Bayesian Hill-Climbing Attack and Its Application to Signature Verification

Javier Galbally, Julian Fierrez, and Javier Ortega-Garcia

Biometric Recognition Group–ATVS, EPS, Universidad Autonoma de Madrid,
C/ Francisco Tomas y Valiente 11, 28049 Madrid, Spain
{javier.galbally,julian.fierrez,javier.ortega}@uam.es

Abstract. A general hill-climbing attack algorithm based on Bayesian adaption is presented. The approach uses the scores provided by the matcher to adapt a global distribution computed from a development set of users, to the local specificities of the client being attacked. The proposed attack is evaluated on a competitive feature-based signature verification system over the 330 users of the MCYT database. The results show a very high efficiency of the hill-climbing algorithm, which successfully bypassed the system for over 95% of the attacks.

1 Introduction

Due to the advantages that biometric security systems present over traditional security approaches [1], they are currently being introduced in many applications, including: access control, sensitive data protection, on-line tracking systems, etc. However, in spite of these advantages they are not free from external attacks which can decrease their level of security. Thus, it is of utmost importance to analyze the vulnerabilities of biometric systems, in order to find their limitations and to develop useful countermeasures for foreseeable attacks.

Attacks on biometric systems can be broadly divided into: *i*) *direct attacks*, which are carried out at the sensor level using synthetic traits (e.g., printed iris images, gummy fingers); and *ii*) *indirect attacks*, which are carried out against the inner modules of the application and, therefore, the attacker needs to have some information about the system operation (e.g., matcher used, storage format). Ratha *et al.* in [2] made a more exhaustive analysis of the vulnerable points of biometric systems, identifying 8 types of possible attacks. The first point corresponded to direct attacks and the remaining seven were included in the indirect attacks group.

There are several works that study the robustness of biometric systems, specially finger- and iris-based, against direct attacks, including [3,4,5]. Some efforts have also been made in the study of indirect attacks to biometric systems. Most of these works use some type of variant of the hill-climbing algorithm [6]. Some examples include an indirect attack to a face-based system in [7], and to a PC and Match-on-Card minutiae-based fingerprint verification systems in [8] and [9], respectively. These attacks, which belong to types 2 or 4 of the classification

S.-W. Lee and S.Z. Li (Eds.): ICB 2007, LNCS 4642, pp. 386–395, 2007.

reported by Ratha *et al.* [2], take advantage of the score given by the matcher to iteratively change a synthetically created template until the similarity score exceeds a fixed decision threshold and the access to the system is granted. These hill-climbing approaches are all highly dependent of the technology used, only being usable for very specific type of matchers.

In the present work, a general hill-climbing algorithm based on Bayesian adaptation [10] is presented. The contribution of this new approach lies in the fact that it can be applied to any system working with fixed length feature vectors. The proposed attack uses the scores provided by the matcher to adapt a global distribution computed from a development set of users, to the local specificities of the client being attacked. We then present a case study where the attack is tested on a feature-based signature verification system using the 330 subjects of the MCYT database [11]. In the experiments the attack showed remarkable performance, being able to bypass over 95% of the accounts attacked for the best configuration of the algorithm found. Furthermore, the hill-climbing approach was faster than a brute-force attack in two of the three operating points of the system evaluated.

The paper is structured as follows. The general hill-climbing algorithm is described in Sect. 2, while the case study in signature verification is reported in Sect. 3. In Sect. 3.1 we present the attacked system, and the database and experimental protocol followed are described in Sect 3.2. The results are detailed in Sect. 3.3. Conclusions are finally drawn in Sect. 4.

2 Bayesian Hill-Climbing Algorithm

Consider the problem of finding a K-dimensional vector $\mathbf{y}$ which, compared to an unknown template $\mathcal{C}$ (in our case related to a specific client), produces a similarity score bigger than a certain threshold δ, according to some matching function J, i.e.: $J(\mathcal{C}, \mathbf{y}) > \delta$. The template can be another K-dimensional vector or a generative model of K-dimensional vectors. Consider a statistical model G (K-variate Gaussian with mean $\boldsymbol{\mu}_G$ and diagonal covariance matrix $\boldsymbol{\Sigma}_G$, with $\boldsymbol{\sigma}_G^2 = \text{diag}(\boldsymbol{\Sigma}_G)$), in our case related to a background set of users, overlapping to some extent with $\mathcal{C}$. Let us assume that we have access to the evaluation of the matching function $J(\mathcal{C}, \mathbf{y})$ for several trials of $\mathbf{y}$. The problem can be solved by adapting the global distribution G to the local specificities of template $\mathcal{C}$, through the following iterative strategy:

1. Take N samples $(\mathbf{y}_i)$ of the global distribution G, and compute the similarity scores $J(\mathcal{C}, \mathbf{y}_i)$, with $i = 1, \ldots, N$.
2. Select the M points (with $M < N$) which have generated higher scores.
3. Compute the local distribution $L(\boldsymbol{\mu}_L, \boldsymbol{\sigma}_L)$, also K-variate Gaussian, based on the M selected points.
4. Compute an adapted distribution $A(\boldsymbol{\mu}_A, \boldsymbol{\sigma}_A)$, also K-variate Gaussian, which trades off the general knowledge provided by $G(\boldsymbol{\mu}_G, \boldsymbol{\sigma}_G)$ and the local information given by $L(\boldsymbol{\mu}_L, \boldsymbol{\sigma}_L)$. This is achieved by adapting the sufficient statistics as follows:

$$\boldsymbol{\mu}_A = \alpha\boldsymbol{\mu}_L + (1-\alpha)\boldsymbol{\mu}_G \tag{1}$$

$$\boldsymbol{\sigma}_A^2 = \alpha(\boldsymbol{\sigma}_L^2 + \boldsymbol{\mu}_L^2) + (1-\alpha)(\boldsymbol{\sigma}_G^2 + \boldsymbol{\mu}_G^2) - \boldsymbol{\mu}_A^2 \tag{2}$$

5. Redefine $G = A$ and return to step 1.

In Eq. (1) and (2), $\boldsymbol{\mu}^2$ is defined as $\boldsymbol{\mu}^2 = \mathrm{diag}(\boldsymbol{\mu}\boldsymbol{\mu}^T)$, and α is an adaptation coefficient in the range [0,1]. The algorithm finishes either when one of the N similarity scores computed in step 2 exceeds the given threshold δ, or when the maximum number of iterations is reached.

In the above algorithm there are two key concepts not to be confused, namely: *i*) number of *iterations* (n_{it}), which refers to the number of times that the statistical distribution G is adapted, and *ii*) number of *comparisons* (n_{comp}), which denotes the total number of matchings carried out through the algorithm. Both numbers are related through the parameter N, being $n_{comp} = N \cdot n_{it}$.

3 Case Study: Attacking a Feature-Based On-Line Signature Verification System

3.1 Signature Verification System

The proposed Bayesian hill-climbing algorithm is used to attack a feature-based on-line signature verification system. The signatures are parameterized using the set of features described in [12]. In that work, a set of 100 global features was proposed, and the individual features were ranked according to their individual discriminant power. A good operating point for the systems tested was found when using the first 40 parameters. In the present contribution we use this 40-feature representation of the signatures, normalizing each of them to the range [0,1] using the tanh-estimators described in [13]:

$$p_k^{'} = \frac{1}{2}\left\{\tanh\left(0.01\left(\frac{p_k - \mu_{p_k}}{\sigma_{p_k}}\right)\right) + 1\right\}, \tag{3}$$

where p_k is the kth parameter, $p_k^{'}$ denotes the normalized parameter, and μ_{p_k} and σ_{p_k} are respectively the estimated mean and standard deviation of the parameter under consideration.

The similarity scores are computed using the Mahalanobis distance between the input vector and a statistical model of the attacked client $\mathcal{C}$ using a number of training signatures (5 in our experiments). Thus,

$$J(\mathcal{C}, \mathbf{y}) = \frac{1}{\left((\mathbf{y} - \boldsymbol{\mu}^{\mathcal{C}})^T (\boldsymbol{\Sigma}^{\mathcal{C}})^{-1} (\mathbf{y} - \boldsymbol{\mu}^{\mathcal{C}})\right)^{1/2}}, \tag{4}$$

where $\boldsymbol{\mu}^{\mathcal{C}}$ and $\boldsymbol{\Sigma}^{\mathcal{C}}$ are the mean vector and covariance matrix obtained from the training signatures, and $\mathbf{y}$ is the 40-feature vector used to attack the system.

3.2 Database and Experimental Protocol

The experiments were carried out on the MCYT signature database [11], comprising 330 users. The database was acquired in 4 different sites with 5 time-spaced capture sets. Every client was asked to sign 5 times in each set, and to carry out 5 skilled forgeries of one of his precedent donors, thus capturing a total 25 genuine signatures and 25 skilled forgeries per user.

The database is divided into a training (used to estimate the initial K-variate distribution G) and a test set (containing the user's accounts being attacked), which are afterwards swapped (two-fold cross-validation). The training set initially comprises the genuine signatures of the odd users in the database and the test set the genuine signatures of the even users. This way, the donors captured in the 4 sites are homogenously distributed over the two sets.

For each user, five different genuine models are computed using one training signature from each acquisition set, this way the temporal variability of the signing process is taken into account. With this approach, a total $330 \times 5 = 1,650$ accounts are attacked (825 in each of the two-fold cross-validation process).

In order to set the threshold δ, where we consider that the attack has been successful, the False Acceptance (FA) and False Rejection (FR) curves of the system are computed. In the case of considering skilled forgeries, each of the 5 estimated models of every user are matched with the remaining 20 genuine signatures ($5 \times 20 \times 330 = 33,000$ genuine scores), while the impostor scores are generated comparing the 5 statistical models to all the 25 skilled forgeries of every user ($5 \times 25 \times 330 = 41,250$ skilled impostor scores). In the case of random forgeries (i.e., impostors try to access other's accounts using their own signature), genuine scores are computed as above, while the set of impostor scores is generated matching the 5 user models with one signature of the remaining donors, making a total of $5 \times 330 \times 329 = 542,850$ random impostor scores. The FA and FR curves both for skilled (left) and random (right) forgeries are depicted in Fig. 1, together with three different realistic operating points used in the attacks experiments(FR=20%, FR=30%, and FR=40%). The similarity scores were normalized following the criterion described in Eq. (3).

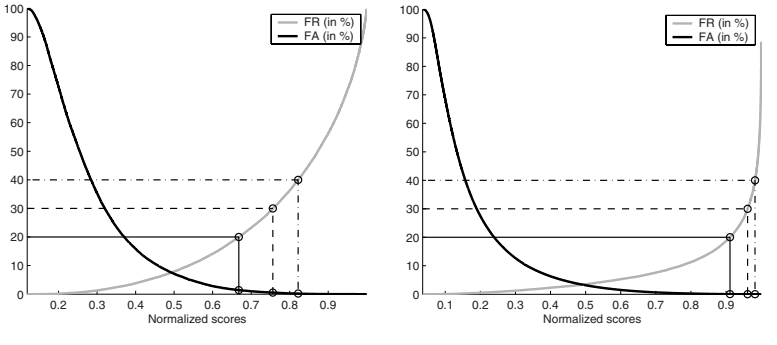

Fig. 1. FA and FR curves for skilled (left) and random (right) forgeries

Table 1. Success rate (in %) of the hill-climbing attack for increasing values of N (number of sampled points) and M (best ranked points). The maximum number of iterations allowed is given in brackets. The success rate (in %) appears in plain text, while the average number of iterations needed to break an account appears in **bold**.

		N				
		10 (2500)	25 (1000)	50 (500)	100 (250)	200 (125)
M	3	5.03 **24,082**	68.18 **11,292**	78.78 **9,725**	86.78 **10,611**	84.00 **14,406**
	5	2.72 **24,404**	71.27 **10,713**	85.57 **7,957**	92.00 **8,752**	91.09 **12,587**
	10		38.18 **17,598**	84.18 **8,609**	92.78 **8,602**	92.06 **12,261**
	25			41.33 **17,972**	89.57 **10,857**	91.63 **13,633**
	50				51.45 **18,909**	83.15 **16,660**
	100					39.39 **22,502**

3.3 Results

The goal of the experiments is to study the effect of varying the three parameters of the algorithm (N, M, and α), on the number of broken accounts, while minimizing the average number of comparisons (n_{comp}) needed to reach the fixed threshold δ. As described in Sect. 2, the above mentioned parameters denote: N the number of sampled points of the adapted distribution at a given iteration, M the number of top ranked samples used at each iteration to adapt the global distribution, and α is an adaptation coefficient.

Although the proposed hill-climbing algorithm and a brute-force attack are not fully comparable (for example, the resources required differ greatly as an efficient brute-force attack needs a database of thousands of signatures), in the experiments we compare n_{comp} with the number of matchings necessary for a successful brute-force attack at the operating point under consideration.

Analysis of N and M (sampled and retained points). For the initial evaluation of the algorithm, a point of [FR=30%, FA=0.01%] for random forgeries was fixed. This FA implies that an eventual brute-force attack would be successful, in average, after 10,000 comparisons. Given this threshold, the algorithm was executed for different values of N and M (fixing $\alpha = 0.5$) and results are given in Table 1. The maximum number of iterations (n_{it}) allowed for the algorithm appears in brackets. This value changes according to N in order to maintain constant the maximum number of comparisons permitted ($n_{comp} = N \cdot n_{it}$). In plain text we show the success rate of the attack (in % over the total 1,650 accounts tested), while the average number of comparisons needed for a successful attack is represented in **bold**.

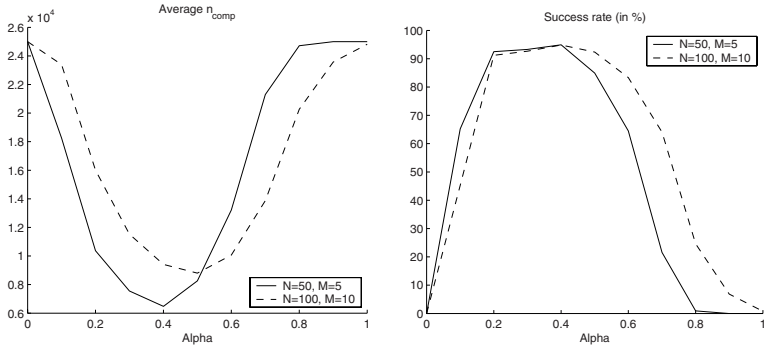

Fig. 2. Impact of α (adaptation coefficient) on the average number of comparisons needed for a successful attack (left), and on the success rate (right)

An analysis of the results given in Table 1 shows that for $N \gg M$, the points selected to estimate the local distribution are too specific and thus, the success rate of the attacks degrades slightly with respect to the best trade-off combination ($N \approx M$). On the other hand, if $N \simeq M$, the local distribution computed is too general, and the attack success rate is significantly reduced. The same effect is observed for the average number of comparisons (n_{comp}).

In this case, two good configurations of the parameters [N,M] can be extracted from Table 1, namely: i) [50,5], and ii) [100,10]. For these two points, the number of accounts broken is close to the total attacked, 85.57% and 92.78% respectively, while n_{comp} reaches a minimum (7,957 and 8,602, respectively) which is lower than the expected number of matchings required for a successful brute-force attack based on random forgeries (10,000 in average).

Analysis of α (adaptation coefficient). For the two best configurations found, the effect of varying α on the performance of the attack is studied sweeping its value from 0 (only the global distribution G is taken into account), to 1 (only the local distribution L affects the adaptation stage). The results are depicted in Fig. 2 where we show the evolution of n_{comp} (left), and the success rate (right), for increasing values of α and for the two configurations mentioned above.

It can be observed that for the point [50,5], the maximum number of accounts broken, and the minimum number of comparisons needed is reached for $\alpha = 0.4$ and both (maximum and minimum) are respectively greater and lower than those achieved with the values [100,10]. Thus, the best configuration of our algorithm is obtained for the values [N,M,α]=[50,5,0.4], which leads to 1,594 broken accounts (out of the 1,650 tested), and an average number of comparisons for a successful attack of 6,076, which represents almost half of the attempts required by a brute-force attack based on random forgeries. This value of α indicates that, for the best performance of the attack, the global and local distributions should be given approximately the same importance.

Table 2. Results of the proposed algorithm for different points of operation considering random and skilled forgeries for the best configuration of the proposed attack (N=50, M=5, $\alpha = 0.4$). The success rate is given in plain text (over a total 1,650), and n_{comp} in **bold**. The average number of matchings needed for a successful brute-force attack (n_{bf}) is also given for reference, together with the FA rate in brackets.

	Points of operation (in %)		
	FR=20	FR=30	FR=40
Success rate (in %)	98.12	96.60	94.90
n_{comp}	**5,712**	**6,076**	**6,475**
n_{bf} (random)	2,000 (FA=0.05)	10,000 (FA=0.01)	40,000 (FA=0.0025)
n_{bf} (skilled)	70 (FA=1.42)	180 (FA=0.55)	475 (FA=0.21)

Analysis of different operating points. Using the best configuration found, the algorithm was evaluated in two additional operating points of the system, namely (random forgeries): *i*) FR=20%, FA=0.05% (which implies a 2,000 attempt random brute-force attack), and *ii*) FR=40%, FA=0.0025%, where a random brute-force attack would need in average 40,000 matches before gaining access to the system. Results are given in Table 2 where the success rate over the total 1,650 accounts appears in plain text , and the average number of comparisons required by the bayesian hill-climbing attack in **bold**.

Smaller values of the FA rate imply a bigger value of the threshold δ to be reached by the algorithm, which causes a rise in the average number of iterations required for a successful attack. Compared to brute-force attacks, this increase of the number of iterations is significantly lower, which entails that the hill-climbing algorithm is clearly better than brute-force for FR rates over 25% and less effective for smaller values of the FR rate. Even though for some operating points the attacking strategy described in the present contribution is slower than a brute-force attack, it has to be emphasized that this latter approach would require, for instance in FR=20%, a database of 2,000 different signatures, which is not straightforward.

As described in Sect. 3.2 the genuine scores for the skilled forgeries case are computed the same way as in the random approach, therefore the FR rates remain unaltered. This means that the threshold δ to be reached by the hill-climbing algorithm is the same in both cases (comparing the proposed hill-climbing to either random or skilled brute-force attack), thus, the performance measures (success rate and number of comparisons n_{comp}) do not change. Only the FA values have to be recomputed and, as a result, the number of comparisons required by a successful skilled brute-force attack also change, being in the skilled forgery case: 70 for FR=20%, 180 for FR=30%, and 475 for FR=40%. These are significantly smaller than the average number of iterations needed by the hill-climbing algorithm, however, it has to be taken into account that in this case, for instance in FR=30%, we would need 180 different *skilled forgeries* of the same signer to access the system.

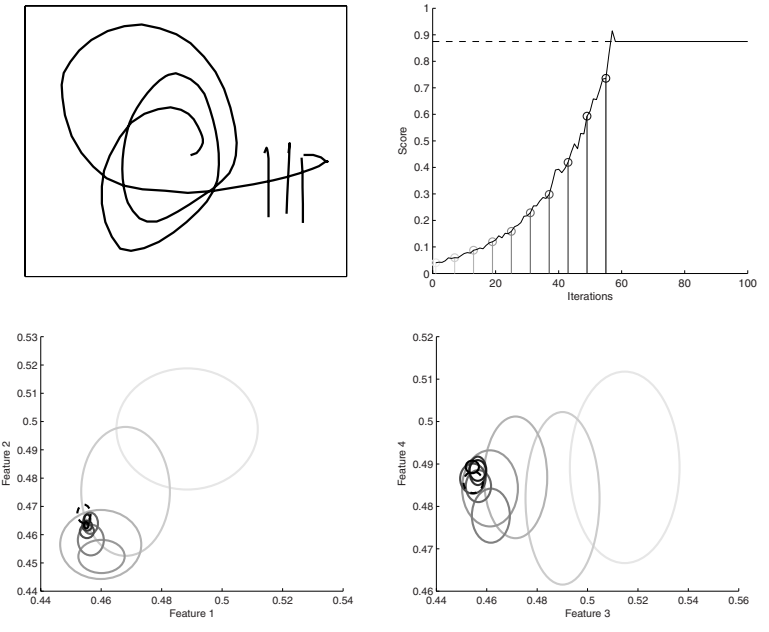

Fig. 3. Example execution of a successful attack, showing a sample signature of the attacked model (top left), evolution of the best score through the iterations (top right) with the threshold δ marked with a dashed line, and progress of the adapted distribution for the first two parameters (bottom left) and for the third and fourth parameters (bottom right). Lighter gray denotes a previous iteration, and the dashed ellipse the target model.

Graphical examples. Two example executions of the attack, at the FR=30% operating point and using the best algorithm configuration (N=50, M=5, α=0.4), are shown in Fig. 3 (successful attack) and Fig. 4 (unsuccessful attack).

In Fig. 3 a signature which was successfully attacked in very few iterations (57), is depicted. The evolution of the best similarity score through all the iterations is shown in the top right plot, where we can see how the threshold δ (dashed line) is quickly reached. In the bottom row we show the evolution followed by the two dimensional Gaussian distributions of the first two parameters (left), and of the parameters 3 and 4 (right). A lighter color denotes a previous iteration (corresponding to the highlighted points of the top right plot) and the dashed ellipse is the target distribution of the attacked model. It can be observed that the adapted distribution rapidly converges towards the objective model.

A sample signature of one of the few models which was not bypassed with the proposed algorithm is given in Fig. 4. The curves depicted are analog to the those plotted in Fig 3. The curves in the bottom row are zoomed versions of the squares shown in the pictures above, in order to show how in this case the adapted distribution does not converge towards the target model (dashed).

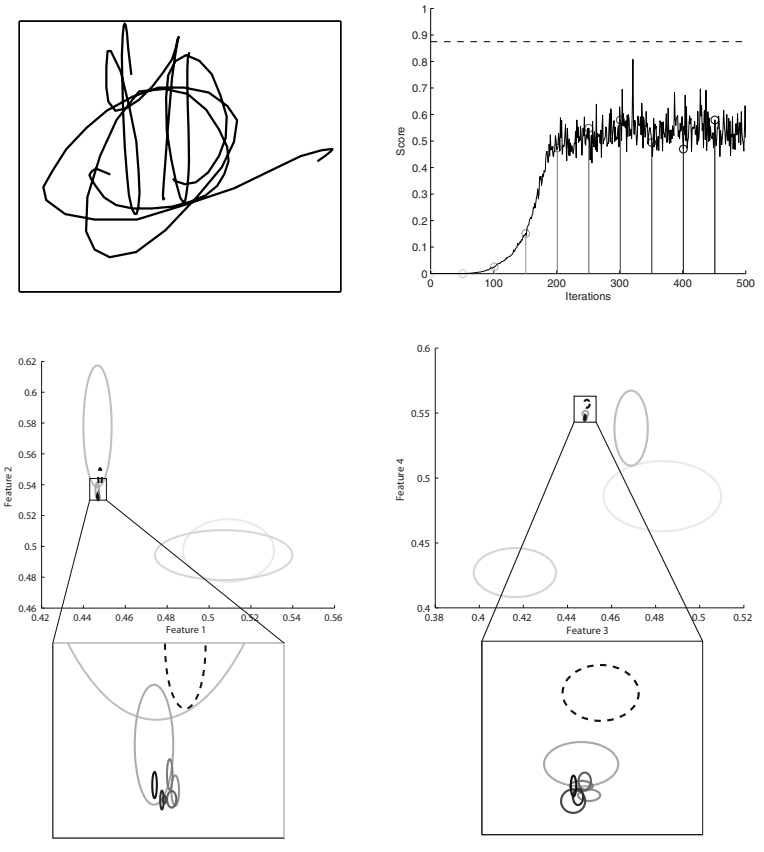

Fig. 4. Example execution of an unsuccessful attack. The images shown are analogue to those reported in Fig. 3. The bottom pictures are enlarged versions of the squares depicted in the above images.

4 Conclusions

A hill-climbing attack algorithm based on Bayesian adaptation was presented and evaluated on a feature-based signature verification system over a database of 330 users. The experiments showed a very high efficiency of the hill-climbing algorithm, reaching a success rate for the attacks of over 95% for the best algorithm configuration found.

The performance of the hill-climbing attack was directly compared to that of a brute-force attack. The algorithm described in the present contribution needed less number of matchings than the brute-force approach in two out of the three operating points evaluated when considering random forgeries. Worth noting that the resources required by both approaches are not comparable. In order to perform an efficient brute-force attack, the attacker must have a database of

more than a thousand real different templates, while the hill-climbing approach does not need real templates to be successful.

It has to be emphasized that the proposed algorithm is not thought for a specific matching strategy, and can be applied for the evaluation of the vulnerabilities of any biometric system based on fixed length templates.

Acknowledgements

J. G. is supported by a FPU Fellowship from Spanish MEC and J. F. is supported by a Marie Curie Fellowship from the European Commission. This work was supported by Spanish MEC under project TEC2006-13141-C03-03 and the European NoE Biosecure.

References

1. Jain, A.K., Ross, A., Pankanti, S.: Biometrics: a tool for information security. IEEE Trans. on Information Forensics and Security 1, 125–143 (2006)
2. Ratha, N., Connell, J., Bolle, R.: An analysis of minutiae matching strength. In: Proc. AVBPA, pp. 223–228 (2001)
3. van der Putte, T., Keuning, J.: Biometrical fingerprint recognition: don't get your fingers burned. In: Proc. IFIP, pp. 289–303 (2000)
4. Galbally, J., Fierrez, J., et al.: On the vulnerability of fingerprint verification systems to fake fingerprint attacks. In: Proc. IEEE of ICCST, pp. 130–136. IEEE Computer Society Press, Los Alamitos (2006)
5. Pacut, A., Czajka, A.: Aliveness detection for iris biometrics. In: Proc. ICCST, pp. 122–129 (2006)
6. Soutar, C.: Biometric system security, `http://www.bioscrypt.com/assets/security_soutar.pdf`
7. Adler, A.: Sample images can be independently restored from face recognition templates. In: Proc. CCECE, vol. 2, pp. 1163–1166 (2003)
8. Uludag, U., Jain, A.K.: Attacks on biometric systems: a case study in fingerprints. In: Proc. SPIE, vol. 5306, pp. 622–633 (2004)
9. Martinez-Diaz, M., Fierrez, J., et al.: Hill-climbing and brute force attacks on biometric systems: a case study in match-on-card fingerprint verification. In: Proc. IEEE of ICCST., pp. 151–159. IEEE Computer Society Press, Los Alamitos (2006)
10. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley, Chichester (2001)
11. Ortega-Garcia, J., Fierrez-Aguilar, J., et al.: MCYT baseline corpus: a bimodal biometric database. IEE Proc. Vis. Image Signal Process. 150, 395–401 (2003)
12. Fierrez-Aguilar, J., Nanni, L., et al.: An on-line signature verification system based on fusion of local and global information. In: Kanade, T., Jain, A., Ratha, N.K. (eds.) AVBPA 2005. LNCS, vol. 3546. Springer, Heidelberg (2005)
13. Jain, A.K., Nandakumar, K., Ross, A.: Score normalization in multimodal biometric systems. Pattern Recognition 38, 2270–2285 (2005)