, E_i, I_i, J_i, K_i)$ be a geometric subgraph such that $g_i \in F$. The frequency set of $g_i$ is conformed in this context by only two graphs $\Gamma_i(g_i, D) = \{g'_{i1}, g'_{i2}\}$, since $|D| = 2$. Using this information, a similarity tuple can be obtained for each frequent geometric subgraphs found, with the form $s_i = \langle sim_i, T_i \rangle$ where $T_i$ is the geometric transformation associated to $g'_{i1}$ and $g'_{i2}$, and $sim_i$ is given by the following expression:

$$sim_i = (\mid V_i \mid + \mid E_i \mid) \times \phi(g'_{i1}, g'_{i2}) \tag{5}$$

The number of edges and vertexes have a great influence in the similarity value previously defined, since bigger correspondences should have more weight. Let $S = \{s_1, \ldots, s_n\}$ be the set of similarity tuples obtained from the frequent geometric subgraphs $F = \{g_1, \ldots, g_n\}$ in $D$; with the goal of computing the final similarity between $G_{R1}$ and $G_{R2}$, an overlapped clustering of $n$ clusters $P_j$ of $S$, is performed, where:

$$P_j = \{s_k \mid d_t(T_j, T_k) < \varepsilon \ \forall \ s_k \in S, 1 \leq j, k \leq M\}, \tag{6}$$

$$d_t(T_j, T_k) = G_s(|\lambda_j - \lambda_k|) + G_s(d_2(t_{xj}, t_{yj}, t_{xk}, t_{yk})) + G_s(|\omega_j - \omega_k|), \tag{7}$$

In this case, $d_2(t_{xj}, t_{yj}, t_{xk}, t_{yk})$ represents the euclidean distance between the translation components of the involved geometric transformations and $G_s$ is the gaussian function:

$$G_s(t) = e^{\frac{t^2}{2d^2}} \tag{8}$$

This process creates $n$ subsets $P_j$ of $S$, where each $P_j$ in conformed by the closest $s_k$ to each $s_j$, using the distance function $d_t(T_j, T_k)$. Finally, the similarity of the two representations is given by:

$$simFinal(G_{R1}, G_{R2}) = \sum_{s_i \in P_j} sim_i. \tag{9}$$

where $P_j$ is the subset of $S$ with the higher cardinality. This algorithm is based on the idea that if some frequent geometric subgraphs are found between two

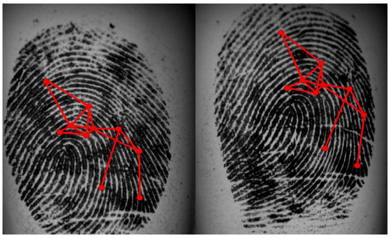
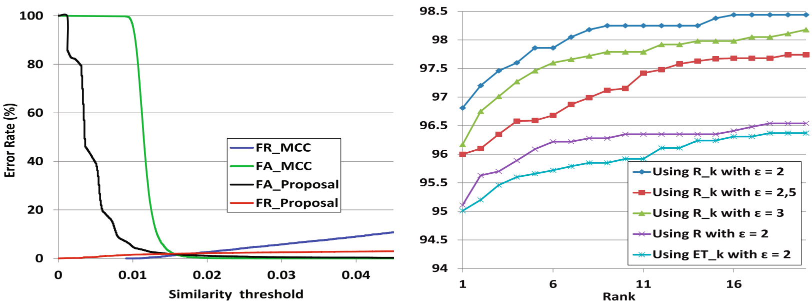


**Fig. 3.** Frequent geometric subgraphs found between two fingerprints of a same finger.

representations of fingerprints, they must have very similar associated geometric transformations. In Figure 3 an example of a relatively large common subgraph found using the mining algorithm can be seen.

## 5 Experimental Evaluation

In the first performed experiment, our proposal was compared with one of the best state-of-the-art approaches, Minutia Cylinder Code (MCC) [8]. For this, False Match Rate (FMR) and False Non-Match Rate (FMR) curves were computed with different similarity thresholds values in the FVC2006 DB2 database, with $\varepsilon = 2$. As can be seen in Figure 4(a), the EER reported by MCC (1.46) is slightly smaller than the obtained with our algorithm (1.90). However, the values of FMR and FNMR of MCC increase more abruptly. This fact makes more difficult the reduction of one measure without affecting the other greatly. Also, Cumulative Match Curves (CMC) were computed by using the last 11 impressions of each finger of the FVC2006 DB2 as queries. Since the accuracy of the fingerprint identification algorithms is affected by the number of possible fingerprint candidates, the comparison dataset was established with the remaining 140 fingerprints of the FVC2006 DB2 and with 31258 rolled impressions contained




(a) FMR and FNMR curves (b) CMC curves

**Fig. 4.** Experimental evaluation.

in the datasets NIST 27, NIST 4 and NIST 14. In Figure 4(b) the CMC curves of our proposal are shown, using different values for $\varepsilon$ and representations. The reported accuracy is higher than 96,8% and 98,4% with $\varepsilon = 2$, in the positions 1 and 20, respectively. As can be seen, the best results are reached using the proposed representation $R_k$. These results are very promising since only minutiae information was used. Also, the mining methods provide many possibilities than can be exploited in the future. Each matching was performed in 28 miliseconds as average, in a PC with a microprocessor i7, 1.7 Ghz and 8 Gb of RAM. These values are lower than the average execution time of the 10 best algorithms reported on FVC2006 competition (53 miliseconds).

## 6 Conclusions

In this work, a new representation of fingerprints based on the fusion of two previous works, was proposed. The result of this fusion is a labeled geometric graph that is able to eliminate most of the noise generated by the absence of minutiae or the distortions found in the fingerprints. Also, a novel matching algorithm between these representations was defined, by using frequent subgraphs mining. In a consolidation step, a coherence criteria between the resulting subgraphs was applied in order to compute the final similarity. The proposed method shows the usefulness of the data mining, specifically the geometric subgraphs mining, in the field of biometrics. Future works are focused on improving *FreqGeom* in order to use other benefits of using graphs in fingerprint matching algorithms.

## References

1. Rajanna, U., Erol, A., Bebis, G.: A comparative study on feature extraction for fingerprint classification and performance improvements using rank-level fusion. Pattern Anal. Appl. **13**(3), 263–272 (2010)
2. Kuramochi, M., Karypis, G.: Discovering frequent geometric subgraphs. Inf. Syst. **32**(8), 1101–1120 (2007)
3. Gago-Alonso, A., Muñoz-Briseño, A., Acosta-Mendoza, N.: A new proposal for graph classification using frequent geometric subgraphs. Data Knowl. Eng. **87**, 243–257 (2013)
4. de Berg, M., van Kreveld, M., Overmars, M., Schwarzkopf, O.: Computational Geometry: Algorithms and Applications, 2nd edn. Springer-Verlag (2000)
5. Bebis, G., Deaconu, T., Georgiopoulos, M.: Fingerprint identification using delaunay triangulation. In: IEEE Int. Conf. on Intell. Inf. Sys., pp. 452–459 (1999)
6. Muñoz-Briseño, A., Gago-Alonso, A., Hernández-Palancar, J.: Fingerprint indexing with bad quality areas. Expert Syst. Appl. **40**(5), 1839–1846 (2013)
7. Gago-Alonso, A., Hernández-Palancar, J., Rodríguez-Reina, E., Muñoz-Briseño, A.: Indexing and retrieving in fingerprint databases under structural distortions. Expert Syst. Appl. **40**(8), 2858–2871 (2013)
8. Cappelli, R.M.F., Maltoni, D.: Minutia cylinder-code: A new representation and matching technique for fingerprint recognition. IEEE Trans. Pattern Anal. Mach. Intell. **32**(12), 2128–2141 (2010)