2017 IEEE/ACS 14th International Conference on Computer Systems and Applications

Fast and Accurate Fingerprint Matching using Expanded Delaunay Triangulation

Mohamed Hedi Ghaddab
MARS Research Lab LR17ES05
ISITCom, University of Sousse, Tunisia
Email:ghaddab.mohamedhedi@gmail.com

Khaled Jouini
MARS Research Lab LR17ES05
ISITCom, University of Sousse, Tunisia
Email: j.khaled@gmail.com

Ouajdi Korbaa
MARS Research Lab LR17ES05
ISITCom, University of Sousse, Tunisia
Email: Ouajdi.Korbaa@centraliens-lille.org

***Abstract*—Efficient and reliable fingerprint matching is crucial for many civilian and forensic applications. Fingerprints are characterized by large intra-class variations ($i.e.$ variability in different impressions from the same finger). This variability manifests itself by the missing or the displacement of genuine minutiae and the detection of spurious minutiae. Displaced, missing and spurious minutiae make fingerprint matching a very challenging pattern recognition problem.**

Although significant improvements have been made in minutiae-based matching, most of minutiae matching algorithms lack of robustness with respect to displaced, missing and spurious minutiae [4]. This paper introduces EDT-C, a new fingerprint matcher based on minutiae triplets. The proposed matcher uses an extended form of Delaunay Triangulation that allows to take into account, not only genuine minutiae, but also spurious, displaced and missing ones. EDT-C characterizes minutiae triplets by a set of innovative geometric features that help in tolerating linear and non-linear distortions. Finally, EDT-C includes some optimizations that allows to quickly consolidate local matchings and filter non-matching minutia triplets. Experiments show that EDT-C has a reasonable computational cost and is far more accurate than its main competitors.

Keywords*—*Minutiae-based fingerprint matching; Expanded Delaunay Triangulation; Triplet descriptor

I. Introduction

Fingerprint-based recognition systems are being extensively used in a broad range of civilian and forensic applications [26]. Fingerprints are used either for verification or identification. The purpose of verification is to corroborate the identity claimed by a person by comparing its captured fingerprint(s) with her own pre-stored template ($i.e.$ one-to-one comparison). The purpose of identification is to establish the identity of a person, given a query impression I and a database of fingerprint templates ($i.e.$ one-to-many comparisons). Fingerprint matching is a crucial step in both verification and identification problems [12].

Minutiae are the points of a fingerprint where its ridge lines end or bifurcate. Due to their high distinctiveness, minutiae-based fingerprint matching is one of the most widely accepted recognition technology [7]. In this work we are only interested in minutiae-based fingerprint matching. Given two sets of minutia points, the aim of minutiae-based matching algorithms is to find the alignment that maximizes the number of matching minutiae and to derive a similarity score accordingly [20].

Several causes make minutiae-based matching a very challenging problem: linear distortions ($e.g.$ translations, scale, rotation, etc.), non-linear distortions (resulting from mapping the three-dimensional shape of a finger onto a two-dimensional image) and other data source damages such as scars, sweat, small overlap and dryness [18]. These factors lead to large intra-class variations ($i.e.$ variability in different impressions from the same finger), which manifests itself by the detection of spurious minutiae and the missing or the displacement/disorientation of genuine minutiae [17].

Minutiae matching can be classified as local and global. Global minutiae matching tries to simultaneously superpose the whole minutia points of a fingerprint I to their mates in a fingerprint T. Local minutiae matching compares I and T according to local structures ($i.e.$ local regions). A local structure is typically formed by a minutia p and minutiae lying within p neighborhood.

Local structures are commonly described by attributes ($e.g.$ angles, distances, etc.) invariant with regard to linear distortion ($e.g.$ translation, rotation, etc.) and can therefore be matched without any prior global alignment. Local structures require also less computational resources than global matching and are more robust to non-linear distortions and partial overlaps [5]. However, by relaxing global spatial relationships, local structures reduce the amount of information available for discriminating fingerprints [17].

To retain the advantages of the global and the local matchings, most of recent minutiae-based algorithms perform a local matching followed by a global matching [20]. The local structures matching quickly and reliably determine pairs of local structures from I and T that match. The global matching step, called the *consolidation step*, checks whether the local matches are consistent at the global level [12].

Many approaches have been proposed to construct local structures. This includes minutiae cylinder [4], texture mixed [21] and minutiae triangles [25], [14], [23], [18]. A minutiae triangle ($i.e.$ triplet) is a structure formed by three minutiae points. Minutiae triangles ($i.e.$ triplets) have the following advantages with regard to other local structures [18]: low computational complexity, tolerance to fingerprint deformations, high discriminative power, embeddability on light architectures, compliancy with interoperability standards (most popular standards are based only on minutiae), etc. In the sequel we only focus on approaches based on minutiae triangles.

Although significant improvement has been made in triplets-based matching, state-of-the-art algorithms lack of

2161-5330/17 $31.00 © 2017 IEEE
DOI 10.1109/AICCSA.2017.33

IEEE computer society

robustness with regard to displaced, missing and spurious minutiae [4].

In this paper we introduce a novel triplets-based fingerprint matcher, called EDT-C, designed to be robust with respect to missing, spurious and displaced minutiae, and to ensure reasonable computational costs. EDT-C follows an hybrid approach mixing local and global matchings. To limit the number of triangles and to select the most discriminating ones, EDT-C only uses the triangles resulting from Delaunay triangulations (as in [1], [25], [14], [23]). As reported in [1], [17], with regard to other topologies, Delaunay triangulation has the best structural stability under random positional perturbations.

Unlike existing approaches which assume that each detected minutia point p is genuine, EDT-C not only considers the case where p is genuine, but also the cases where p is spurious, displaced or missing. More precisely, EDT-C uses the triangles resulting from the Delaunay triangulation of the minutiae set and also those resulting from the Delaunay triangulations that would be obtained if each point of the minutiae set was removed. The resulting triangulation is illustrated in figure 1(d) and is called Expanded Delaunay Triangulation [10].

EDT-C describes the built triangles by a set of geometric and fingerprint features sensitive to reflection and robust with regard to non-linear distortions. In addition, EDT-C speeds up the triplets-matching stage through a modified merge-join and the consolidation stage through a voting step.

The remainder of the present paper is organized as follows. Section II briefly reviews prior related work. Section III introduces EDT-C, our algorithm for minutiae-based fingerprint matching. Section IV presents an experimental study of EDT-C on FVC databases. Section V concludes the paper.

II. Background and Prior Work

A. Minutiae-based matching

In the sequel, we denote a fingerprint acquired by enrollment as *template* T and the query or *input* fingerprint as I. I is said to be *genuine* if it comes from the same finger as T and *impostor* if not [20].

A minutia point is commonly represented by triplet (x, y, θ), where (x, y) represents its spatial location and $\theta \in [0, 2\pi]$ represents its orientation. Each minutia point is typically associated with a tolerance box ($i.e.$ maximum spatial and orientation difference permitted) to compensate spatial changes caused by distortions and extraction errors.

The main idea in minutiae-based matching is to calculate a likeliness score between two fingerprints I and T, according to the number of minutia points from I and T that match. To do so, each minutia point from I is paired with either exactly one minutiae from T, or none at all. This minutiae pairing is commonly preceded by a global alignment of I and T. The alignment consists of rotation, displacement, scaling and other geometric transformations applied to I minutia points.

Finding the correct alignment of I and T is not a trivial task as a minutia point from I may fall within the tolerance box of several minutiae from T [20]. It is widely accepted that the best alignment is the one that maximizes the number of matching minutiae ($i.e.$ minutiae pairs).

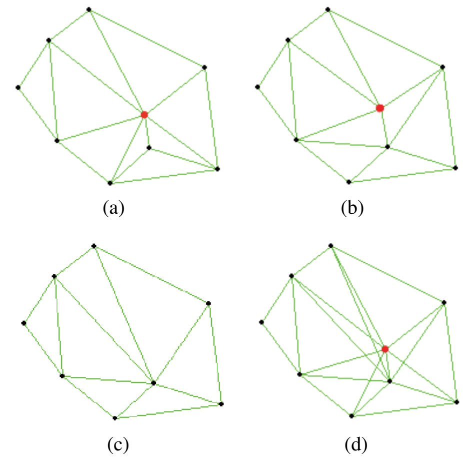


Fig. 1. (a) Delaunay Triangulation. (b) Delaunay Triangulation after the displacement of the red point. (c) Delaunay Triangulation without the red point. (d) Expanded Delaunay Triangulation (EDT). EDT includes the triangles of figures (a) and (c) and most of the triangles of figure (b).

As mentioned before, most of recent minutiae-based fingerprint matching, perform a local minutiae matching followed by a global minutiae matching ($i.e.$ *consolidation*). The purpose of local minutiae matching is to compare local regions ($i.e.$ local structures) from I and T to find out, for each minutia from I, a set of potential matching minutiae from T. The consolidation step tries to derive the transformation(s) that better align I to T based on matching local structures.

B. Triplets-based local structures

Minutiae triplets can be built under different spatial assumptions. [11], [6], [2] consider all possible combinations of triplets in the fingerprint. The use of all possible triplets in a fingerprint containing N minutia points, produces C_N^3 triangles and leads therefore to a high computational cost and to an increased risk of fortuitous (false) matches [5].

[18] uses a Nearest Neighbor (NN) approach. Given a minutia point p, [18] builds all possible triangles formed by p and two of its N nearest neighbors. As reported in [20], NN approaches can be highly affected by missing and spurious minutiae.

In [1], [25], [14] and [23] minutiae triangles are selected from the Delaunay Triangulation (DT) of the entire fingerprint minutiae. DT for a set of points P with cardinality N, is a triangulation $DT(P)$ such that no point in P is inside the circumcircle of each triangle in $DT(P)$ [14]. $DT(P)$ contains $O(N)$ triangles and describes a unique topological structure ($i.e.$ $graph$) of a fingerprint (figure 1(a)). Due to their small number, minutiae triangles can be computed efficiently in $O(NlogN)$ [1]. Furthermore, DT maximize the minimum angle of all angles in the triangulation [1]. These properties are very useful in the context of fingerprint verification. However, as reported in [10], DT is not robust enough with regard to displaced, missing and spurious minutiae, as even a small minutia displacement ($e.g.$ displacement of the red point in figure 1(b)) can severely impacts the DT graph structure.

To minimize the negative effects caused by displaced minutiae, [14] considers a variation of DT named Low-Order

LoD ($i.e.$ order-0 and order-1). An order-r DT of a set of points P, noted $DTr(N)$, is a triangulation such that r points of N are inside the circumcircle of each triangle of $DTr(P)$. [14] proposes to consider all the triangles resulting from the union of $DT_0(P)$ and $DT_1(P)$:

$$LoD(P) = DT_0(P) \cup DT_1(P)$$

While more robust to displaced minutiae than DT, LoD can be highly impacted by missing and spurious minutiae [10]. This illustrated in in figure 1(c) where missing and/or spurious minutiae severely affect the triangulation graph by introducing spurious triangles or eliminating important ones.

In this paper we use a variant of Delaunay Triangulation called Expanded Delaunay Triangulation [10]. Expanded Delaunay Triangulation was first introduced for indexing purpose in [10]. In this paper we show that it can be used as well for fingerprint matching. Expanded Delaunay Triangulation is discussed in detail in section III-A.

C. Consolidation

Consolidation consists in finding the global transformation $(\delta_x, \delta_y, \delta_\theta)$ that best aligns I to T, $i.e.$ that maximizes the number of I and T matching minutiae.

Roughly there exists four consolidation types [17]: single transformation, consensus, incremental and multiple transformations. In single transformation approaches, the transformation aligning the best matching local structures (or a very restricted number of matching local structures) is used to globally align I to T. Single transformation is very efficient as it substantially reduces the search space. However, if the global matching algorithm is not robust enough with regard to displaced and spurious minutiae, the overall robustness will be highly impacted [17].

Consensus consolidation, used in [22], [13], evaluates the consistency of each transformation obtained from a matching pair of local structures with the remaining matching pairs. The main drawback of this approach is that it maximizes the number of matching local structures, rather than the number of matching minutiae [20].

Incremental Transformation [5] arranges local structures of a fingerprint into a graph whose edges represent spatial relationships. The matching is performed by a dual graph traversal. The traversal starts from a given pair, propagates to neighboring nodes, stops when a pair of non-matching nodes is found and returns the number of matching nodes. The process is repeated for each pair of nodes and the maximum number of matched nodes is retained [17]. Incremental transformation is robust but has a high computational cost.

Multiple transformations consolidation [19], [18], [24], [9] consists in testing multiple candidate transformations and: (i) select the one that achieves the highest final score; or (ii) fuse the results obtained by the candidate transformations. Multiple transformations consolidation provides a good trade-off between robustness and efficiency: it is faster than incremental consolidation and more robust than single transformation and consensus consolidations.

III. EDT-C: EDT-BASED MINUTIAE MATCHING

This section introduces EDT-C, a new fingerprint matching algorithm designed to be more tolerant to missing, displaced and spurious minutiae than existing matchers. EDT-C consists of four stages: (i) minutiae triangles generation; (ii) triangles characterization; (iii) triangles pairing; (iii) minutiae pairing; and (iv) consolidation (global matching). The remainder of this section details each of these stages.

A. Local structures determination

EDT-C uses a variant of Delaunay Triangulation, termed *Expanded Delaunay Triangulation* (EDT) [10].

Let P=$\{p_1,\ p_2,\ \ldots,\ p_N\}$ be a set of N minutia points, $DT(P)$ the Delaunay triangulation of P, E the set of $DT(P)$ edges and $G = (P, E)$ the graph described by $DT(P)$.

To be able to define $EDT(P)$, we need first to define the *triangular hulls* [10]. Let p_i be a point of P and $P_i = \{p_j | \{p_i, p_j\} \in E\}$ the set of points of P which are directly connected to p_i in the graph G. The number of points in P_i is the degree of p_i in G. The *triangular hull* H_i of p_i is defined as the Delaunay Triangulation of P_i.

The Expanded Delaunay Triangulation $EDT(P)$ of P is defined as the union of $DT(P)$ triangles and those of the triangular hulls of the points of P.

$$EDT(P) = DT(P) \cup DT(P_1) \cup DT(P_2) \cup \cdots \cup DT(P_N) \quad (1)$$

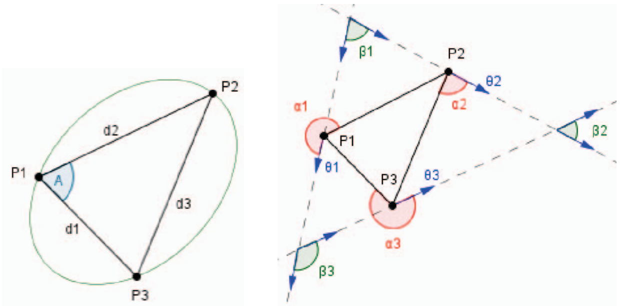
Intuitively, $EDT(P)$ contains, in addition to the triangles of $DT(P)$, all the triangles that would be obtained if each of the points of P was eliminated (figure 1(d)). As $DT(P)$, $EDT(P)$ yields a unique topological structure.

$DT(P)$ based approaches assume that each detected minutia p_i is genuine and do not consider the case where p_i is spurious/missing or displaced. In our approach we consider both cases : $DT(P)$ allows to include all the triangles that would be obtained if p_i was genuine, where $DT(P_i)$ allows to include all the triangles that would be obtained if p_i was spurious/missing or displaced. Accordingly, our approach is expected to be more tolerant to displaced, missing and spurious minutiae than approaches based on Delaunay Triangulation.

The counterpart of EDT robustness is that it constructs more triangles than DT. However, as demonstrated in [10], the number of triangles of $EDT(P)$ remains linear with respect to N and in all cases lower than $|EDT(P)| < 13N - 25$.

B. Triplet descriptor

Each minutiae triangle is represented by a descriptor taking the form of a vector of numerical values (features). Triangles matching and pairing are based on their associated descriptors. The descriptor that we associate with each minutiae triangle follows the recommendations of [18] and includes geometric features that describe the shape of the triangle (figure 2(a)) and features related to the minutiae points that form the triangle (figure 2(b)).



(a) (b)

Fig. 2. (a) Triangle geometric features (elongation of the Steiner circumellipse, cosine of the greatest angle and the perimeter). (b) Fingerprint features [18].

1) Triangle geometric features: Several shape measures can be extracted from a triangle (*e.g.* perimeter, lengths of sides, angles, etc.). With respect to the lengths of sides, angles and relative/derived measures are less sensitive to distortions, but are also less discriminating.

In contrast with most of existing approaches, in our approach we do not describe a minutiae triangle by the lengths of its sides. Instead, we consider a mix of derived and relative measures. To this end, we conducted an experimental study on FVC databases using different combinations of shape measures and retained the one with the best results.

In our approach we describe the shape of a triangle Δ by: the elongation of Δ's Steiner circumellipse, $\epsilon(\Delta)$, the cosine of Δ's greatest angle, $CosA(\Delta)$ and the perimeter of Δ, $sp(\Delta)$. We should notice here that $\epsilon(\Delta)$ and $CosA(\Delta)$ remain invariant if Δ is rotated or scaled uniformly.

Let $d_1 \leq d_2 \leq d_3$ be the lengths of Δ's sides. $\epsilon(\Delta)$, can be calculated using the following formula [8] :

$$\epsilon(\Delta) = \frac{\sigma_1}{\sigma_2}, \text{ where}$$
$$\sigma_1 = \frac{\sqrt{d_1^2+d_2^2+d_3^2+2Z}}{3}, \ \sigma_2 = \frac{\sqrt{d_1^2+d_2^2+d_3^2-2Z}}{3} \text{ and}$$
$$Z = \sqrt{d_1^4 + d_2^4 + d_3^4 - d_1^2 d_2^2 - d_1^2 d_3^2 - d_2^2 d_3^2}$$

The cosine of Δ greatest angle can be calculated with the following formula :

$$cosA(\Delta) = min\left(\frac{d_1^2+d_2^2-d_3^2}{2d_1^2 d_2^2}, \frac{d_1^2+d_3^2-d_2^2}{2d_1^2 d_3^2}, \frac{d_2^2+d_3^2-d_1^2}{2d_2^2 d_3^2}\right)$$

2) Minutiae features: Triangle measures taken alone are not sufficient to uniquely identify a minutiae triplet, as they only consider minutiae spatial locations and not their orientations θ_i [18]. As in [18] we consider the following additional minutiae features.

- $\alpha_{i\in[1,3]}$: the angles required to rotate the direction θ_i of a minutia p_i to superpose it to the vector associated with the minutiae (p_i, p_{i+1}).
- $\beta_{i\in[1,3]}$: the angle required to rotate the direction θ_i of a minutia p_i to superpose it to the direction θ_{i+1} of the minutia p_{i+1}.

C. Local matching

As in most existing triangles-based matching approaches, we perform the local matching in two steps : triangles matching and minutiae pairing.

1) Triangles matching: Let I and T be two fingerprints to be compared. The aim of the triangles matching stage is to find the list A of pairs of triangles from I and T that are similar, with respect to a similarity function SL and a threshold Th_{SL} :

$$A = \{(\Delta^I, \Delta^T) | \Delta^I \in I, \, \Delta^T \in T \, and \, SL(\Delta^I, \Delta^T) \geq Th_{SL}\} \tag{2}$$

Let $S_\epsilon(\Delta^I, \Delta^T)$, $S_{CosA}(\Delta^I, \Delta^T)$, $S_{sp}(\Delta^I, \Delta^T)$, $S_\alpha(\Delta^I, \Delta^T)$ and $S_\beta(\Delta^I, \Delta^T)$ be, respectively, the functions allowing to calculate the similarity between elongations, cosines of the largest angles, the perimeters, the angles α and β of two triangles Δ^I and Δ^T.

$S_\epsilon(\Delta^I, \Delta^T)$, $S_{CosA}(\Delta^I, \Delta^T)$ and $S_{sp}(\Delta^I, \Delta^T)$ are respectively calculated according to the following self-explanatory formulas.

$$S_\epsilon(\Delta^I, \Delta^T) = \begin{cases} 0 \ \ if \ \ |\epsilon(\Delta^I) - \epsilon(\Delta^T)| \ > \ Thr_\epsilon \\ 1 - \frac{|\epsilon(\Delta^I) - \epsilon(\Delta^T)|}{Thr_\epsilon}, \qquad otherwise \end{cases} \tag{3}$$

$$S_{cosA}(\Delta^I, \Delta^T) = \begin{cases} 0 \ if \ |cosA(\Delta^I) - cosA(\Delta^T)| > Thr_{cosA} \\ 1 - \frac{|cosA(\Delta^I) - cosA(\Delta^T)|}{Thr_{cosA}}, \quad otherwise \end{cases} \tag{4}$$

$$S_{sp}(\Delta^I, \Delta^T) = \begin{cases} 0 \ \ if \ \ |sp(\Delta^I) - sp(\Delta^T)| \ > \ Thr_{sp} \\ 1 - \frac{|sp(\Delta^I) - sp(\Delta^T)|}{Thr_{sp}}, \qquad otherwise \end{cases} \tag{5}$$

$$S_\alpha(\Delta^I, \Delta^T) = \begin{cases} 0 \ if \ \exists i \in [1,3] \ | \ \left(ad(\alpha_i^I, \alpha_i^T) \ > \ Thr_a\right) \\ 1 - \frac{max_{i\in[1,3]}\left\{ad(\alpha_i^I, \alpha_i^T)\right\}}{Thr_a}, \quad otherwise \end{cases} \tag{6}$$

$$S_\beta(\Delta^I, \Delta^T) = \begin{cases} 0 \ if \ \exists i \in [1,3] \ | \ \left(ad(\beta_i^I, \beta_i^T) \ > \ Thr_a\right) \\ 1 - \frac{max_{i\in[1,3]}\left\{ad(\beta_i^I, \beta_i^T)\right\}}{Thr_a}, \quad otherwise \end{cases} \tag{7}$$

The tolerance boxes defined by the thresholds (Thr_ϵ, Thr_{cos}, Thr_{sp} and Thr_a) are necessary to compensate the unavoidable errors made during fingerprint acquisition. As in [18], for two given angles a_1 and a_2, we use the function $ad(a_1, a_2) = min\left(|a_1 - a_2|, 2\pi - |a_1 - a_2|\right)$ to compute the minimum angle required to superpose two vectors with the same origin and angles a_1 and a_2, respectively.

The similarity between two triangles $\Delta^I \in I$ and $\Delta^T \in T$ is calculated by the following formula.

$$SL(\Delta^I,\Delta^T) = \begin{cases} 0 \; if \; S_\epsilon(\Delta^I,\Delta^T)=0 \vee S_{cosA}(\Delta^I,\Delta^T)=0 \\ \qquad \vee S_{sp}(\Delta^I,\Delta^T)=0 \vee S_\alpha(\Delta^I,\Delta^T)=0 \\ \qquad \vee S_\beta(\Delta^I,\Delta^T)=0 \\ 1-\left(1-S_\epsilon(\Delta^I,\Delta^T)\right)\left(1-S_{cosA}(\Delta^I,\Delta^T)\right) \\ \quad \left(1-S_{sp}(\Delta^I,\Delta^T)\right)\left(1-S_\alpha(\Delta^I,\Delta^T)\right) \\ \quad \left(1-S_\beta(\Delta^I,\Delta^T)\right) \qquad otherwise \end{cases} \tag{8}$$

If $SL(\Delta^I,\Delta^T) \leq Th_{SL}$, the pair (Δ^I,Δ^T) is added to the list A. Otherwise, the pair is rejected. To speed up local structures matching, the triangles of I and those of T are maintained sorted according to their perimeters and a slightly modified merge-join is used to determine matching triangles.

2) Local minutiae pairing: The goal of this stage is to build a list M containing the pairs of minutiae from I and T that match : $M = \{((p,q),v) \,|\, p \in I, \, q \in T \; and \; v > 0\}$, where v corresponds to the number of triangles where p and q match. In other words, M is used to accumulate votes or evidences on minutiae matching.

M is constructed as follows :

1. For each pair of matching triangles $(\Delta^I,\Delta^T) \in A$, we consider the minutiae triplets forming them : $(p_1,p_2,p_3) \in \Delta^I$ and $(q_1,q_2,q_3) \in \Delta^T$.
2. For each minutiae mates (p_i,q_j), if p_i and q_j match and (p_i,q_j) appears in M, then the number of occurrences ($i.e.$ votes) associated to (p_i,q_j) is incremented by one.
3. If p_i and q_j match and (p_i,q_j) does not appear in M, $((p_i,q_j),1)$ is added to M

D. Consolidation

Consolidation is the ultimate stage in our approach. EDT-C uses a slightly modified multiple transformations consolidation, aiming at achieving a good trade off between robustness and efficiency. In our approach we consider the m minutiae pairs of M that have the highest v ($i.e.$ highest numbers of votes). Each transformation that align a minutiae pair from m, is considered as a candidate transformation. Rather than testing each candidate transformation on the entire set of minutiae, we only tested it on the m pairs with the highest votes. The transformation that better align the m pairs, is then applied at the global level and the similarity score is calculated accordingly.

Let $|I|$ and $|T|$ be respectively the number of minutia points in the enrolled and the query fingerprints, and k the number of matching minutiae. The similarity score is calculated using the formula [17][20].

$$Score(I,T) = \frac{k}{(|I|+|T|)/2} \tag{9}$$

IV. Experimental results

FVC databases [15], [16], [3] are commonly used as benchmarks for evaluating fingerprint verification algorithms. To validate our work, we used FVC protocol and databases to compute the following accuracy indicators:

Triangulation	EER%	FMR100%	FMR1000%	zeroFMR%	Temps(ms)
NN	2,500	3,179	4,893	6,750	4,059
DT	2,358	3,143	4,679	5,321	**1,195**
LoD	2,256	2,821	4,107	6,500	1,703
EDT	**1,572**	**1,893**	**2,893**	**3,357**	4,378

TABLE I. Experimental results on database DB1_A of FVC2002.

Algorithm	Triplets	Shape measures	Consolidation
PN	Delaunay Triangulation	Lengths of sides	Multiple transformations
M3gl	N nearest neighboring minutiae	Lengths of sides	Multiple transformations
EDT-C	Expanded Delaunay Triangulation	Elongation of the $Steiner$ circumellipse, Cosine of the greatest angle and the perimeter	Multiple transformations (only on the m matching pairs with the highest votes)

TABLE II. Main differences between PN[19], M3gl[18] and EDT-C

- FMR: *False Match Rate*, rate of incorrectly matched fingerprints. Each score threshold has an associated FMR.
- FNMR: *False Non-Match Rate*, rate of corresponding fingerprints that are incorrectly considered different. Each score threshold has an associated FNMR.
- EER: *Equal-Error Rate*, score threshold where FMR and FNMR are equal.
- FMR100: lowest achievable FNMR for an FMR $\leq$ 1%.
- FMR1000: lowest achievable FNMR for an FMR $\leq$ 0.1%.
- ZeroFMR: lowest FNMR at which no false matches occur.
- ROC: curve that plots the FNMR as function of the FMR.

Experiments were performed on a dedicated dual core i3-2330M system. Each core offers a base speed of 2.2 GHz. The computer features 4 GB main memory (DDR3-1066/1333) and 3 MB L3 cache. The indicator Time is used in the sequel to refer to the average matching time in milliseconds.

DB	Algorithm	EER%	FMR100%	FMR1000%	ZeroFMR%	Time(ms)
DB1_A	PN	3,657	4,679	7,071	12,750	50,662
	M3gl	2,406	3,071	5,286	6,964	**3,296**
	EDT-C	**1,572**	**1,893**	**2,893**	**3,357**	4,378
DB2_A	PN	2,272	2,750	3,750	4,286	61,088
	M3gl	1,716	1,893	3,036	4,643	**3,400**
	EDT-C	**1,122**	**1,250**	**1,964**	**2,500**	4,341
DB3_A	PN	5,944	8,357	12,500	16,107	182,105
	M3gl	5,726	8,929	12,536	13,893	**6,201**
	EDT-C	**4,228**	**6,036**	**7,750**	**13,464**	8,656
DB4_A	PN	5,051	5,857	7,250	7,964	15,730
	M3gl	2,498	3,286	5,571	**8,179**	1,706
	EDT-C	**2,074**	**2,714**	**4,893**	9,250	**1,403**

TABLE III. Experimental results on FVC2000 databases

DB	Algorithm	EER%	FMR100%	FMR1000%	ZeroFMR%	Time(ms)
	PN	1,355	1,393	2,536	4,214	39,885
DB1_A	M3gl	0,995	**1,000**	1,714	3,643	**2,833**
	EDT-C	**0,955**	**1,000**	**1,464**	**1,964**	4,782
	PN	1,622	1,821	2,857	3,929	107,707
DB2_A	M3gl	0,902	1,036	1,500	1,786	**4,245**
	EDT-C	**0,632**	**0,679**	**0,821**	**1,143**	8,514
	PN	5,210	7,250	9,929	15,143	14,731
DB3_A	M3gl	4,142	5,464	9,429	**11,286**	**1,335**
	EDT-C	**3,569**	**5,000**	**7,250**	13,357	1,662
	PN	2,426	3,179	6,500	9,179	22,753
DB4_A	M3gl	2,046	2,643	5,714	**8,036**	**2,654**
	EDT-C	**1,305**	**1,429**	**4,000**	8,357	3,026

TABLE IV. EXPERIMENTAL RESULTS ON FVC2002 DATABASES

DB	Algorithm	EER%	FMR100%	FMR1000%	ZeroFMR%	Time(ms)
	PN	7,640	11,179	15,214	19,321	44,006
DB1_A	M3gl	**4,434**	**7,893**	14,893	18,357	**3,264**
	EDT-C	5,125	8,393	**12,964**	**14,500**	5,846
	PN	8,026	10,036	12,607	15,500	35,590
DB2_A	M3gl	5,838	8,321	11,429	13,607	**3,444**
	EDT-C	**5,059**	**6,536**	**9,571**	**10,071**	4,269
	PN	5,825	7,750	15,714	18,286	89,485
DB3_A	M3gl	4,150	6,536	10,357	12,536	**5,261**
	EDT-C	**3,645**	**4,786**	**7,107**	**12,500**	8,024
	PN	4,094	5,143	7,071	10,643	45,537
DB4_A	M3gl	2,566	3,607	5,786	6,571	**2,865**
	EDT-C	**2,148**	**2,964**	**5,000**	**5,857**	4,184

TABLE V. EXPERIMENTAL RESULTS ON FVC2004 DATABASES

We conducted two sets of experiments. The aim of the first one is to bring to light the relevancy of using Expanded Delaunay Triangulation (EDT) for fingerprint verification purpose (without any other consideration). The aim of the second set is to compare EDT-C to its main competitors. To show the relevancy of using Expanded Delaunay Triangulation, we implemented our approach with alternative triplets generation methods, namely : the N nearest neighbors (NN), the Delaunay Triangulation (DT) and the combined Low-order Delaunay Triangulations (LoD). The results obtained on the database DB1_A of FVC 2002 are reported in table I. As expected, although EDT consumes a slightly higher execution time, it is by far more accurate than any other topology.

DB	Algorithm	EER%	FMR100%	FMR1000%	ZeroFMR%	Time(ms)
	PN	0,975	0,942	1,764	3,452	196,373
DB2_A	M3gl	0,933	0,920	1,623	2,197	**4,872**
	EDT-C	**0,448**	**0,346**	**0,671**	**1,028**	8,786
	PN	6,161	7,695	10,584	16,494	98,584
DB3_A	M3gl	4,898	7,998	16,786	29,329	**3,742**
	EDT-C	**4,36**	**5,812**	**8,647**	**14,751**	5,507
	PN	4,017	5,054	9,058	12,619	56,258
DB4_A	M3gl	3,686	5,141	8,885	12,403	**2,609**
	EDT-C	**2,436**	**2,998**	**5,108**	**8,626**	3,576

TABLE VI. EXPERIMENTAL RESULTS ON FVC2006 DATABASES

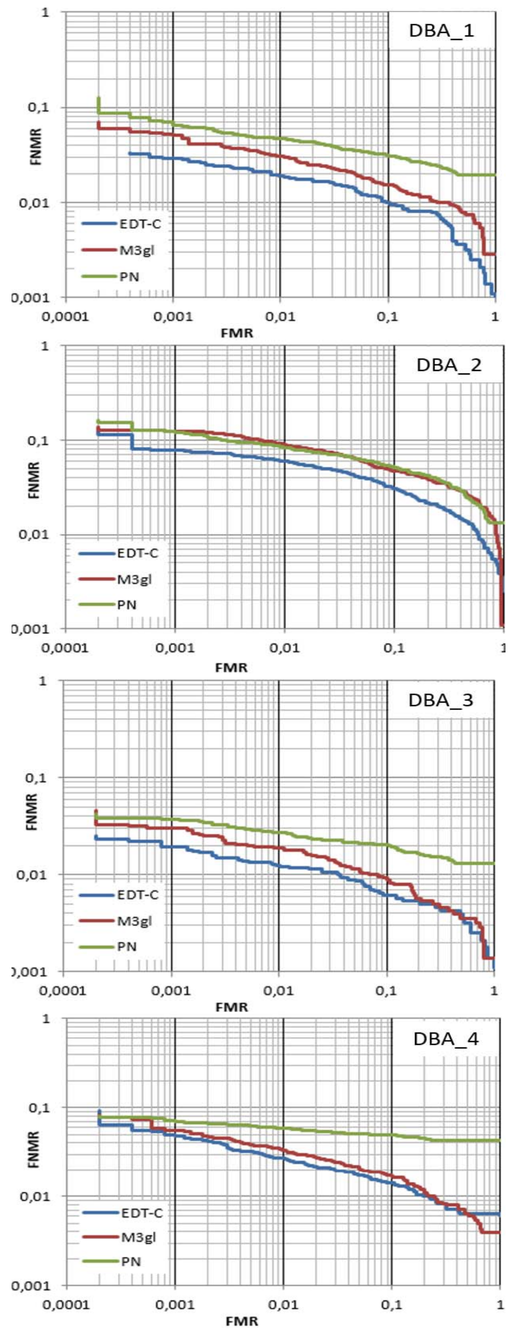


Fig. 3. ROC curves obtained in FVC2000 databases

A large number of experiments have been performed to compare EDT-C to its competitors. Due to the lack of space, only few results are presented herein. In our experimental comparison we include the algorithm proposed by Parziale and Niel [19] (PN) and the algorithm proposed by Medina-Prez $et\ al.$ [18] (M3gl). We implemented PN because it is one of the most popular algorithm based on minutiae triplets in the literature. We implemented M3gl because it proves itself as one of the best triplets-based matchers [18].

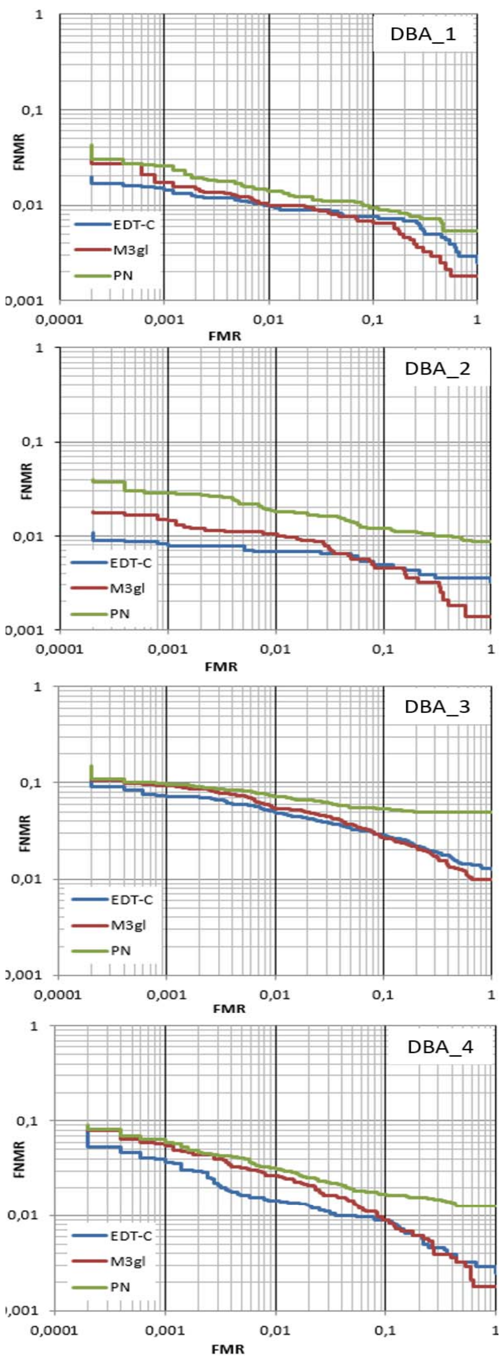


Fig. 4. ROC curves obtained in FVC2002 databases

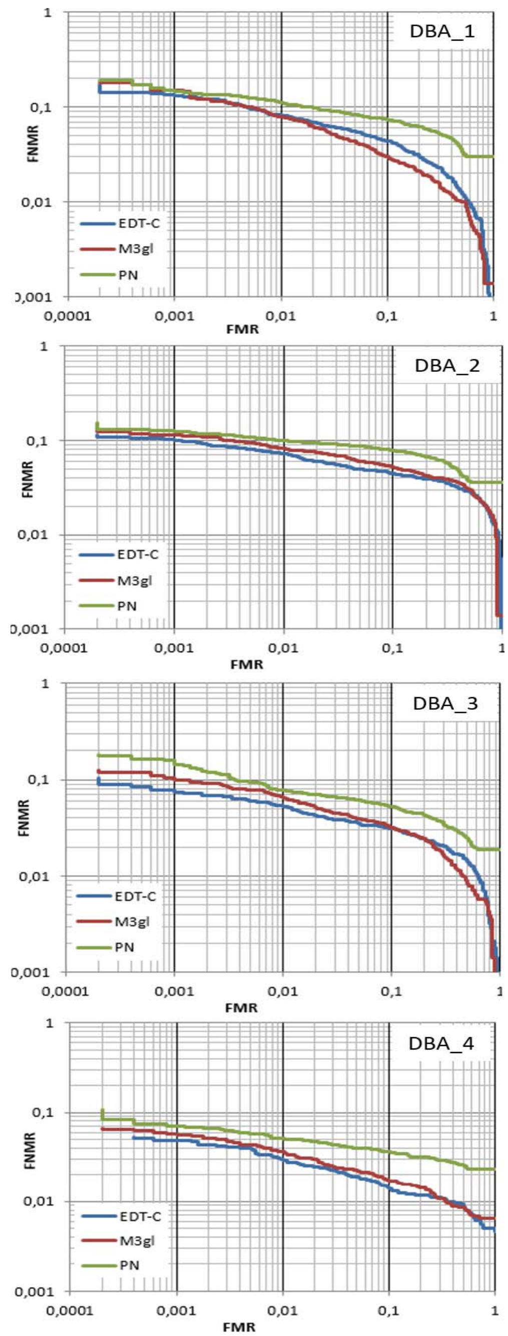


Fig. 5. ROC curves obtained in FVC2004 databases

EDT-C is very close to PN and M3gl in design philosophy. As EDT-C, PN and M3gl: (i) only use standard minutiae features ($i.e.$ spatial location, orientation and type) and, hence, are suitable for systems based on interoperability standards; (ii) use minutiae triangles as local structures; and (iii) perform a local matching and consolidate it by a global matching. PN and M3gl use respectively DT and NN approaches to construct minutiae triangles. Table II summarizes the main properties of the three approaches. A salient feature of M3gl is that it arranges minutiae in clockwise sense and performs the three possible rotations of the triplets to ensure invariance to the order of minutiae and sensitivity to the reflection of minutiae triplets [18]. As illustrated in tables III, IV, V and VI, EDT-C and M3gl outperform PN in almost all databases. This can be explained by the fact that PN uses Delaunay Triangulation and is hence highly impacted by displaced, missing and spurious minutiae. M3gl is slightly faster than EDT-C. However, EDT-C is more accurate in almost all databases. This result is confirmed in figures IV and IV and IV which show that EDT-C achieves lower FNMR for most of the FMR values.

The good performances of EDT-C and its slight additional computational cost can be explained by the fact that Expanded

Delaunay Triangulation generates more triangles than DT and NN, but helps in better support missing, spurious and displaced minutiae.

V. Conclusion and Future Work

Displaced, missing and spurious minutiae make fingerprint matching a very challenging pattern recognition problem. In this paper we present EDT-C, a new minutiae-based matcher. As most of recent algorithms, EDT-C performs a local minutiae matching followed by a consolidation stage. In contrast with existing approaches, EDT-C uses an Expanded Delaunay Triangulation (EDT) to yield local structures. As shown in this paper, EDT is very robust with regard to missing and spurious minutiae. The robustness of EDT can be explained by the fact that it constructs local structures around a minutiae p by taking into account the case where p is genuine as well as the case where it is not. EDT-C uses also a set of innovative features to characterize minutiae triplets. The considered features are more tolerant to displaced minutiae than the usually used lengths of sides. Finally, to fast the entire matching process, EDT-C uses a modified merge-join to speed up local structures matching and a lightweight multiple transformations consolidation.

Experiments conducted on FVC databases show that, while EDT-C has comparable execution time with other triplets-based algorithms, it is by far more accurate. Encouraged by the good behavior of EDT-C, our next goal is to adapt it to latent fingerprint identification.

Acknowledgements : This research paper was performed in the framework of a MOBIDOC thesis funded by the European Union under the program PASRI managed by the ANPR.

References

[1] George Bebis, Taisa Deaconu, and Michael Georgiopoulos. Fingerprint identification using delaunay triangulation. In *Information Intelligence and Systems, 1999. Proceedings. 1999 International Conference on*, pages 452–459. IEEE, 1999.

[2] Soma Biswas, Nalini K Ratha, Gaurav Aggarwal, and Jonathan Connell. Exploring ridge curvature for fingerprint indexing. In *Biometrics: Theory, Applications and Systems, 2008. BTAS 2008. 2nd IEEE International Conference on*, pages 1–6. IEEE, 2008.

[3] Raffaele Cappelli, Matteo Ferrara, Annalisa Franco, and Davide Maltoni. Fingerprint verification competition 2006. *Biometric Technology Today*, 15(7):7–9, 2007.

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

[5] Sharat Chikkerur and Nalini Ratha. Impact of singular point detection on fingerprint matching performance. In *Fourth IEEE Workshop on Automatic Identification Advanced Technologies (AutoID'05)*, pages 207–212. IEEE, 2005.

[6] Kyoungtaek Choi, Dongjae Lee, Sanghoon Lee, and Jaihie Kim. An improved fingerprint indexing algorithm based on the triplet approach. In *International Conference on Audio-and Video-Based Biometric Person Authentication*, pages 584–591. Springer, 2003.

[7] Sarat C Dass. Fingerprint-based recognition. *International Statistical Review*, 81(2):175–187, 2013.

[8] Gerald Farin. Shape measures for triangles. *IEEE transactions on visualization and computer graphics*, 18(1):43–46, 2012.

[9] Jianjiang Feng. Combining minutiae descriptors for fingerprint matching. *Pattern Recognition*, 41(1):342–352, 2008.

[10] AndréS Gago-Alonso, José HernáNdez-Palancar, Ernesto RodríGuez-Reina, and Alfredo MuñOz-BriseñO. Indexing and retrieving in fingerprint databases under structural distortions. *Expert Systems with Applications*, 40(8):2858–2871, 2013.

[11] Robert S Germain, Andrea Califano, Scott Colville, et al. Fingerprint matching using transformation parameter clustering. *IEEE Computational Science and Engineering*, 4(4):42–49, 1997.

[12] Mohamed Hedi Ghaddab, Khaled Jouini, and Ouajdi Korbaa. Fusion de minuties pour une reconnaissance efficiente des empreintes digitales. In *Colloque sur l'Optimisation et les Systmes d'Information*, 2017.

[13] Xiaoguang He, Jie Tian, Liang Li, Yuliang He, and Xin Yang. Modeling and analysis of local comprehensive minutia relation for fingerprint matching. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(5):1204–1211, 2007.

[14] Xuefeng Liang, Arijit Bishnu, and Tetsuo Asano. A robust fingerprint indexing scheme using minutia neighborhood structure and low-order delaunay triangles. *IEEE Transactions on Information Forensics and Security*, 2(4):721–733, 2007.

[15] Dario Maio, Davide Maltoni, Raffaele Cappelli, James L Wayman, and Anil K Jain. Fvc2002: Second fingerprint verification competition. In *Pattern recognition, 2002. Proceedings. 16th international conference on*, volume 3, pages 811–814. IEEE, 2002.

[16] Dario Maio, Davide Maltoni, Raffaele Cappelli, Jim L Wayman, and Anil K Jain. Fvc2004: third fingerprint verification competition. In *Biometric Authentication*, pages 1–7. Springer, 2004.

[17] Davide Maltoni, Dario Maio, Anil Jain, and Salil Prabhakar. *Handbook of fingerprint recognition*. Springer Science & Business Media, 2009.

[18] Miguel Angel Medina-Pérez, Milton García-Borroto, Andres Eduardo Gutierrez-Rodríguez, and Leopoldo Altamirano-Robles. Improving fingerprint verification using minutiae triplets. *Sensors*, 12(3):3418–3437, 2012.

[19] Giuseppe Parziale and Albert Niel. A fingerprint matching using minutiae triangulation. In *Biometric Authentication*, pages 241–248. Springer, 2004.

[20] Daniel Peralta, Mikel Galar, Isaac Triguero, Daniel Paternain, Salvador García, Edurne Barrenechea, José M Benítez, Humberto Bustince, and Francisco Herrera. A survey on fingerprint minutiae-based local matching for verification and identification: Taxonomy and experimental evaluation. *Information Sciences*, 315:67–87, 2015.

[21] Jin Qi and Yangsheng Wang. A robust fingerprint matching method. *Pattern Recognition*, 38(10):1665–1671, 2005.

[22] Nalini K Ratha, Ruud M Bolle, Vinayaka D Pandit, and Vaibhav Vaish. Robust fingerprint authentication using local structural similarity. In *Applications of Computer Vision, 2000, Fifth IEEE Workshop on.*, pages 29–34. IEEE, 2000.

[23] Arun Ross and Rajiv Mukherjee. Augmenting ridge curves with minutiae triplets for fingerprint indexing. In *Defense and Security Symposium*, pages 65390C–65390C. International Society for Optics and Photonics, 2007.

[24] Lifeng Sha, Feng Zhao, and Xiaoou Tang. A two-stage fusion scheme using multiple fingerprint impressions. In *Proceedings of the International Conference on Image Processing, San Antonio, Texas, USA*, pages 385–388, 2007.

[25] Tamer Uz, George Bebis, Ali Erol, and Salil Prabhakar. Minutiae-based template synthesis and matching using hierarchical delaunay triangulations. In *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, pages 1–8. IEEE, 2007.

[26] Akhil Vij and Anoop Namboodiri. Learning minutiae neighborhoods: A new binary representation for matching fingerprints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 64–69, 2014.