



HAL
open science

Comparative Study of Minutiae Selection Algorithms for ISO Fingerprint Templates

B Vibert, Christophe Charrier, Jean-Marie Le Bars, Christophe Rosenberger

► **To cite this version:**

B Vibert, Christophe Charrier, Jean-Marie Le Bars, Christophe Rosenberger. Comparative Study of Minutiae Selection Algorithms for ISO Fingerprint Templates. IS&T/SPIE Electronic Imaging,, Feb 2015, San Francisco, United States. hal-01120639

HAL Id: hal-01120639

<https://hal.science/hal-01120639v1>

Submitted on 26 Feb 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Comparative Study of Minutiae Selection Algorithms for ISO Fingerprint Templates

B. Vibert, C. Charrier, J.-M. Lebars, C. Rosenberger

Normandie Univ, France;
UNICAEN, GREYC F-14032 Caen, France;
ENSICAEN, GREYC, F-14032 Caen, France;
CNRS, UMR 6072, F-14032 Caen, France;
(benoit.vibert,christophe.rosenberger)@ensicaen.fr,
(christophe.charrier,jean-marie.lebars)@unicaen.fr

ABSTRACT

We address the selection of fingerprint minutiae given a fingerprint ISO template. Minutiae selection plays a very important role when a secure element (i.e. a smart-card) is used. Because of the limited capability of computation and memory, the number of minutiae of a stored reference in the secure element is limited. We propose in this paper a comparative study of 6 minutiae selection methods including 2 methods from the literature and 1 like reference (No Selection). Experimental results on 3 fingerprint databases from the Fingerprint Verification Competition show their relative efficiency in terms of performance and computation time.

Keywords: Minutiae Selection, ISO fingerprint template

1. INTRODUCTION

Nowadays, electronic transactions are part of our daily life (e-commerce, smartphones, physical access control ...). In order to guarantee the security of authentication, biometrics are often used. Many real applications benefit from this technology such as for user access control or e-payment. Nevertheless, a biometric data is very sensitive and cannot be revoked in general (like a password). In order to ensure its security and privacy, a biometric data is usually stored in a Secure Element (SE). The Secure Element could be a SmartCard, with an (OCC) On-Card-Comparison algorithm inside. Two steps are necessary when using a biometric system: 1) the enrolment and 2) the verification as described in Figure 1. The OCC algorithm permits to compute a comparison score between a captured biometric template and the reference one. As common practice, the biometric template stored in the SE follows the ISO Compact Card standard¹ to ensure the interoperability between biometric systems. This template is composed of a set of minutiae represented by 3 octets and 4 values $(x_i, y_i, T_i, \theta_i)$, $i = 1 : N_j$ where the coordinates (x_i, y_i) correspond to the location of the minutiae in the image, T_i corresponds to the minutiae type (bifurcation, ridge ending ...), θ_i to the minutiae orientation (related to the ridge) and N_j the number of minutiae for the sample j of the user (see Figure 2).

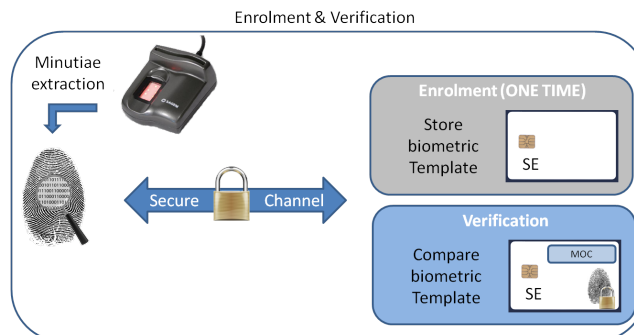


Figure 1: Enrolment and Verification step

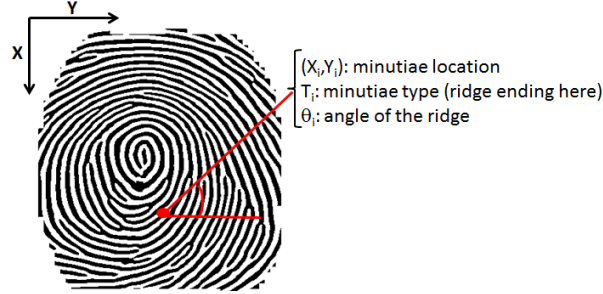


Figure 2: Description of a fingerprint minutiae

A SE has hardware and software constraints as for example the size of memory, the number of data we can send with an APDU command² (ISO 7816 standard for the communication with a SE). These limitations have an impact on the embedded algorithm and the size of the fingerprint template. Generally, in an operational application, a fingerprint template is limited to 48 minutiae when stored in a SE. When the sensor extracts more minutiae than the OCC is able to process, we have to reduce the size of the template by selecting the most appropriate minutiae. Considering the state-of-the-art, only few methods has been proposed. The first one is recommended by the ISO consortium.¹ This method is based on the peeling of minutiae and we refer to it as ISOTruncation in this paper. The second is presented by the NIST:³ this method is based on the CORE distance with all minutiae, we keep only the nearest minutiae to the CORE point, we allude to it as NISTBarycenter. yet, it is tedious to determine if a method is better than another one.

For this study, we are placed in the worst case in practice, when the image is not available. We have no information about the chosen minutiae extraction and the On-Card-Comparison algorithms. We just know the format of the template (ISO Compact Card). In our setting, we have as input the template and we send it to the OCC. If the size is too large, the OCC send us a response with the max length that is authorized for the comparison. To reduce the template, we used the Minutiae selection block as depicted in Figure 3; the output contains the reduced template which will be send to the OCC with the expected size.

The aim of this paper is therefore to determine how the way to select the minutiae without any knowledge on minutiae extraction and On-Card-Comparison algorithms impacts the performance of the biometric authentication. For this purpose, we study several selection methods which implements the Minutiae selection block.

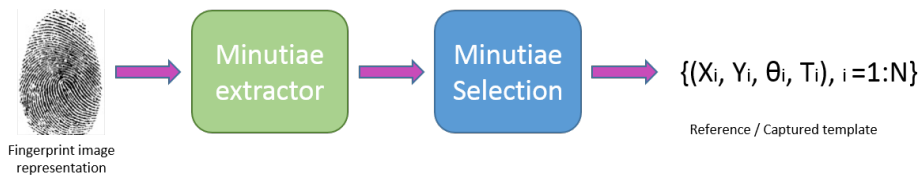


Figure 3: Minutiae selection from the fingerprint template

2. MINUTIAE SELECTION METHODS

In this section, we present the studied selection methods. Some come from the state of the art and we also proposed many of them. In total, we consider in this study 6 selection methods. The methods are placed in three categories, original template, reordering minutiae by truncation and minutiae clustering.

2.1 No selection "Original template"

The first method is to consider all minutiae in the template, we are placed in the ideal case. The performance associated to the original template is used as reference for the experimental results. We could expect as for example a decrease of the performance of other selection methods face to this method.

2.2 Truncation-based method

2.2.1 NIST Barycenter

This method is based on a pruning mechanism. This approach is simple and fast (few milliseconds). It has been shown that minutiae located near the core of the fingerprint are the most useful for the matching process. Given a fingerprint template, the core location is in general unknown but the centroid of minutiae is a good estimate in general. This minutiae selection approach tends to keep minutiae near the centroid for this reason. We have four steps in this method :

1. Compute the centroid of the fingerprint template (containing N_j minutiae);

$$Centroid = (X_c, Y_c) = \frac{1}{N_j} \left(\sum_{i=1}^{N_j} X_i, \sum_{i=1}^{N_j} Y_i \right) \quad (1)$$

2. Compute the Euclidean distance of each minutiae to the centroid;

$$r_i = \sqrt{(X_i - X_c)^2 + (Y_i - Y_c)^2}, \quad i = 1 : N_j \quad (2)$$

3. Sort in ascending order minutiae according to the distance r_i , $i = 1 : N_j$;
4. Select the first N_{max} minutiae.

2.2.2 Evolutive barycenter

This method is based on selection combined with the NIST barycenter method.

- Compute the centroid of the fingerprint template (see equation 1);
- Compute the distance of each minutiae to the centroid (see equation 2);
- Sort in ascending order minutiae according to the distance r_i , $i = 1 : N_j$;
- Eliminate the minutiae having the largest distance.
- Repeat the 4 steps until obtain the good number of minutiae.

2.2.3 Median Y

This method used nearly the same scheme than barycenter but we only used the Y information. We have four steps in this method :

- Compute the Y median of the template;

$$Median = (Y_m) = \frac{1}{N_j} \left(\sum_{i=1}^{N_j} Y_i \right) \quad (3)$$

- Compute the distance of each minutiae to the Y median;

$$r_i = \sqrt{(Y_i - Y_m)^2}, \quad i = 1 : N_j \quad (4)$$

- Sort in ascending order minutiae according to the distance r_i , $i = 1 : N_j$;
- Select the first N_{max} minutiae.

2.2.4 ISO Truncation

This method is based on a simple truncation *i.e.*, we only keep the first N_{max} minutiae from the initial template. The reason why this simple approach could be effective is related to the generation of the fingerprint template. For many commercial biometric systems, a fingerprint template is generated with a specific method. It can be generated considering minutiae with the ascending locations Y for example. In the case where multiple captures have been made, high quality minutiae (always present in the different captures for example) can be placed at the beginning of the template. Selecting the N_{max} first minutiae could be in this case very efficient and simple.

2.2.5 Truncation Random Permutation

This method is based on a Random permutation of the initial template and we only keep the N_{max} first minutiae. With this method we want to have a reduced template with minutiae in different places in the initial template. We want to determine if we need minutiae in different part of the template or not.

2.3 Minutiae clustering : K-means quantization

We used the Fuzzy C-Means (FCM).⁴ This method takes as parameters the minutiae template and the number of cluster (number of minutiae expected). We have seven steps in this method :

- Take a random minutiae for centroid of the first cluster ($C1$)
- Take another point with a good distance with $C1$ and become the centroid of the second cluster ($C2$)
- Calculate the distance of each minutiae with $C1$ and $C2$
- Class in each cluster the minutia with the minimal distance with the centroid.
- Take a new minutiae with the greatest distance with the other cluster
- Repeat the 5 steps until obtain the good number of cluster.
- For each centroid cluster take the minutiae with the minimal distance

3. EXPERIMENTAL SETUP

For the experiments, we need to define multiple aspects such as: the biometric databases, the minutiae extractor, the matching algorithm and the performance evaluation metrics. We detail all of them in the following.

3.1 Biometric databases

In this study, we used five fingerprint databases composed of 800 images from 100 individuals with 8 samples from each user:

- FVC2002 benchmark database DB2:⁵ the image resolution is 296×560 pixels with an optical sensor "FX2000" by Biometrika. This database has been used for the competition (Fingerprint Verification competition) in 2002;
- FVC2004 benchmark database DB1:⁶ the image resolution is 640×480 pixels with an optical sensor "V300" by CrossMatch;
- FVC2004 benchmark database DB2:⁷ the image resolution is 328×364 pixels with an optical sensor "U.are.U 4000" by Digital Persona;

Figure 4 presents one fingerprint from each database. It shows the diversity of fingerprints considered in this study.



Figure 4: One fingerprint example from each used database

3.2 Minutiae extractor and matching algorithm

Minutiae templates used in the experiments have been extracted by using the NBIS tool, MINDTCT⁸ from the NIST. In order to realize the matching of fingerprint templates, we used a very well known minutiae matching algorithm proposed in 1997 by Jain et al⁹ and the NIST named "Bozorth3". This method consists of an alignment stage (translation and rotation estimation between the two templates to compare) and a matching stage after transformation. We have used a second matching algorithm to compare and validate the best method of reduction. This algorithm is proposed by a commercial company, we have no information about the algorithm workflow.

3.3 Performance evaluation

In order to assess the performance of a biometric system, we use the Detection Error Tradeoff curve (DET),¹⁰ This curve plots the False Match Rate (FMR)¹⁰ (i.e., accepted impostor attempts) on the x-axis against the corresponding False Non Match Rate (FNMR)¹⁰ (i.e., rejected genuine attempts) on the y-axis plotted parametrically as a function of the decision threshold. The associated measure is called AUC (Area Under the Curve)¹⁰ and is often considered as global performance criterion. We use this value to quantify the efficiency of a minutiae selection method. The associated Confidence Interval (CI) will be provided. We also use EER (Error Equal Rate)¹⁰ value to have one point of work of the system.

4. EXPERIMENTAL RESULTS

In the following, we present the experimental results of the study.

4.1 Localization of minutiae in the reduced template

We show on Figure 5 the result of minutiae selection using the six methods. Selected minutiae are represented by a red star and others with a blue circle. We can see that the structure of the fingerprint template uses an Y ascending sorting of the minutiae locations. The CORE point is the focus of the innermost recurring ridge in a fingerprint.¹¹ With the barycentre and iterative barycentre approach, we select minutiae near the estimated CORE. With the K-means method, we have minutiae in different part of the template, we have nearly the same reduced template than the barycentre and iterative barycentre ones. With the vertical selection method, we loose minutiae in top and bottom of the template. With the truncation method, we loose the right part of the template. With the random truncation method, we obtain a similar shape of the fingerprint than the K-means one. We have this kind of result because we have a small size difference between the initial template and the reduce one.

4.2 Time for generating the template

In this section, we present the computation time to generate a reduced template with each method. Table 1 is an average time for each method on all databases. It permit to observe, for example, than quantization is around 3000 times more slower than truncation method. This indicator help us to choose the best fingerprint reduction method, because time is one of criteria very important on a commercial system.

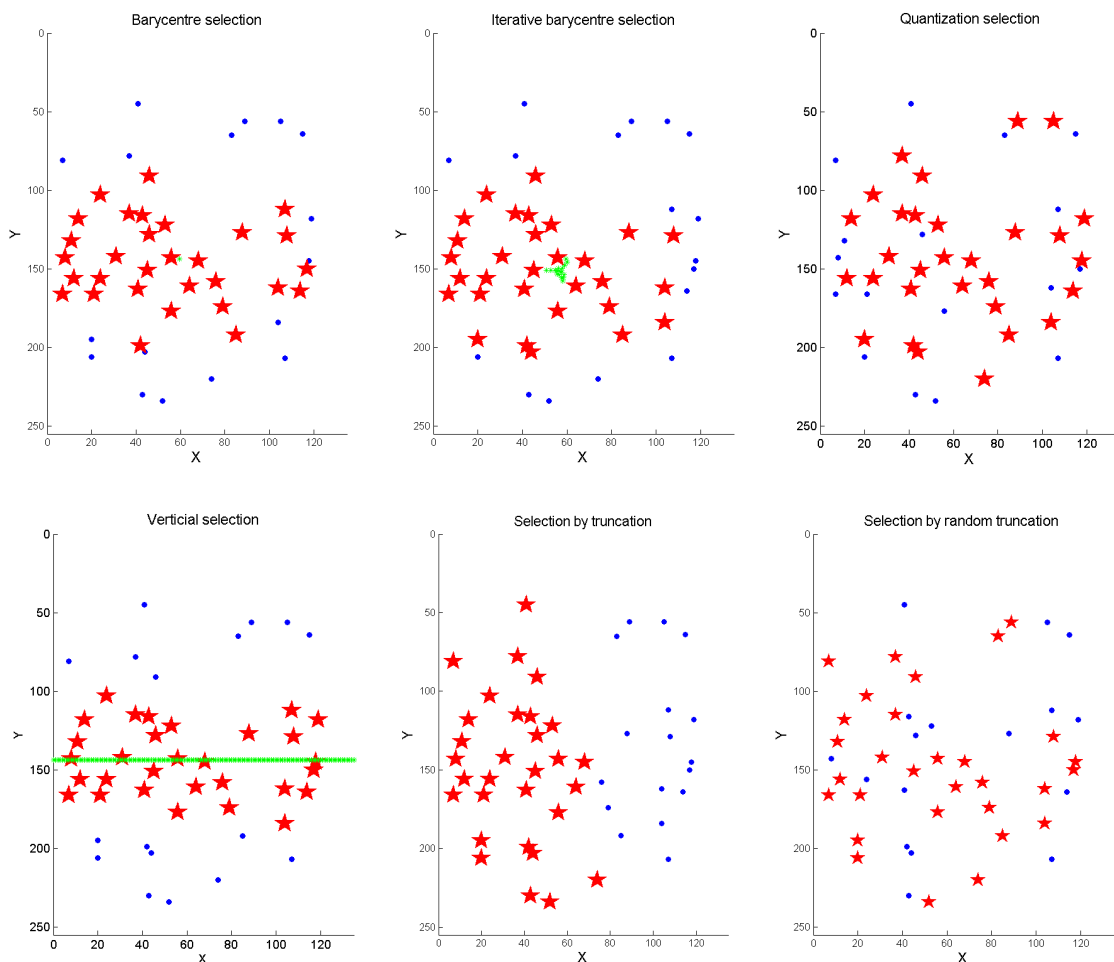


Figure 5: Example of minutiae selection on a fingerprint sample: stars represent selected minutiae by the different methods.

N_{max}	30	34	38	42	46	50
Truncation	7.9 ms	7.6 ms	7.4 ms	7.2 ms	7 ms	6.8 ms
Random truncation	20.7 ms	19.9 ms	18.9 ms	17.4 ms	16.5 ms	14.6 ms
Barycentre	476 ms	409 ms	376 ms	364 ms	301 ms	244 ms
Iterative Barycentre	5387 ms	4747 ms	3988 ms	3435 ms	2789 ms	2206 ms
Quantization	24349 ms	24047 ms	22783 ms	20569 ms	18298 ms	15473 ms
Vertical	171 ms	164 ms	193 ms	186 ms	189 ms	196 ms

Table 1: Average Time for all the minutiae size and all the methods

4.3 Performance evaluation

For all databases, we compute the AUC, CI (Confidence Interval) values for each selection method with N_{max} from 30 to 50 by step of 4. The table 2 presents the value of AUC for each database for the original template, it serves as reference for the comparison on our study.

<i>MatchingAlgorithm</i>	Bozorth	ID3
FVC2002DB2	14%	0.04%
FVC2004DB1	11.1%	3.77%
FVC2004DB2	11.1%	3.68%

Table 2: AUC values for the three databases with Bozorth and ID3

4.3.1 Bozorth matching algorithm

Tables 3, 4 and 5 detail the AUC value for each minutiae selection method for the FVC2002DB2, FVC2004DB1 and FVC2004DB2 database and for different numbers of selected minutiae N_{max} . We can notice that the quantization selection method provides most of time the best result. For the FVC2002DB2 database, most of selection methods permit to obtain a better performance compared to the initial template (Table 2). On the contrary, for the FVC2004DB1 and FVC2004DB2 database, the initial template often provides the best result.

N_{max}	30	34	38	42	46	50
Truncation	12.2%	12.2%	10.6%	9.84%	10.3%	11.3%
Random truncation	10.0%	10.1%	8.22%	8.06%	8.67%	9.55%
Barycentre	9.1%	9.87%	10.1%	10.3%	11.1%	12.0%
Iterative Barycentre	12.8%	12.5%	12.6%	11.4%	12.0%	12.5%
Quantization	7.67%	9.44%	7.86%	7.14%	7.63%	10.07%
Vertical	11.1%	11.3%	11.2%	10.9%	11%	11.4%

Table 3: AUC values for different values of N_{max} minutiae on FVC2002DB2 with Bozorth

N_{max}	30	34	38	42	46	50
Truncation	20.5%	18.8%	16.5%	15.3%	13.8%	13.1%
Random truncation	18%	16.6%	15.5%	13.2%	12.6%	12%
Barycentre	20%	17.3%	15.5%	13.4%	12.4%	11.8%
Iterative Barycentre	20.8%	19.4%	16.5%	14.5%	12.9%	12.2%
Quantization	15.7%	15.2%	14.2%	12.9%	12.2%	11.4%
Vertical	21.3%	19.1%	16.7%	14.4%	13%	11.9%

Table 4: AUC values for different values of N_{max} minutiae on FVC2004DB1 with Bozorth

N_{max}	30	34	38	42	46	50
Truncation	20.6%	18.7%	16.7%	15.4%	13.8%	13%
Random truncation	17.5%	16.5%	15.2%	13.6%	12.5%	11.8%
Barycentre	19.9%	17.3%	15.4%	13.4%	12.2%	11.7%
Iterative Barycentre	20.8%	19.3%	16.6%	14.7%	12.9%	12.1%
Quantization	16.7%	15%	14.4%	13.1%	12.1%	11.3%
Vertical	21.4%	18.9%	16.8%	14.5%	13%	11.8%

Table 5: AUC values for different values of N_{max} minutiae on FVC2004DB2 with Bozorth

Figure 6a presents the AUC values for the FVC2002DB2 database. In this case, some selection methods such as the truncation are able to improve the performance face to the initial template. The quantization (k-means) selection method remains the best for all the number of minutiae. Figure 6c presents the same result for the FVC2004DB1 database. In this case, the best performance is obtained using the initial template. That means that the selection will decrease the performance. The quantization, barycenter and random permutation

truncation selection methods permit to obtain the best results. It is the same way for FVC2004DB2 database Figure 6e, all the selection methods provide a higher AUC than the initial template, but we observe the method with the lower AUC is K-Means one.

4.3.2 SDK matching algorithm

Table 6, table 7 and table 8 details the AUC value for each minutiae selection method for the FVC2002DB2, FVC2004DB1 and FVC2004DB2 databases with SDK matching algorithm. We can notice that the quantization selection method provide most of time the best result. For the FVC2004DB1 and FVC2004DB2 database, barycenter and iterative barycenter have also a good performance.

N_{max}	30	34	38	42	46	50
Truncation	2.54%	1.59%	1.36%	0.95%	0.83%	0.59%
Random truncation	3.64%	1.46%	0.77%	0.85%	0.34%	0.15%
Barycentre	1.35%	0.84%	0.52%	0.31%	0.29%	0.28%
Iterative Barycentre	3.99%	2.53%	1.72%	1.01%	0.55%	0.37%
Quantization	1.04%	0.61%	0.32%	0.17%	0.04%	0.08%
Vertical	1.72%	1.08%	0.87%	0.57%	0.46%	0.44%

Table 6: AUC values for different values of N_{max} minutiae on FVC2002DB2 with SDK

N_{max}	30	34	38	42	46	50
Truncation	12.1%	9.47%	8.18%	6.61%	5.95%	5.12%
Random truncation	13.5%	11.7%	8.64%	6.22%	5.35%	5.06%
Barycentre	7.62%	7.05%	5.43%	4.85%	4.34%	4.09%
Iterative Barycentre	8.96%	7.28%	6%	5.16%	4.19%	4.19%
Quantization	7.21%	6.76%	5.78%	5.12%	4.26%	4.04%
Vertical	9.27%	7.76%	6.48%	5.46%	4.55%	4.29%

Table 7: AUC values for different values of N_{max} minutiae on FVC2004DB1 with SDK

N_{max}	30	34	38	42	46	50
Truncation	8.44%	7.21%	5.92%	5.28%	4.47%	4.3%
Random truncation	13%	11%	8.49%	6.51%	5.59%	4.78%
Barycentre	7.61%	7.17%	5.42%	4.85%	4.31%	4.11%
Iterative Barycentre	8.86%	7.45%	5.96%	5.13%	4.23%	4.17%
Quantization	9.14%	6.74%	5.69%	4.59%	4.51%	3.98%
Vertical	9.25%	7.89%	6.39%	5.38%	4.58%	4.23%

Table 8: AUC values for different values of N_{max} minutiae on FVC2004DB2 with SDK

For all databases, the best performance is obtained using the initial template. That means that the selection will decrease the performance. The figure 6b, presents the AUC values for the FVC2002DB2 database. In this case, we obtain a very good performance with low AUC value but we have worst result than the initial template. The quantization selection method remains the best all of time on this databases. Figure 6d presents the same result for the FVC2004DB1 database. The quantization, barycenter and iterative barycenter selection methods permit to obtain the best results. It is the same way for FVC2004DB2 database Figure 6f, all the selection methods provide a higher AUC than the initial template, but we observe the method with the lower AUC is K-Means (quantization) one with barycenter and iterative barycenter for some numbers of minutiae. We can conclude than quantization is the best method for all the database. The reason is probably than

quantization is a clustering method and we keep the shape of the original template. The other methods change the shape and we loose a lot of important minutiae needed for matching algorithm.

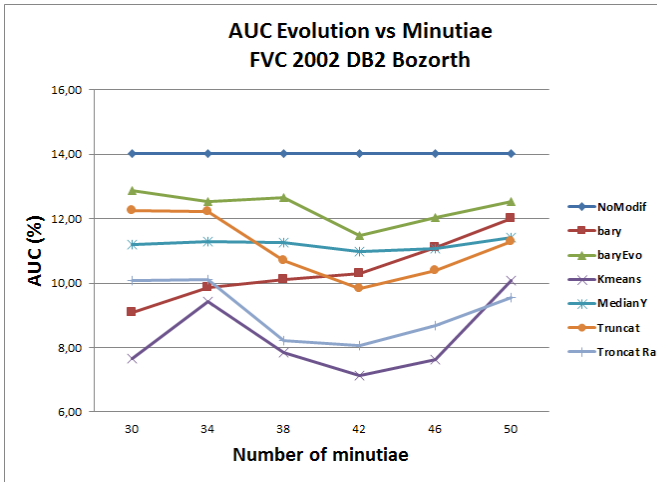
5. CONCLUSION

In this paper, we have presented a comparative study of fingerprint minutiae selection algorithms for ISO Compact Card fingerprint binary template. For this study, 2 minutiae selection method of the state-of-the-art are used, and we proposed 4 methods. We also have used 3 databases used from the FVC competition and 2 matching algorithms, one for the state-of-the-art and the other is a commercial sdk. We have observed, than using a minutiae selection method reduce the performance of the matching algorithm. This is a normal result because we reduce the number of minutiae to compare it to the reference one. During this study, we have observed for all the database and matching algorithm, than the best method in general is the quantization process, consisting in using a K-Means. We can explain this result by the fact than quantization is a clustering method and we keep the shape of the minutiae template.

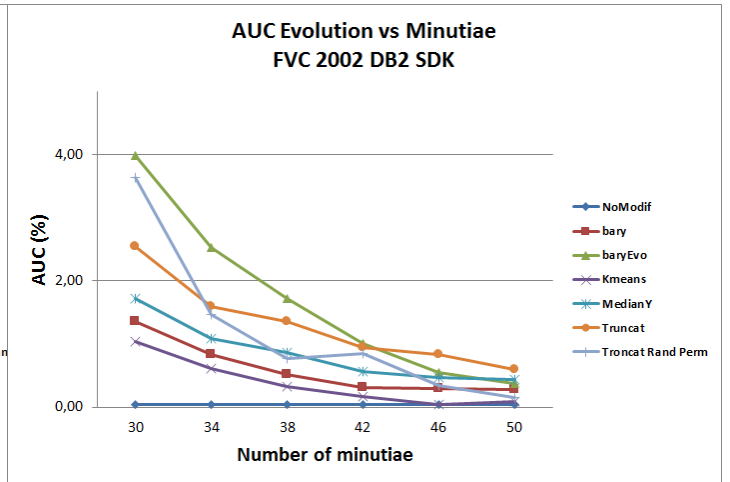
In perspective, we intend to combine different strategies in order to combine many minutiae selection methods in order to obtain better results. We also want to confirm with other clustering method than this kind of process has minimal impact on performance.

REFERENCES

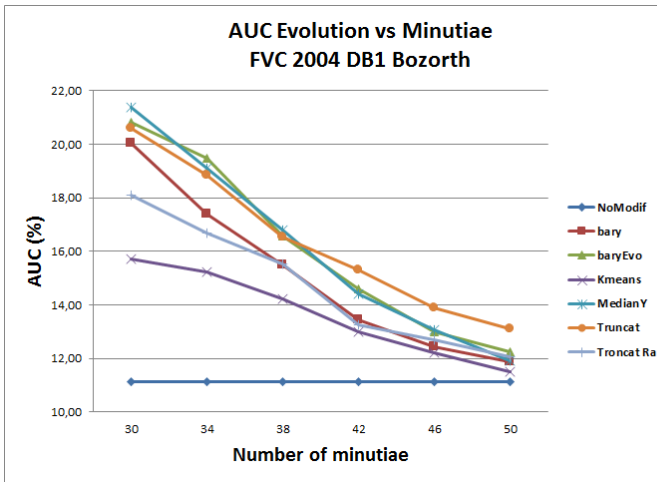
1. "Iso/iec 19795-2. information technology - biometric data interchange format - part 2 : Finger minutiae data, 2004.."
2. ISO/IEC, <http://www.iso.org>, *ISO/IEC 7816-1 to 15: Identification cards - Integrated circuit(s) cards with contacts(Parts 1 to 15)*.
3. Grother, P. and Salamon, W., "Interoperability of the iso/iec 19794-2 compact card and 10 iso/iec 7816-11 match-on-card specifications 11," (2007).
4. Pal, N. R. and Bezdek, J. C., "On cluster validity for the fuzzy c-means model," *Fuzzy Systems, IEEE Transactions on* **3**(3), 370–379 (1995).
5. "Fvc2002db2." <http://bias.csr.unibo.it/fvc2002/download.asp>.
6. "Fvc2004db1." <http://bias.csr.unibo.it/fvc2004/databases.asp>.
7. Maltoni, D., Maio, D., Jain, A. K., and Prabhakar, S., [*Handbook of fingerprint recognition*], springer (2009).
8. Watson, C. I., Garris, M. D., Tabassi, E., Wilson, C. L., McCabe, R. M., Janet, S., and Ko, K., "Users guide to nist biometric image software (nbis)," tech. rep., NIST (2007).
9. Jain, A. K., Hong, L., Pankanti, S., and Bolle, R., "An identity-authentication system using fingerprints," *Proceedings of the IEEE* **9**, 1365–1388 (1997).
10. "ISO/IEC 2382-37. information technology - vocabulary - part 37 : Biometrics, 2012.."
11. NIST, "Nist glossary." <http://www.nist.gov/forensics/EFSTrainingTool/ResourcesTab/Glossary.html>.



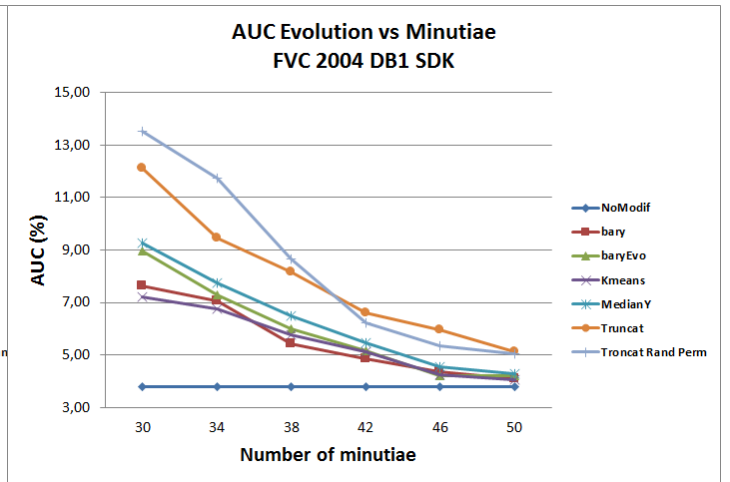
(a) AUC evolution on FVC2002DB2 with Bozorth



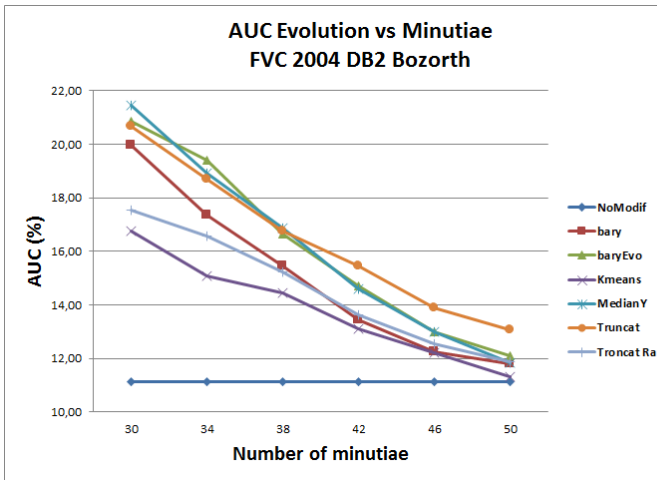
(b) AUC evolution on FVC2002DB2 with SDK



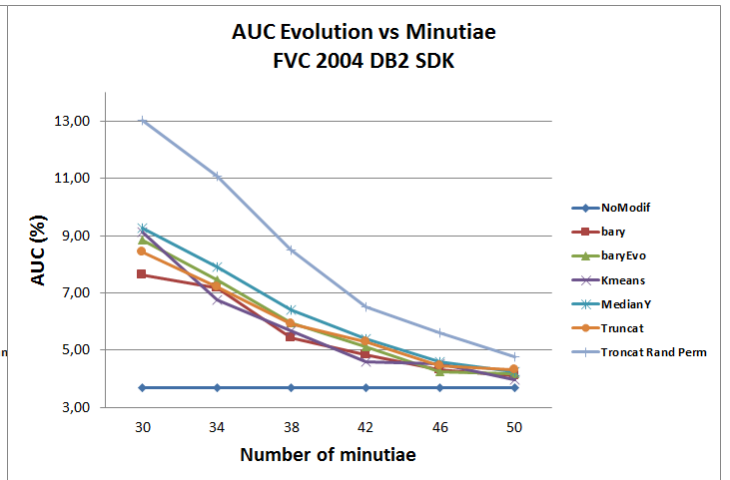
(c) AUC evolution on FVC2004DB1 with Bozorth



(d) AUC evolution on FVC2004DB1 with SDK



(e) AUC evolution on FVC2004DB2 with Bozorth



(f) AUC evolution on FVC2004DB2 with SDK

Figure 6: Evolution of the AUC value face to minutiae selection