Hill-Climbing and Brute-Force Attacks on Biometric Systems: A Case Study in Match-on-Card Fingerprint Verification

M. Martinez-Diaz, J. Fierrez-Aguilar, F. Alonso-Fernandez, J. Ortega-Garcia, J.A. Siguenza
ATVS/Biometrics Research Lab, Escuela Politecnica Superior - Universidad Autonoma de Madrid
C/ Francisco Tomas y Valiente, 11 - Campus de Cantoblanco - 28049 Madrid, Spain
{julian.fierrez, fernando.alonso, javier.ortega}@uam.es

Abstract

In this paper, we study the robustness of state-of-the-art automatic fingerprint verification systems against hill-climbing and brute-force attacks. We compare the performance of this type of attacks against two different minutiae-based systems, the NIST Fingerprint Image Software 2 (NFIS2) reference system and a Match-on-Card based system. In order to study their success rate, the attacks are analyzed and modified in each scenario. We focus on the influence of initial conditions in hill-climbing attacks, like the number of minutiae in the synthetically generated templates or the performance of each type of modification in the template. We demonstrate how slight modifications in the hill-climbing algorithm lead to very different success rates.

1. Introduction

Biometrics is becoming an important issue in our society [1]. The heightened interest in biometrics-based automated personal identification has resulted in the development of several commercial biometric recognition systems. Fingerprints are one of the most commonly used biometrics due to their reduced size and acceptability [2]. Despite the development of fingerprint recognition techniques, there are many security concerns [3] which still make it a topic for discussion.

One of the hot topics within biometrics are Match-on-Devices, and in particular Match-on-Card based systems for fingerprint recognition. Smart-cards allow to encrypt and protect stored information and to execute matching algorithms [4]. Thus, the user's fingerprint template and the matching algorithm can be stored in a smart-card without compromising its security. Corroborating this increasing interest in Match-on-Card systems, in the Fingerprint Verification Competition (FVC) 2004 [5], a special evaluation track for matching systems with reduced time and memory restrictions was introduced. Furthermore, in this year's competition, FVC 2006 [6], a new category including Match-on-Card systems has been proposed.

A fingerprint recognition system is vulnerable to attacks which may decrease its security level. Ratha *et al.* [7] have studied and classified these attacks in 8 different types. Attacks from type 1 are aimed at the sensor and can be carried out using fake fingerprints. Types 3, 5 and 6 may be performed as Trojan Horse attacks, bypassing the feature extractor, the matcher, and the system database respectively. Types 2, 4, 7 and 8 attack communication channels and can either try to intercept information or insert it into the channel. Possible attack points in a general biometric recognition system are depicted in Fig. 1.

In this study, we focus on attacks known as *hill-climbing* attacks [8]. Hill-climbing attacks consist of an application that sends synthetically generated minutiae templates to the matcher and, according to the match score, randomly modifies the templates until the decision threshold is exceeded. We implement hill-climbing attacks against both the NFIS2 reference system [9] and a Match-on-Card (MoC) system, and then study some factors involved in the success rate of the attack. A direct comparison is also made between our hill-climbing attacks and brute-force attacks.

Using smart-card embedded matching systems for fingerprint recognition has already been studied [4, 10] but, to the best of our knowledge, no attacks aimed directly to the smart-card matcher have been reported in the literature.

The rest of the paper is organized as follows. Hill-climbing and brute force attacks are explained in Sect. 2, the fingerprint recognition systems under attack are presented in Sect. 3, experiments are described in Sect. 4, and conclusions are finally drawn in Sect. 5.

2. Hill-climbing and Brute-Force Attacks

Hill-climbing attacks against automated fingerprint recognition systems have been studied by Uludag and Jain [8] and Soutar [12]. A hill-climbing attack may be performed by an application that sends random templates to the system, which are perturbed iteratively. The application reads the output match score and continues with the perturbed template only when the matching score increases until the decision threshold is exceeded.

A hill-climbing attack may be of type 2 or 4, depending on the point of attack. Soutar proposed in [12] a type

Published as: M. Martinez-Diaz, J. Fierrez-Aguilar, F. Alonso-Fernandez, J. Ortega-Garcia and J. A. Siguenza, "Hill-climbing and brute-force attacks on biometric systems: A case study in Match-on-Card fingerprint verification", in Proc. IEEE Intl. Carnahan Conf. on Security Technology, ICCST, pp. 151-159, Lexington, USA, October 2006.

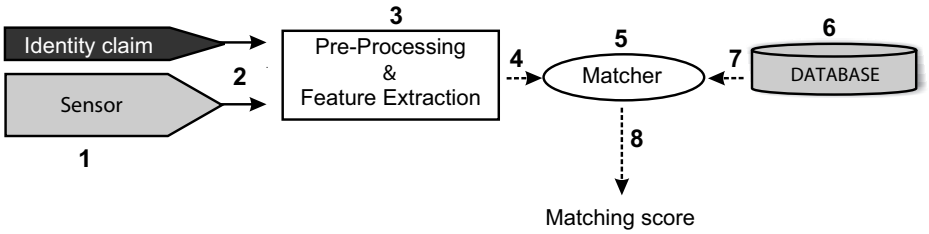


Figure 1. Architecture and dataflow paths of an automated biometric verification system. Possible attack points are numbered from 1 to 8.

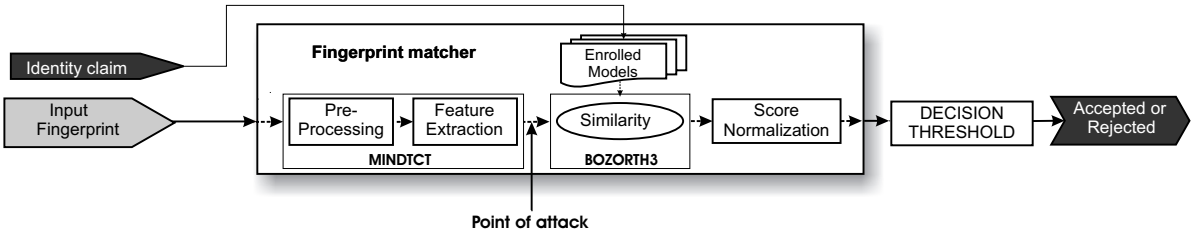


Figure 2. Architecture, dataflow paths and point of attack of the NFIS2 system.

2 attack with a face recognition system. The input image is conveniently modified until a desired matching score is attained. In [8], a type 4 attack against a minutiae-based fingerprint recognition system is described. Uludag and Jain [8] propose an attack based on synthetic random minutiae templates which are modified, one minutia at a time, until the decision threshold is exceeded.

In this paper, we study a hill-climbing attack based on the one presented by Uludag and Jain [8]. The template format of the matching systems must be known by the attacker as well as the input image size. Note that the image size is easy to obtain in general as it is normally made public by the fingerprint sensor vendors.

The efficiency of a hill-climbing attack may be evaluated by comparing the mean number of iterations needed to break each user account with the estimated number of attempts a brute-force attack would require [8]. The average number of attempts needed by a brute-force attack can be derived from the FAR of the system. A type 4 brute-force attack may be performed by sending real minutiae templates to the matcher until the system wrongly accepts one as corresponding to the template from the user's account under attack. Note that a brute-force attack using random synthetic templates would need more iterations than the number derived from the FAR as the FAR is calculated using real minutiae templates as inputs, not synthetic random ones.

3. Fingerprint Matching Systems

3.1. Reference System

We use the minutiae-based verification system from the NIST Fingerprint Image Software package (NFIS2) [9] as a reference system for our attacks. The architecture of the system and the point of attack, where the synthetic templates shall be introduced, are depicted in Fig. 2.

NFIS2 is a PC-based fingerprint processing and recognition system composed of independent software modules. In our experiments we will use two of the software modules included in NFIS2: MINDTCT and BOZORTH3. MINDTCT is the minutiae extraction subsystem. It generates an output text file containing the location, orientation and quality of each minutia from a fingerprint image input file. Direction maps and quality maps, among other output files are also generated for each image file.

BOZORTH3 performs the matching between any number of fingerprint templates which must have the same format as the output of MINDTCT. It is a rotation and translation invariant algorithm since it computes only relative distances and orientations. BOZORTH3 first constructs intra-compatibility tables, which are lists of associations between pairs of minutiae and their relative distance from the same fingerprint. It then looks for potential compatible minutiae pairs from the two fingerprints based on a specified tolerance and stores them in an inter-compatibility table. In the last step, it first traverses the inter-compatibility table, combining its entries into clusters, and then combines the clusters, building graphs. The larger the graph, the larger the match score will be. In our system, the match score is not normalized.

3.2. Match-on-Card (MoC) System

The Match-on-Card (MoC) system under consideration is a proprietary prototype. The matcher is fully embedded in a smart-card and may only be accessed via a smart-card reader connected to a PC.

One of the main differences between this system and the

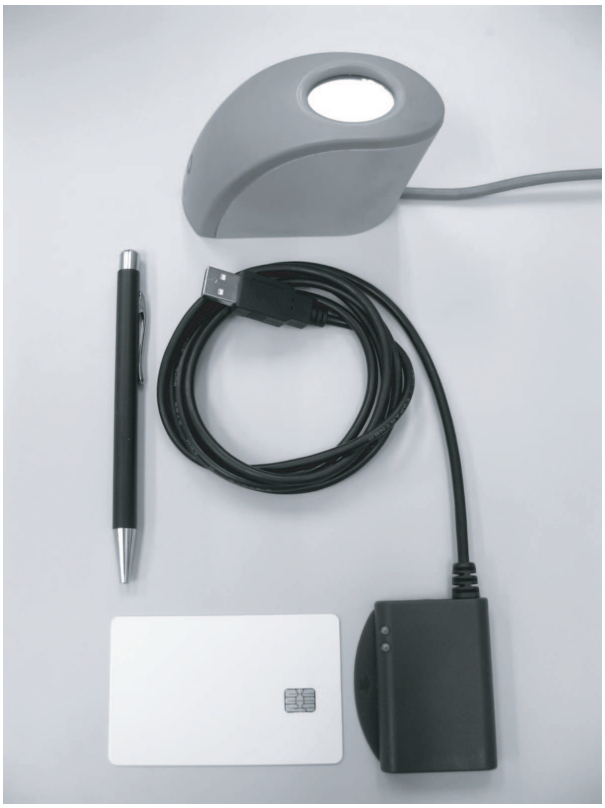


Figure 3. Top: Digital Persona fingerprint sensor used for acquiring the fingerprints used in our experiment [11]. Bottom: Smart-card and smart-card reader used in our experiments.

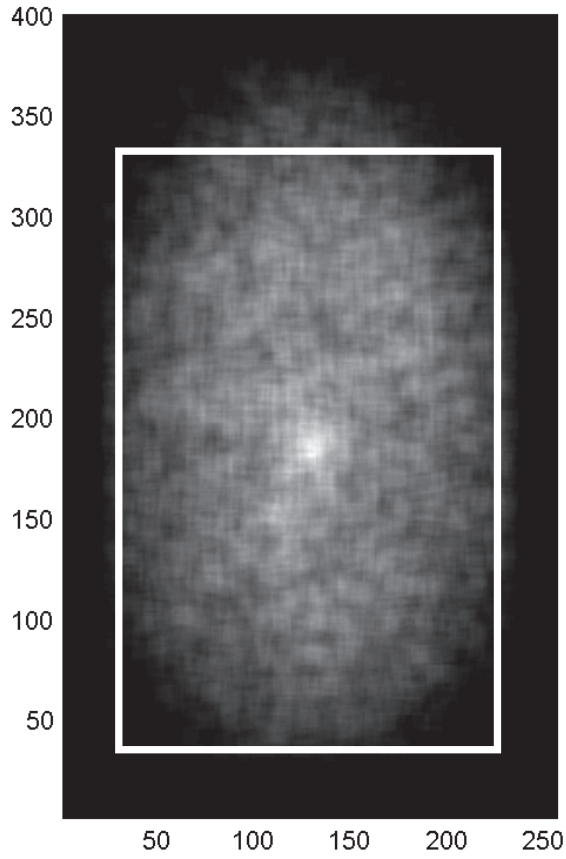


Figure 4. Minutiae location histogram from the selected subcorpus of MCYT [11]

reference system explained in Sect. 3.1 is that a MoC system is hardware limited. A smart-card has a very limited processing capacity and the matching algorithm should be efficient enough to perform the match in a reasonably short time. There has been much work in this field resulting in many algorithm proposals which try to reduce the matching computational cost [4, 10, 13]. The MoC system used in our experiments is shown in Fig. 3.

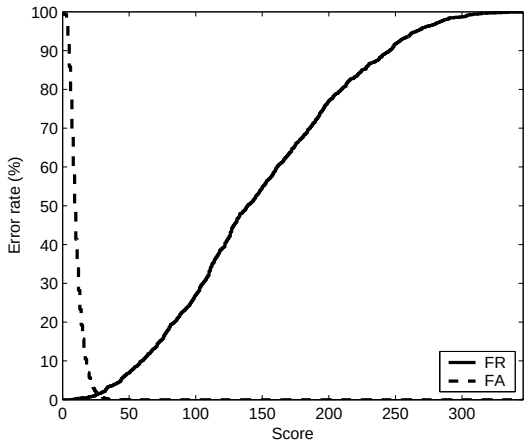
The smart-card reader is attached to a PC via USB, so the attacks can be performed from the PC. The user's fingerprint template is also stored in the smart-card memory. We only know the template storing format, which is also minutiae-based. In our experiments, we use the NFIS2 MINTDCT module (see Sect. 3.1) for the minutiae extraction phase and then perform the required transformations to the output file to make it compatible with this system. Note that the minutiae extraction phase would never be carried out by the smart-card, it must be done by an external application. The matcher returns the score as an integer value in a range from 0 to 100, 100 being the maximum likelihood between both fingerprints. The matching algorithm is unknown, but as the template format and the matching score are accessible, a hill-climbing attack may be performed.

4. Experiments

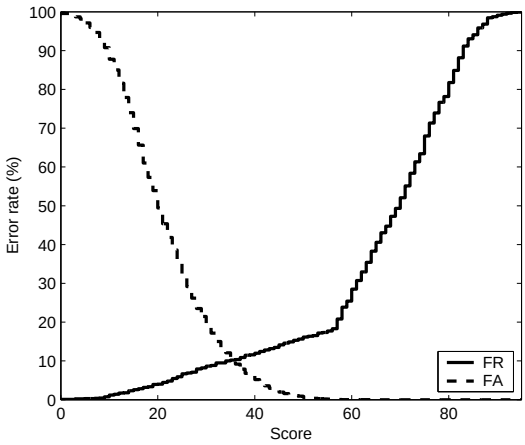
4.1. Database

The attack algorithms presented above have been tested on a sub-corpus from the MCYT database [11]. The fingerprint images are acquired with a 500 dpi optical sensor, model UareU by Digital Persona (see Fig. 3). We consider 10 samples from the right and left index fingers of 75 users, with 6 samples acquired with a high level of control [11] (i.e. small rotation or displacement from the center of the sensor), 2 with medium control level, and the last 2 samples with low control level. Therefore there are $75 \times 2 \times 10 = 1500$ samples.

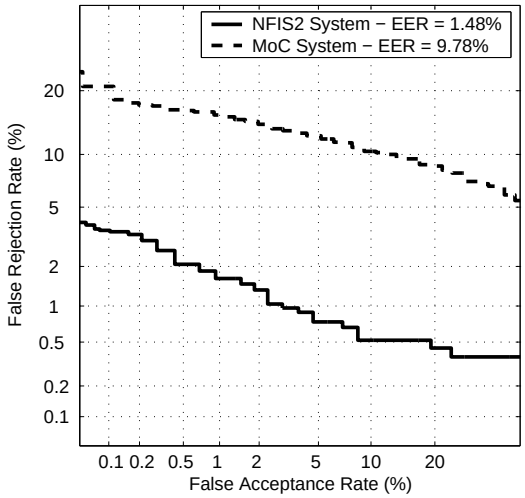
We compute the two-dimensional histogram of the minutia locations of all the fingerprints of the considered subcorpus. Fig. 4 depicts this histogram and a rectangle obtained heuristically that contains most minutiae. It can be seen that there are nearly no minutiae outside an elliptic region. Minutiae are nearly uniformly distributed in the rectangle with a higher concentration at its center. This rectangle will be used in our hill-climbing attacks as explained in Sect. 4.2. In the selected sub-corpus from the database



(a) FA and FR curves for the NFIS2 system



(b) FA and FR curves for the MoC system



(c) DET curves

Figure 5. Verification performance of both systems

described in Sect. 4.1, the mean number of minutiae is 38.

The verification performance of both systems is also studied. We use one of the low control samples as a template and the other 9 samples from the same finger as probes to test genuine matches, leading to $150 \times 9 = 1350$ genuine user scores. Impostor scores are obtained comparing each template to one sample from each other finger of the subcorpus, thus we have $150 \times 149 = 22350$ impostor scores. Fig. 5 depicts the FA, FR and DET curves from both systems.

4.2. Experimental Protocol

Our experiments are based on the ones presented in [8]. A number of 100 initial synthetic random templates are generated and sent to the matcher to attack a specific user account. Synthetic templates are generated with a fixed number of minutiae which is the mean number of minutiae in the fingerprints from the database (25 in [8]) and dividing the template into 9×9 pixel cells. A cell can contain only one minutia to avoid generating minutia which are closer than the inter-ridge distance.

The template that attains the highest matching score is saved. This template is iteratively modified by:

a) Perturbing an existing minutia by moving it to an adjacent cell or by changing its orientation.
b) Adding a minutia.
c) Substituting a minutia.
d) Deleting a minutia from the template.

If the matching score increases in any of these iterations, the modified template is saved, otherwise it is not.

In our experiments, we study the effects of different attack parameters by observing which iterations achieve, on average, more matching score increases during the attacks. We also study the influence of the initial number of minutiae and how these can be generated to improve the perfor-

ROI	Iterations	Initial Minutiae	Mean score raises				Success Rate	Total accounts broken in 5000 iterations
			a	b	c	d		
No	a, b, c, d	38	1,87	5,16	6,13	0,90	2/150	64/150
Yes	a, b, c, d	38	2,41	4,93	5,60	1,35	7/150	85/150

(a) Hill-climbing statistics using all iterations with and without ROI.

ROI	Iterations	Initial Minutiae	Mean score raises				Success Rate	Total accounts broken in 5000 iterations
			a	b	c	d		
Yes	a, b, c, d	38	2,41	4,93	5,60	1,35	7/150	85/150
Yes	a, b, c	38	3,18	7,70	7,91	-	28/150	145/150
Yes	b, c	38	-	9,25	9,76	-	40/150	143/150

(b) Hill-climbing statistics deleting low performing iterations.

ROI	Iterations	Initial Minutiae	Mean score raises				Success Rate	Total accounts broken in 5000 iterations
			a	b	c	d		
Yes	b, c	25	-	10,85	8,95	-	28/150	136/150
Yes	b, c	38	-	9,25	9,76	-	40/150	143/150
Yes	b, c	55	-	5,68	13,67	-	12/150	132/150

(c) Hill-climbing statistics using a different number of initial minutiae.

Table 1. Hill-climbing results on NFIS2

ROI	Iterations	Initial Minutiae	Mean score raises				Success Rate	Total accounts broken in 5000 iterations
			a	b	c	d		
Yes	b, c	10	-	7,70	5,30	-	65/150	133/150
Yes	b, c	25	-	5,53	10,08	-	123/150	146/150
Yes	b, c	38	-	3,55	13,27	-	78/150	139/150

(a) Hill-climbing statistics using a different number of initial minutiae.

ROI	Iterations	Initial Minutiae	Mean score raises				Success Rate	Total accounts broken in 5000 iterations
			a	b	c	d		
Yes	a, b, c, d	25	1,22	4,60	5,71	4,68	52/150	132/150
Yes	b, c, d	25	-	5,24	5,98	5,03	79/150	138/150
Yes	b, c	25	-	5,53	10,08	-	123/150	146/150

(b) Hill-climbing statistics deleting low performing iterations.

ROI	Iterations	Initial Minutiae	Mean score raises				Success Rate	Total accounts broken in 5000 iterations
			a	b	c	d		
Yes	b, c	25	-	5,33	10,08	-	123/150	146/150
No	b, c	25	-	6,13	9,15	-	91/150	148/150

(c) Hill-climbing statistics with and without rectangular ROI.

Table 2. Hill-climbing results on the Match-on-Card system

mance of our attacks. A first attack will be performed using the mean number of minutiae in our database and the four types of iterations.

In each attack, the 150 user templates are attacked using the same method (same initial minutiae generating scheme, same type of iterations and decision point) and statistics are gathered about the success rate of the attack. According to the results, the attacks are modified in order to better understand the factors involved in the success rate.

The synthetically generated templates will first have a random uniform minutiae distribution of 38 minutiae, which is the mean number of minutiae in our database.

For the NFIS2 system, we choose a decision threshold of 35 for the match score, leading to a 0.10% FAR and a 3.33% FRR. This means that a brute-force attack would theoretically need an average of 1000 iterations. For the Match-on-Card system a decision threshold of 55 is selected, resulting in a FAR of approximately 0.16% and a FRR of 17.33%. Thus, for the MoC system, 640 iterations would be needed by a brute-force attack.

We define the *success-rate* of an attack as the proportion of fingerprints for which the decision threshold is reached using less iterations than a brute-force attack. We establish a maximum of 5000 and 2000 iterations for the NFIS2 and the MoC system respectively.

4.3. Experimental Results

We first attack the NFIS2 system using the method described in [8]. As it has been said in Sect. 4.1, there is a region where it is most probable to find minutiae. From now on, we will refer to this region as the Region Of Interest (ROI). We subsequently run the algorithm considering only minutiae in the inside of the ROI, i.e. without generating, adding or displacing any minutia outside the ROI. Table 1.(a) studies the introduction of the ROI in the basic configuration of the attack, showing an improvement in the attack success rate when considering minutiae only within the ROI (from 2/150 to 7/150).

Next we focus on the influence of the different types of iterations. As it can be seen in Table 1.(a), each type of iteration achieves a different mean number of score raises during the attacks. Table 1.(b) shows the success improvements using only the best performing iterations. Iterations b and c, (add and substitute a minutia respectively) are the ones which achieve a higher rate of score raises, achieving a success rate of 40/150.

Finally, for the NFIS2 system, we study how the initial number of minutiae in the 100 synthetic initial random templates affect the attack performance. Table 1.(c) shows the different success rates for three different initial configurations. It can be seen that attacks with a number of initial minutiae different to the average in the database (38 in our case) perform much worse than those with this average number of initial minutiae. Fig. 6 shows the score progression and the minutiae maps of a successful hill-climbing attack on NFIS2 while Fig. 7 shows the same data for an unsuccessful attack.

For the Match-On-Card system, we start with the best configuration obtained by the NFIS2 system. We first study the influence of the initial number of minutiae, see Table 2.(a). As it can be seen, 25 initial minutia achieve better results (success rate of 123/150) than the mean number of minutiae (success rate of 78/150). This may be an effect of the limited capacity of the MoC matching algorithm.

In the next experiment, we study the influence of each iteration. Table 2.b shows the results obtained. Again, as on the NFIS2 system, iterations b and c are the most effective ones. The poor performance of iteration a points out that the MoC system is not very sensitive to small minutiae displacements or rotations. The attack score progression and the minutiae maps of a successful attack are depicted Fig. 8. The same data same data for an unsuccessful attack is shown in Fig. 9.

Finally we study the relevance of using the ROI under this configuration. In Table 2.(c) we see the decrease in performance without using the ROI (from 123/150 to 91/150 success rate).

5. Conclusions

In this paper, we have performed and studied hill-climbing attacks on the NFIS2 reference system and a Match-on-Card embedded system. NFIS2 is a PC-based fingerprint recognition system while the MoC system is a hardware limited system.

As it has been shown, the performance of hill-climbing attacks is heavily dependent upon the system under attack and the iterations that are performed. Attacks with a reduced number of minutiae are highly successful against the MoC system, while their performance against NFIS2 is very poor.

NFIS2 has proven to be more robust against hill-climbing attacks, at least with a reduced number of iterations. On the other hand, if we allow for a higher number of iterations (such as 5000 in our experiments), most accounts can be broken. It may be derived from the results that hill-climbing attacks are less effective than brute-force attacks, at least in the case of NFIS2. This statement must be taken with care, as hill-climbing attacks require much less resources than the ones needed by a brute-force attack. In fact, to perform an efficient brute force attack, the attacker must have a database of more than a thousand of real fingerprint templates, whereas there is no need for real templates in the case of a hill-climbing attack.

Acknowledgments

This work has been supported by the Spanish Ministry of Defense, BioSecure NoE and the TIC2003-08382-C05-01 project of the Spanish Ministry of Science and Technology. F. A.-F. and J. F.-A. thank Consejeria de Educacion de la Comunidad de Madrid and Fondo Social Europeo for supporting their studies.

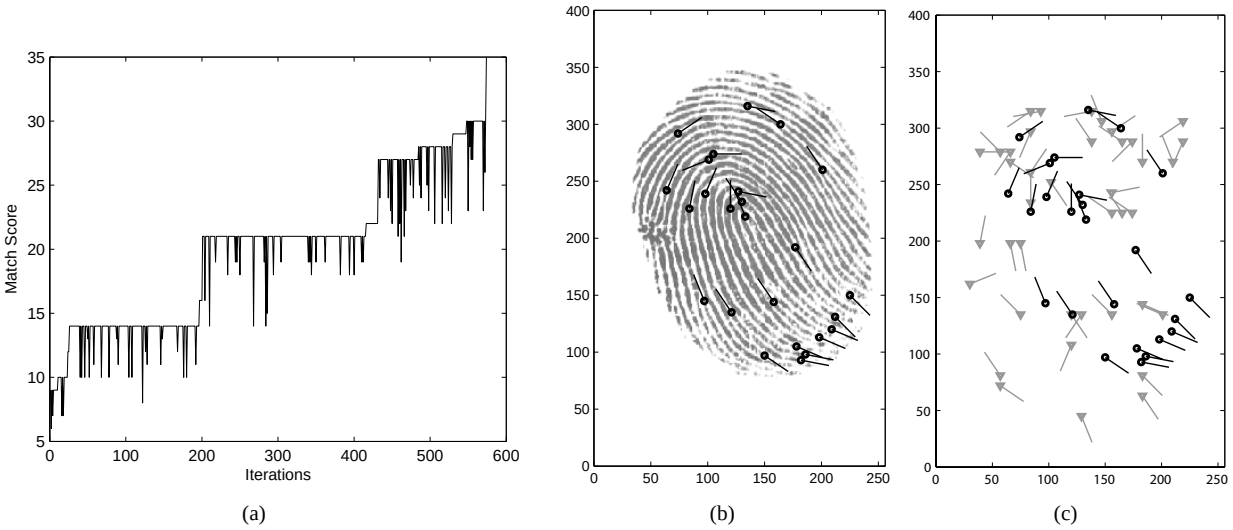


Figure 6. (a) Score progression, (b) original fingerprint minutiae, and (c) original minutiae (black circles) vs. synthetic minutiae (grey triangles) that achieve a higher score than the decision point on NFIS2 in a relatively short attack

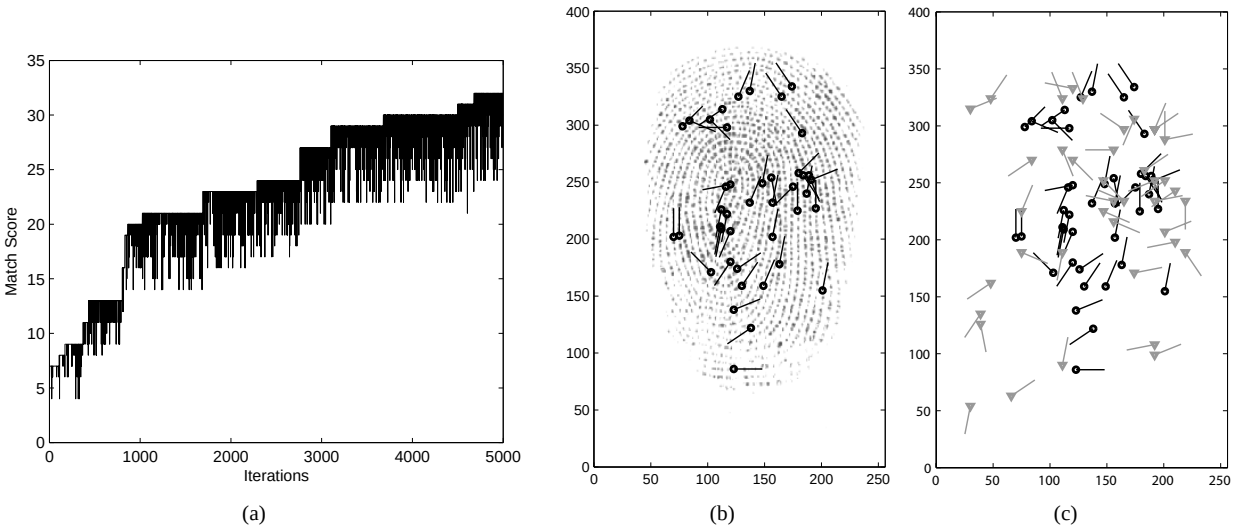
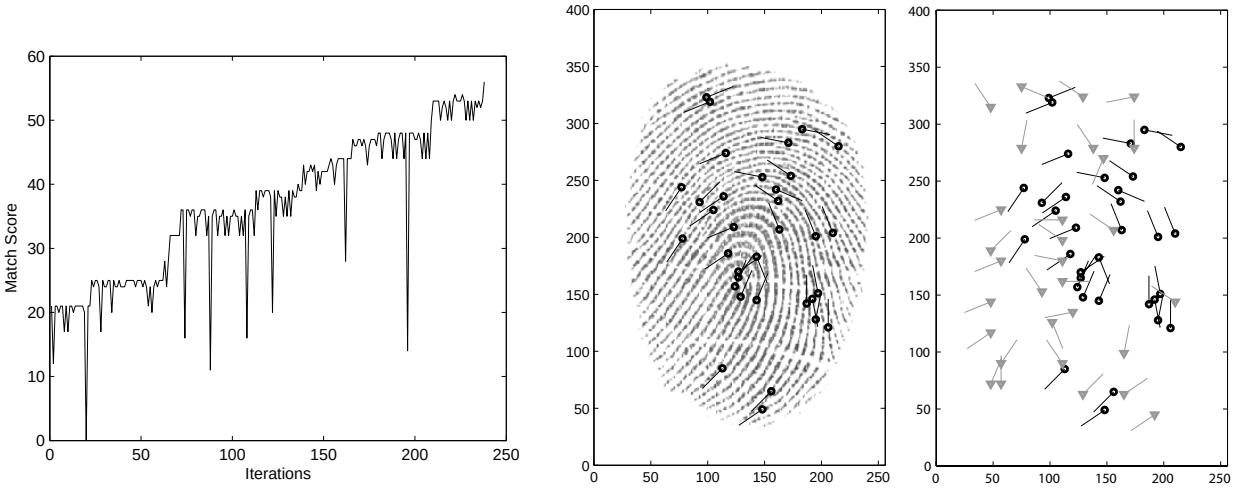
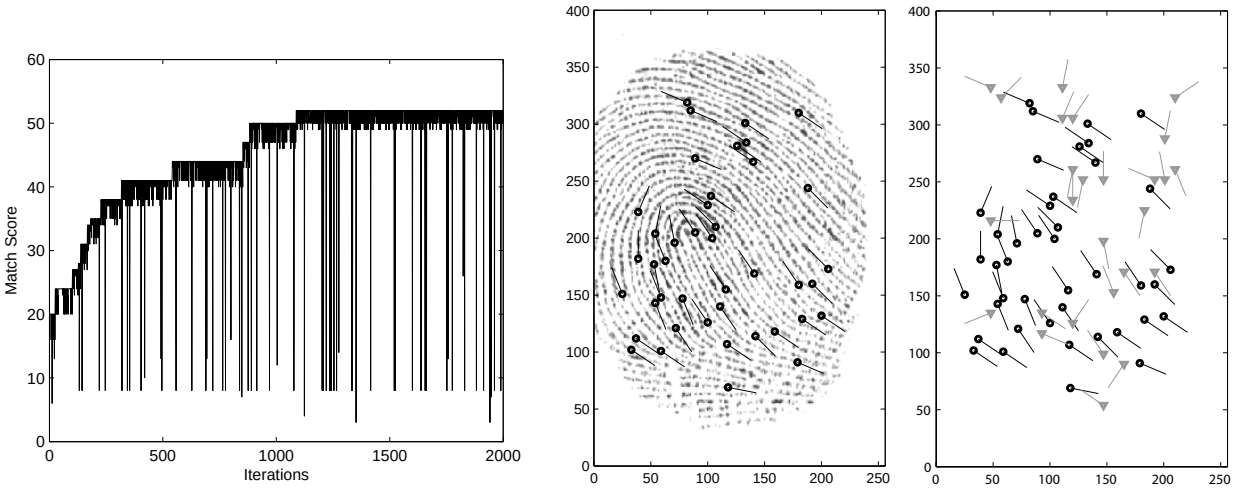


Figure 7. (a) Score progression, (b) original fingerprint minutiae, and (c) original minutiae (black circles) vs. synthetic minutiae (grey triangles) after 5000 iterations on NFIS2 in an unsuccessful attack



(a) (b) (c)

Figure 8. (a) Score progression, (b) original fingerprint minutiae, and (c) original minutiae (black circles) vs. synthetic minutiae (grey triangles) that achieve a higher score than the decision point on the MoC system in a relatively short attack



(a) (b) (c)

Figure 9. (a) Score progression, (b) original fingerprint minutiae, and (c) original minutiae (black circles) vs. synthetic minutiae (grey triangles) after 2000 iterations on the MoC system in an unsuccessful attack

6 References

[1] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, 2004.

[2] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer, 2003.

[3] A. K. Jain, A. Ross, and U. Uludag, "Biometric template security: Challenges and solutions," *Proc. of 13th European Signal Processing Conference (EUSIPCO), Antalya, Turkey*, 2005.

[4] R. Sanchez-Reillo, L. Mengihar-Pozo, and C. Sanchez-Avila, "Microprocessor smart cards with fingerprint user authentication," *IEEE AESS Systems Magazine*, vol. 18(3), pp. 22–24, March 2003.

[5] R. Cappelli, D. Maio, D. Maltoni, J. L. Wayman, and A. K. Jain, "Performance evaluation of fingerprint verification systems," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 3–18, 2006.

[6] FVC, "Fingerprint Verification Competition," 2006, (http://bias.csr.unibo.it/fvc2006).

[7] N.K. Ratha, J.H. Connell, and R.M. Bolle, "An analysis of minutiae matching strength," *Third International Conference on Audio- and Video-Based Biometric Person Authentication, Proc. AVBPA 2001*, pp. 223–228, 2001.

[8] U. Uludag and A. K. Jain, "Attacks on biometric systems: a case study in fingerprints," *Proc. SPIE-EI 2004, Security, Seganography and Watermarking of Multimedia Contents VI, San Jose, CA*, pp. 622–633, 2004.

[9] G.I. Watson, M.D. Garris, E. Tabassi, C.L. Wilson, R.M. McCabe, and S. Janet, *User's Guide to NIST Fingerprint Image Software 2 (NFIS2)*, National Institute of Standards and Technology (http://fingerprint.nist.gov/NFIS), 2004.

[10] M. Mimura, S. Ishida, and Y. Y. Seto, "Fingerprint verification system on smart card," *ICCE Digest of Technical Papers*, pp. 182–183, June 2002.

[11] J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon, J. Gonzalez, M. Faundez-Zanuy, V. Espinosa, A. Satue, I. Hernaez, J.-J. Igarza, C. Vivaracho, C. Escudero, and Q.-I. Moro, "MCYT baseline corpus: a bimodal biometric database," *IEE Proc. Vision, Image and Signal Processing*, vol. 150, no. 6, pp. 391–401, December 2003.

[12] Colin Soutar, "Biometric system security. http://www.bioscrypt.com/assets/security_soutar.pdf," 2002.

[13] FVC, "Fingerprint Verification Competition," 2004, (http://bias.csr.unibo.it/fvc2004).