

---

# Adaptation of Rigid Registration Algorithm to the Fingerprints Identification

Mostafa Boutahri\*, Samir Zerriouh, Said El Yamani, Abdenbi Bouzid, Ahmed Roukhe

Optronic and Information Treatment Team, Atomic, Mechanical, Photonic and Energy Laboratory, Faculty of Science, Moulay Ismail University, Zitoune, Meknès, Morocco

## Email address:

boumost-1980@hotmail.fr (M. Boutahri)

## To cite this article:

Mostafa Boutahri, Samir Zerriouh, Said El Yamani, Abdenbi Bouzid, Ahmed Roukhe. Adaptation of Rigid Registration Algorithm to the Fingerprints Identification. *American Journal of Software Engineering and Applications*. Vol. 4, No. 6, 2015, pp. 107-114.

doi: 10.11648/j.ajsea.20150406.12

---

**Abstract:** In this paper, we present an automated system for the recognition and identification of fingerprints based on rigid registration algorithms. Indeed, after preprocessing carried on a fingerprint database collected in the laboratory, we have built maps of minutiae for each fingerprint. Subsequently, we applied a rigid registration algorithm based on iterative search for closed points ICP (Iterative Closest Point), which allowed us to compare shifted fingerprints serving as test with the fingerprints of the reference database. This comparison gives convincing results and shows high accuracy.

**Keywords:** Recognition, Identification, Fingerprints, Rigid Registration, Minutiae, ICP

---

## 1. Introduction

Nowadays, authentication becomes one of the essential points at the security access controls in informatics systems. It is based on the recognition that verifies the presumed user identities [1].

The identification is generally based on several criteria and according to the extent of the investigation. It may be an identification by DNA search, by using images from surveillance cameras, for removal of odors and scents or most often by use of fingerprints [2]. This process remains in our day, the most reliable and most cost especially that the police have fingerprint database of all citizens holder a national identity card.

The important progress experienced by the fingerprint verification area in recent years have required new image processing tools, especially for multimodal image (from different sources). The registration algorithms are part of these tools. There are many, both within the rigid or non-rigid registration. However, these algorithms are difficult to validate in the absence of a reference standard for verifying the fields obtained.

In this context, our work proposes adapting rigid registration algorithm variants, allowing a more effective comparison. Indeed, after scanning fingerprints, object of study, we conducted a binarization and thinning of

fingerprints to finally build the maps corresponding to each minutiae fingerprint. Subsequently, we established a mathematical modeling of the registration problem to fit these data based on iterative method ICP. The simulation results obtained after several tests resulted in the recognition of fingerprint test in a reference database with largely optimized errors.

## 2. Extraction of Fingerprint's Biometric - Signature

To build our database, we proceeded to the taking of fingerprints on different people; these fingerprint images were enhanced by a pretreatment process to extract the corresponding signature to each fingerprint. Thus, these signatures are stored in a reference database.

### 2.1. Preprocessing of Captured Fingerprints

In this step, we proceeded the acquisition of fingerprints of different people with a scanner of type Ko-UF100 500- dpi, performed in our laboratory. The resulting images are hence stored as grayscale with the appropriate format. However these images often present a noise during the acquisition phase that require a adapted filtering, which the goal is to strengthen the contrast by eliminating redundant points [3].

Then, these images are binarized by thresholding process

that associates the number 0 to the pixel if its value is below a predetermined threshold value and the number 1 if the pixel value is equal to or greater than this threshold value. This last, varies according to the quality of the captured image and was therefore set to a gray level equal to 160.

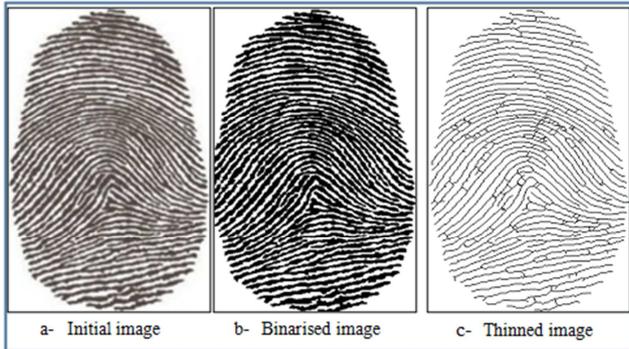


Figure 1. Fingerprint pretreatment process.

To facilitate its exploitation, the image must undergo a thinning treatment that consist to obtain a schematic image of the fingerprint, in which all the lines have the thickness of a pixel [4] [5].

The simulation results was carried out using Matlab software-R2014a (Figure. 1).

### 2.2. Extraction of Minutiae

To extract the fingerprint minutiae (bifurcations and endings), the calculation method used allows to detect them according to the neighborhood of each pixel following the continuity or discontinuity of the different lines [6] [7]. Once this step is completed, false minutiae are eliminated from the picture for finally have a two-dimensional minutiae map, containing the two values 0 or 1 (where 1 indicates the presence of the minutiae and 0 his absence) [8]. (Figure 2)

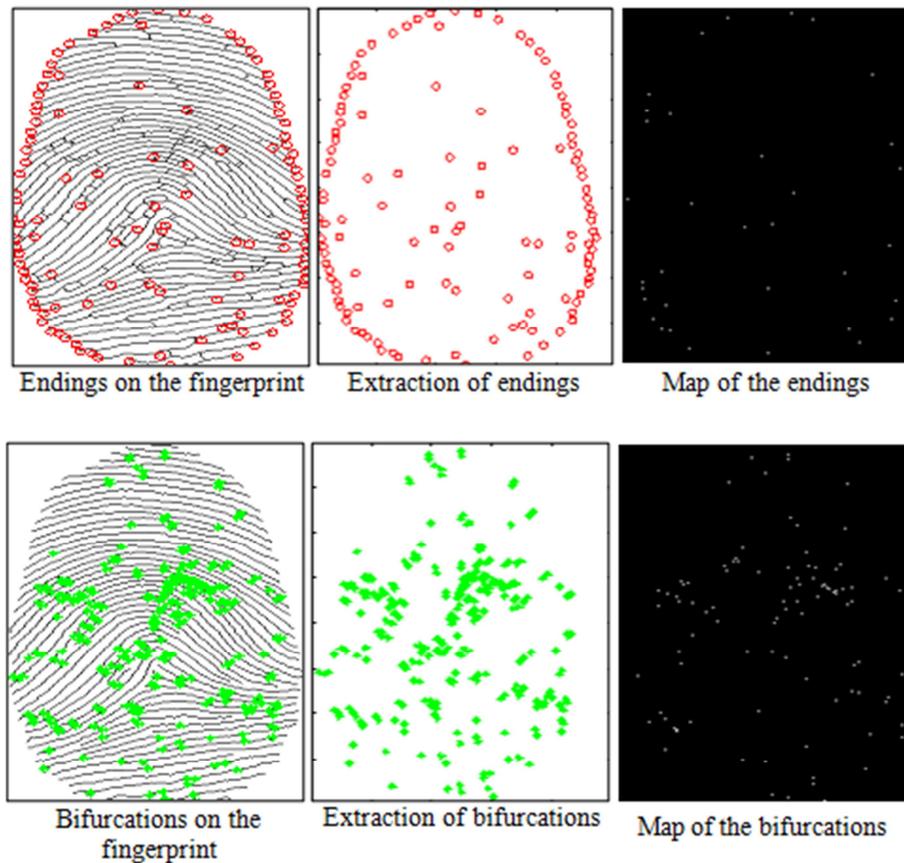


Figure 2. Process of minutiae extraction.

### 2.3. Data validation

The biometric signatures obtained are stored in a reference database. During the identification and authentication phase, a biometric signature is extracted from the fingerprint, object of recognition, and is compared to the signatures of the reference database using the iterative algorithm of fingerprints rigid registration.

## 3. Registration of Fingerprints Images by ICP

The ICP algorithm applied on fingerprints is based on the iterative matching of minutiae of a test fingerprint with their nearest neighbors on the next reference fingerprint.

Meanwhile, a least squares technique is used to estimate the transformation from these correspondences. Indeed, at the end of each iteration, the algorithm provides a list of matched points, and an estimate of the registration transformation, this transformation is used in the next iteration for updating the list of matched points. They used their towers to calculate a new estimate of the transformation. These steps are repeated until convergence of the algorithm.

**3.1. Modeling of the Problem**

The registration problem can be laid more formally by introducing the different notations used throughout this work and describing the general principle of the registration procedure (figure 3).

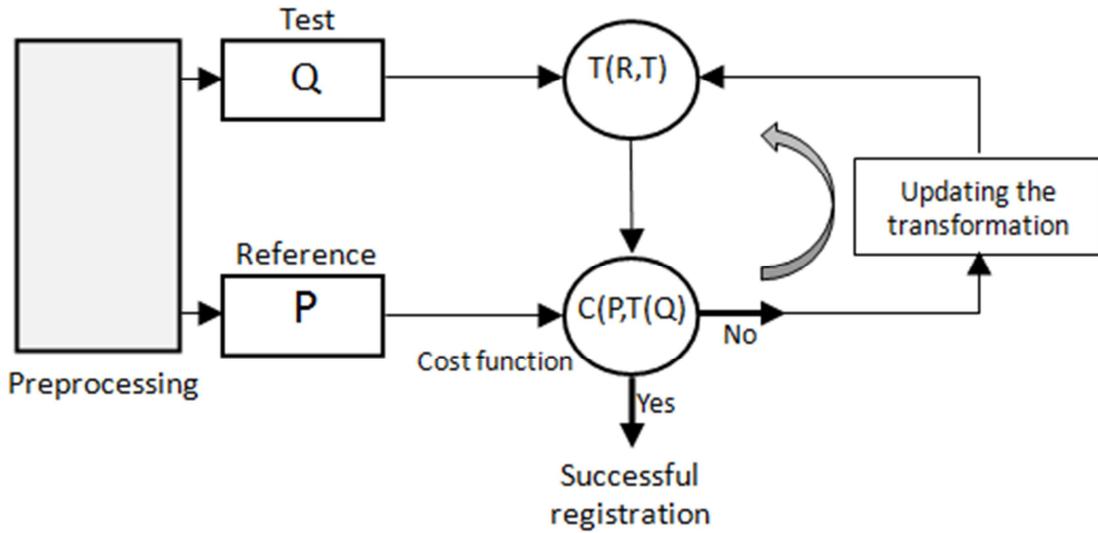


Figure 3. The principle of registration process.

Given two fingerprint images (Two maps of minutiae):

$$Q = \{q_i / i = 1 : N_q\}; \text{ The source image to register;}$$

$$P = \{p_j / j = 1 : N_p\}; \text{ The reference Image.}$$

We consider the rigid registration of a source image  $Q : \Omega_s \rightarrow \mathbb{R}$  to a target image  $P : \Omega_c \rightarrow \mathbb{R}$

The domains  $\Omega_s$  and  $\Omega_c$  of the two images  $Q$  and  $P$  are subsets of  $\mathbb{R}^2$ .

**3.2. ICP Algorithm Applied to the Fingerprints**

The proposed registration method is an iterative method ICP, which takes as input two point clouds of two-dimensional, and outputs an optimal estimate of the transformation between the two sets.

Therefore, the purpose of this algorithm is to find the parameters of the transformation (rotation and translation) which aligns a test fingerprint  $Q$  to another as reference  $P$ .

The cost function used is given by:

$$C = \min_{R, \vec{t}, j} \left( \sum_{i=1}^{N_q} \left\| (R \vec{q}_i + \vec{t}) - \vec{p}_j \right\|^2 \right) \begin{matrix} R, R^T = R^T, R = I_2 \\ \det(R) = 1 \\ j = 1 : N_p \end{matrix}$$

With  $R$  and  $\vec{t}$  are the parameters of the transformation (rotation and translation respectively).

The ICP algorithm reaches its speed and precision performance by iterating the following two steps:

1. First, we establish the correspondences linking the two sets of points:

$$C_k(i) = \arg \min_{j=1:N_p} \left( \left\| (R_{k-1} \vec{q}_i + \vec{t}_{k-1}) - \vec{p}_j \right\|^2 \right) \quad i = 1 : N_q$$

2. Then, we calculate the new rotation and translation minimizing the least square criterion:

$$(R_k, T_k) = \arg \min_{R, \vec{t}} \left( \sum_{i=1}^{N_q} \left\| R \vec{q}_i + \vec{t} - \vec{p}_{C_k(i)} \right\|^2 \right)$$

Note that at each iteration, ICP method uses the results obtained at the last iteration to yield a better result. That is until the result converges or that we have exceeded the maximum number of iterations allocated to the algorithm[9].

**3.2.1. Matching Step**

In first step of the algorithm, the matching is done by a simple algorithm to search neighborhood, which consist to pass on all the points  $N_p$  of  $P$  and to see if it's nearer or not that one of the already selected close neighbors, and if so, insert it. We then obtain a linear calculation time in the size of  $P$ . This method is called linear search or sequential search. However, this method suffers from a problem of slowness if the set  $P$  is too large, then it is extremely expensive to test  $N_p$  points in space. Fortunately, in our case of fingerprints, the number of matching minutiae points does not exceed a few

dozen.

This search algorithm will be used as reference to compare other algorithms that use more complex data storage structures like Delaunay triangulation Matlab [10][11], Kd-tree [12], etc...

- Delaunay triangulation invented by the Russian mathematician Boris Delaunay in 1934. This triangulation is applied for each of the two sets (source and target), then they are merged afterwards.
- The Kd-Tree introduced in 1975, the principle is to try to place each point of the source image in the tree, and up until one gets to the root of the tree while searching closest neighbors.

Still in the same step, looking for the nearest point is an important part if you want to get a method, which is robust as well as efficient. This is why the detection of the existence of aberrant points (outliers) is significant. The existence of outliers in point sets to match is very detrimental to our algorithm because when we seek the transformation between the two sets, we assume that we have the same cloud of points at different positions, assumption that's violated in presence of outliers.

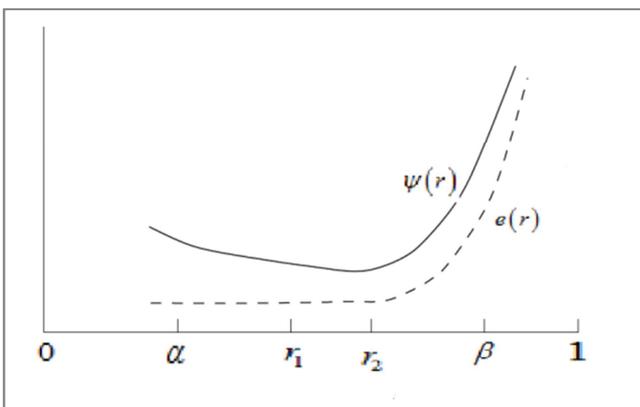
To solve the problem of outliers, we sort in ascending order all pairings by comparing the distances between paired points. Then we select the  $n_r = r * n_q$  first pairings with  $r$  is a parameter called overlay coefficient including in the interval  $[r_{\min}, r_{\max}]$ , and obtained by minimizing the function:

$$\psi(r) = \frac{e(r)}{r^{1+\lambda}}; \text{ (See figure 4)}$$

Such as

$$e(r) = \frac{1}{n_r} \sum_{i=1}^{n_r} (d_i^{k+1})^2, d_i^{k+1} = \|R_k \vec{q}_i + \vec{t}_k - \vec{p}_{c_{k+1}(i)}\|$$

$\lambda = 2$ , and typically ( $r_{\min} = 0.4$  and  $r_{\max} = 1$ )



(Golden Section Method) [13]

Figure 4. Golden Section Method.

These pairings will be those that we will use to estimate the transformation between the two clouds following the

approach of the ICP method without outliers. The other points are dismissed as outliers.

### 3.2.2. Transformation Update

In the second step of the algorithm, the used approach is that of the singular value decomposition (SVD) applied in 2D. The idea is this:

1. Calculate the center of mass  $p_c$  and  $q_c$  of the two point clouds  $\{p_i\}$  and  $\{q_i\}$ :

$$P_c = \frac{1}{n_r} * \sum_{i=1}^{n_r} p_i \quad \text{and} \quad Q_c = \frac{1}{n_r} * \sum_{i=1}^{n_r} q_i \quad (i = 1 \dots n_r)$$

2. Construct the covariance matrices:

$$P' = \{p_i - p_c\} = \{p'_i\} \quad \text{and} \quad Q' = \{q_i - q_c\} = \{q'_i\} \\ (i = 1 \dots n_r)$$

3. Calculate the matrix  $K \in \mathbb{R}^{2*2}$  such as  $K = P'.Q'^T$

$$(P', Q' \in \mathbb{R}^{2*n_r})$$

4. Write the matrix K as  $K = U.S.V^T$ , this decomposition is done by the SVD function predefined in Matlab by writing  $[U.S.V] = SVD(K)$  where U and V are orthogonal matrices of orders 2 and S a diagonal matrix of size  $2 * 2$  whose diagonal contains the singular values of K.
5. The parameters of the transformation are deduced by:

$$R = U.V^T \text{ (Rotation)}$$

$$T = p_c - R.q_c \text{ (Translation)}$$

Note that if  $\det(R) < 0$

$$R = V * B * U^T \quad \text{where} \quad B = \begin{pmatrix} 1 & 0 \\ 0 & \det(V.U^T) \end{pmatrix}$$

### 3.3. Evaluation of the Method on Real Data of Fingerprints

We tested our approach by taking an incorporated reference database of 20 fingerprints taken from different people and another database for test containing 10 fingerprints, which six are taken from people already on the database with different positions of those carried out beforehand, whereas the remaining four are removed from people not included in database. (Figure 5)

- Preprocessing stage:

For simplicity, we limit ourselves to such minutiae endings. After pretreatment and extraction of minutiae, we clearly observe the presence of outliers, this is where the parameter  $r$  is automatically updated with each iteration.

The figure 6 shows two samples of fingerprint images taken from the reference and test database respectively and

this, before and after the preprocessing stage.

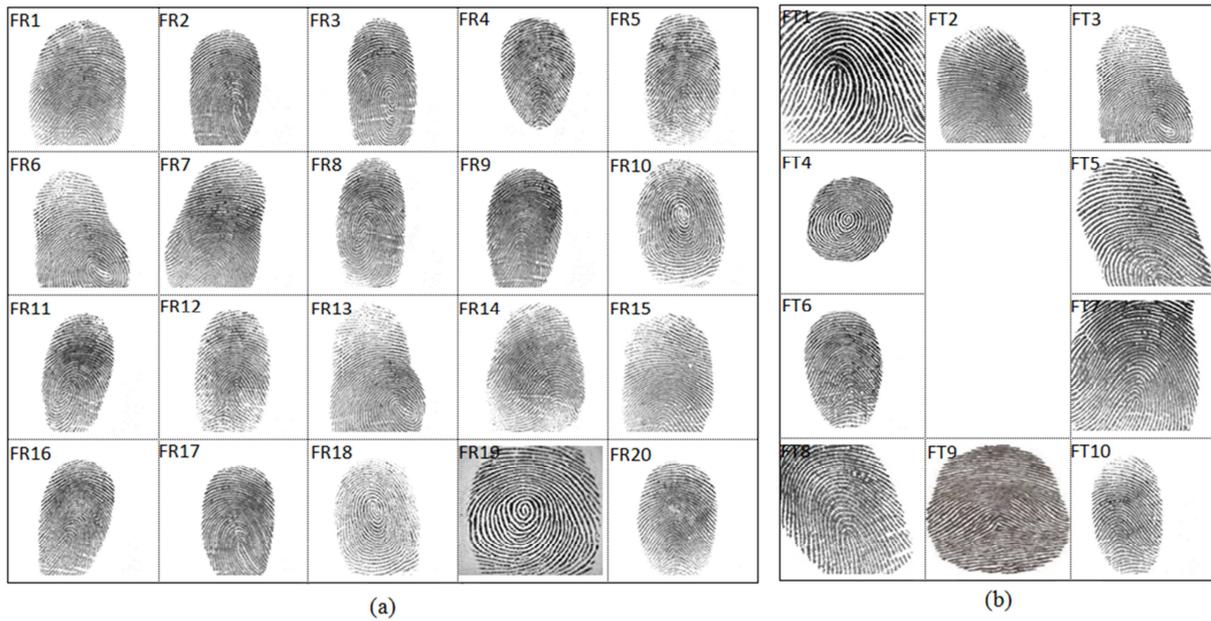


Figure 5. (a) Reference database; (b) Test database.

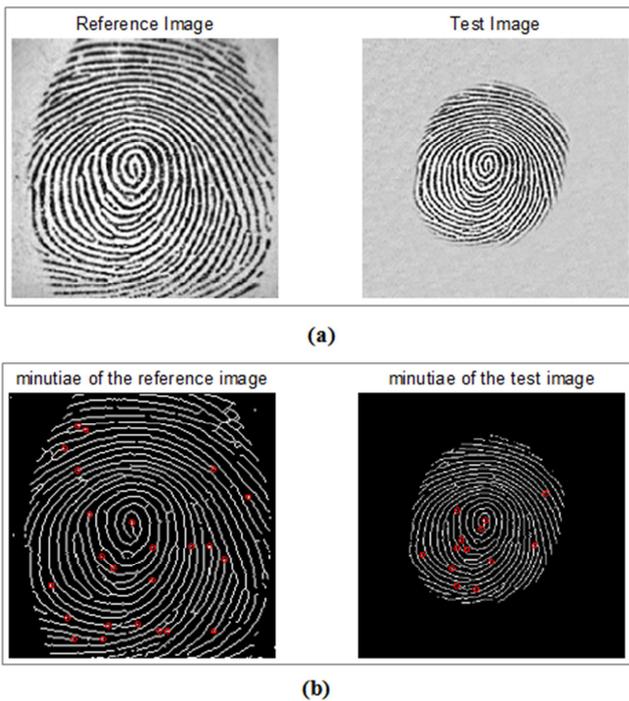


Fig. 6. Two samples of fingerprints taken from reference and test database respectively. (a) Before pretreatment; (b) After pretreatment.

- Alignment step of fingerprint images:

This stage includes the matching based on the naive method of searching the nearest neighbor cited in (3.2.1), then an intermediate step is to reject false comparison points (outliers) to reach finally the calculation of the new transformation (rotation and translation) minimizing the cost function. (Figures 7, 8, 9 and 10)

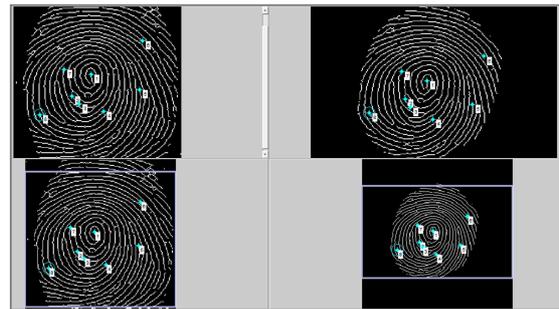


Fig. 7. Extraction of the two point clouds without outliers.

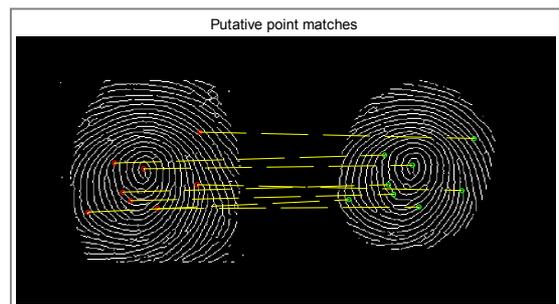


Fig. 8. Appariements between the two point sets of extracted minutiae.

## 4. Results and Discussions

The confrontation results of the test fingerprints ( $FT_i^1$  / $i=1, 2, \dots, 10$ ) to those of the reference ( $FR_j^2$  / $j=1, 2, \dots, 20$ ) by the studied method are provided in the following table (Table 1).

1  $FT_i$ : Test fingerprint number  $i$  where  $i=(1,2, \dots, 10)$ ;  
 2  $FR_j$ : Reference fingerprint number  $j$  where  $j=(1,2, \dots, 20)$ .

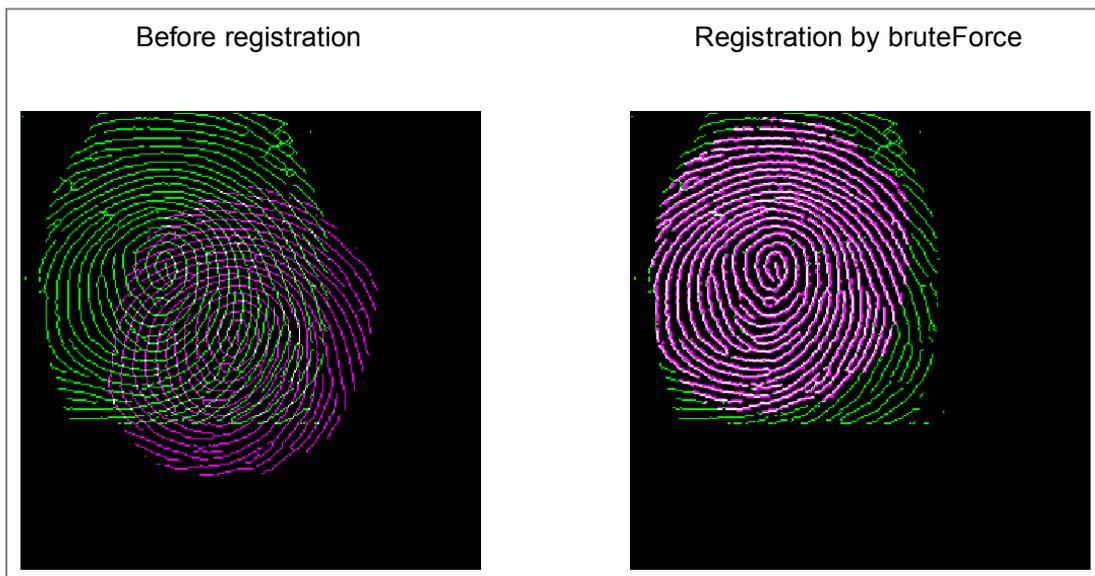


Fig. 9. Fingerprint images before and after registration by ICP.

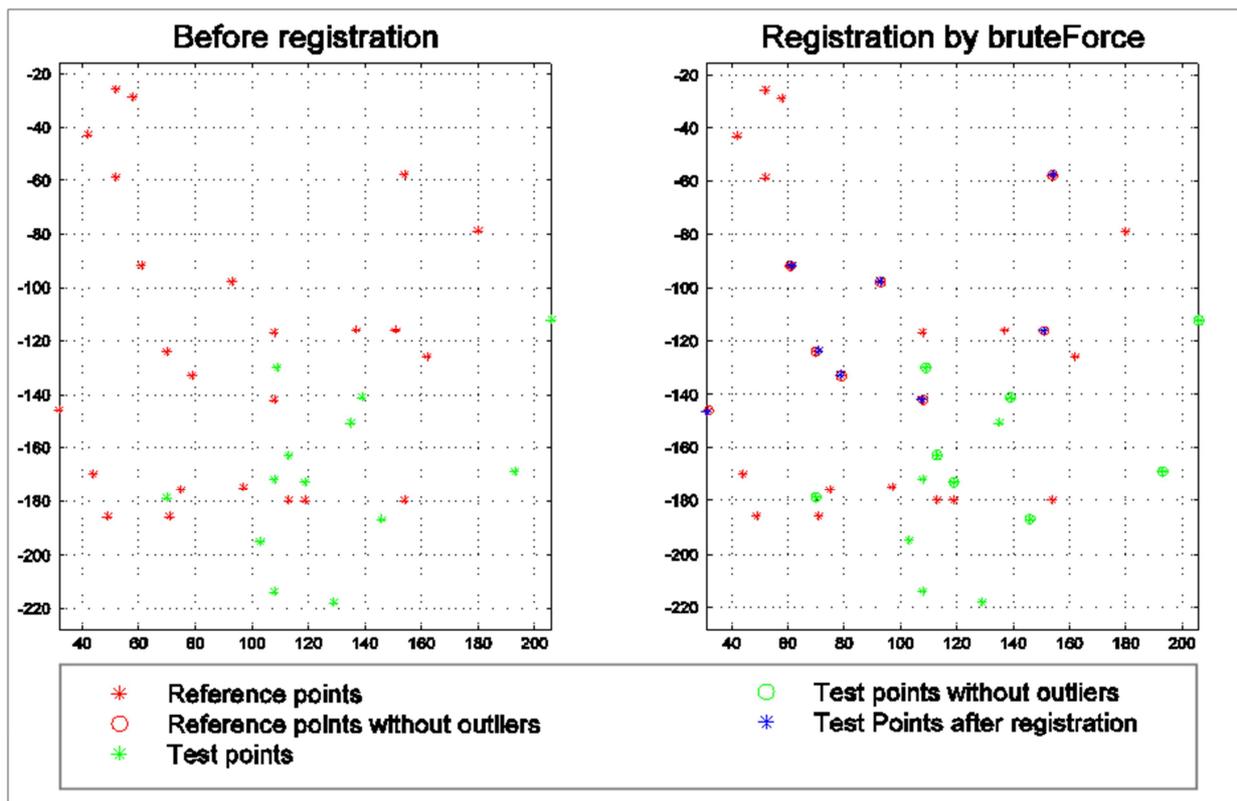


Fig. 10. Visualization of the two point clouds before and after registration by ICP.

Table 1. Least squares error between the set of minutiae in the test and reference database.

	FT1	FT2	FT3	FT4	FT5	FT6	FT7	FT8	FT9	FT10
FR1	1.841e2	2.241e2	0.705e2	3.510e2	3.001e3	2.110e2	2.143e2	1.726e3	7.520e1	7.021e2
FR2	2.245e2	7.217e1	1.235e2	0.271e3	4.231e-3	3.237e1	3.125e3	3.115e2	4.112e3	3.210e3
FR3	7.210e1	4.378e2	0.710e1	5.143e2	2.800e2	1.628e3	2.620e1	2.720e2	7.520e1	5.200e1
FR4	4.387e2	1.435e3	1.804e2	1.780e3	3.478e1	1.443e2	1.376e2	4.890e2	0.370e2	3.760e2
FR5	1.405e3	0.807e2	1.403e3	6.607e1	0.115e2	7.617e1	3.450e3	4.450e2	1.214e3	2.850e3

	FT1	FT2	FT3	FT4	FT5	FT6	FT7	FT8	FT9	FT10
FR6	0.867e2	1.023e2	6.035e-4	1.073e2	5.011e4	5.022e2	3.257e4	8.215e2	7.125e2	0.453e4
FR7	1.035e2	2.720e1	9.072e2	9.702e1	3.712e1	6.420e1	5.112e-4	9.100e1	6.034e3	7.500e2
FR8	2.721e1	0.860e3	0.770e3	0.896e3	6.823e0	1.031e3	1.750e3	6.750e3	3.712e2	3.230e2
FR9	0.864e3	1.075e3	4.800e1	0.089e3	2.700e2	1.912e1	3.719e2	5.300e-2	5.072e1	1.067e3
FR10	1.750e3	2.307e2	3.840e3	3.790e2	3.021e2	5.027e2	4.730e2	0.160e4	3.650e3	3.177e2
FR11	2.300e2	1.924e1	1.043e2	2.103e2	7.750e1	1.204e3	1.072e2	1.083e3	5.363e2	4.150e3
FR12	1.942e1	5.872e2	5.072e1	9.325e1	1.100e4	6.112e2	3.061e1	5.642e1	1.642e3	6.628e2
FR13	5.824e2	1.107e3	2.004e-2	6.300e2	8.114e2	5.117e1	7.194e2	3.414e2	9.100e2	3.110e3
FR14	1.177e3	2.450e2	1.487e3	1.622e1	2.810e2	1.092e2	1.037e4	2.657e3	1.847e3	4.720e2
FR15	2.450e2	1.380e-3	1.253e3	4.281e2	1.276e3	1.380e3	1.253e3	2.493e3	8.431e2	9.070e2
FR16	4.237e1	8.641e2	4.237e3	1.322e1	1.097e2	7.311e2	4.237e2	4.270e2	2.340e-2	5.017e2
FR17	8.654e2	9.045e1	5.301e1	4.623e1	3.654e-2	1.045e2	5.301e3	0.511e3	3.110e3	6.533e2
FR18	9.046e1	0.809e2	2.086e3	3.280e2	7.053e1	5.119e2	2.029e2	3.019e2	7.216e2	7.100e1
FR19	0.807e2	7.475e1	4.750e2	5.620e-4	2.381e2	7.041e2	4.712e2	4.312e1	5.327e3	5.970e2
FR20	7.450e1	4.273e1	8.310e1	6.211e2	4.610e2	1.323e3	4.623e2	0.610e3	1.972e1	1.410e3

These results demonstrate that people whose fingerprints FT1, FT6 and FT10 are not listed on the reference database because the value of the Least squares error is in the range of  $10^1$  to  $10^4$ .

As to persons whose fingerprints FT2, FT4, FT7, FT8 and FT9, they are identified in the reference database and correspond to people whose fingerprints are respectively FR15, FR19, FR7, FR9 and FR16. Indeed, the errors stored vary in the range of  $10^{-2}$  to  $10^{-4}$ .

Still based on the table, we note that the fingerprint FT3 corresponds well to the reference fingerprints FR6 and FR13 with respective errors  $6,035.10^{-4}$  and  $2,004.10^{-2}$ . This demonstrate that the fingerprints FR6 and FR13 are taken from the same person. Similarly for the test fingerprint FT5 that matches the person suffering the two prelevements FR2 and FR17 with respective errors in the range of  $10^{-2}$  to  $10^{-3}$ .

## 5. Conclusion

Thus, the work proposed in this paper is based on a purely geometric approach by implementing a system of identification or verification by rigid registration of fingerprints, with the main objectives processing speed and eliminating subjectivity based in particular on mathematical criteria to evaluate the results, this approach includes a chain of operations ranging from preprocessing fingerprints to rigid registration and based on the minutiae extracted. Indeed, we have chosen the minutiae type terminations as information to consider for guiding the registration. Then, after formalization of the general problem of registration, a registration method by ICP algorithm was proposed. At the rate of its sensitivity to noise and outliers, this algorithm has been improved by the introduction of a parameter, which acts on the selection of pairings. The latter are used to estimate the transformation between the two fingerprint images following the approach of the ICP method without outliers.

The results thus obtained in our study are compelling and indicate the effectiveness of the method in the field of identification of persons based on their fingerprints.

## References

- [1] Lorène, "La biométrie multimodale: stratégies de fusion de scores et mesures de dépendance virtuelle". Thèse de doctorat de Al Mutaz M. Abdalla, Safaai Dress, Nazar Zaki, "Detection of Masses in Digital Mammogram Using Second Order Statistics and Artificial Neural Network", International Journal of Computer Science & Information Technology (IJCSIT), Vol 3, No 3, pp. 176-186 June 2011.
- [2] Morizet, "Reconnaissance biométrique par fusion multimodale du visage et de l'iris", Thèse de doctorat Télécom 2009.
- [3] Doublet, Revenu, Olivier, "Reconnaissance biométrique sans contact de la main intégrant des informations de forme et de texture", France Telecom.2003.
- [4] Salil Prabhakar, Anil K. Jain, "Learning Fingerprint Minutiae Location and Type", International Conference on Pattern Recognition (ICPR), 2000.
- [5] Chaohong, "Advanced feature algorithms for automatic fingerprint recognition system", University of New Yorkatbuffalo.2007.
- [6] A. Chaari, S. Lelandais, M. B. Ahmed, "Face classification scheme simplifying identification in biometric databases", Transactions on Systems, Signals & Devices (TSSD), Shaker-Verlag, sous presse, 2009.
- [7] LIU L., JIANG T., YANG J., and al., Fingerprint Registration by Maximization of Mutual Information, IEEE Transactions on image processing,15(5), 1100-1110, 2006.
- [8] M. Boutahri, S. El Yamani, S. Zeriuoh, A. Bouzid and A. Roukhe, Fingerprint Identification by Artificial Neural Network, Journal of Physical Science and Application (David publishing), pp.381-384 Jun 2014.
- [9] D. Maltoni, D. Maio, A.K. Jain, S. Prabhakar Handbook of Fingerprint Recognition Springer, New York, 2003.
- [10] E. M.Gross , D. Wagner, KD trees and Delaunay-based linear interpolation for function learning: a comparison to neural networks with error backpropagation, pp.649 – 653 Nov 1996.

- [11] Barber, C. B., Dobkin, D. P., Huhdanpaa, H., The quickhull algorithm for convexhulls. *ACM Trans. Math. Software* 22 (4), 469–483, 1996.
- [12] Nuchter, A., Lingemann, K., Hertzberg, J., Cached k-d tree search for ICP algorithms. In: *Proc. Sixth Internat. Conf. on 3-D Digital Imaging and Modeling(3DIM)*, pp. 419–426, 2007.
- [13] D. Chetverikov, D. Stepanov, P. Krsek, Robust Euclidean alignment of 3D point sets: the trimmed iterative closest point algorithm, *Vol 23, Number 3*, pp. 299-309, March 2005.