A New Protocol to Evaluate the Resistance of Template Update Systems Against Zero-Effort Attacks

Romain Giot and Christophe Rosenberger
Université de Caen, UMR 6072 GREYC
ENSICAEN, UMR 6072 GREYC
CNRS, UMR 6072 GREYC
firstname.lastname@ensicaen.fr

Bernadette Dorizzi
Institut Mines Télécom-Télécom SudParis
UMR 5157 SAMOVAR
bernadette.dorizzi@it-sudparis.eu

Abstract

Since several years, there is a high interest in the creation of template update mechanisms in biometrics. By automatically updating over time the biometric reference of an individual, these update systems aim at avoiding a performance decreases over time. However, update errors occur when impostor samples are included in the biometric reference. In this paper, we are interested in analysing the behavior of template update against zero effort attacks depending on the difficulty of these attacks which is parametrized by the ratio of impostors samples included. We propose a protocol, which allows to evaluate an update mechanism under such kind of attacks by preserving an important amount of genuine samples, even when the ratio of impostor samples is important. Then, we apply this protocol on an existing template update system from the state of the art at various attack rates in order to assess its benefits.

1. Introduction

Several biometric systems suffer of problems of template ageing because of various reasons (lack of enrolment samples, slight modification of the appearance for a morphological modality, adaptation of the user to the device, ...). This impacts the performances by increasing the recognition error rate over time. Re-enrolling the user at a fixed frequency can solve the problem, but cost a lot because of the necessity to have an operator for ensuring the right user is re-enrolling. The aim of template update systems is to allow to automatically adapt the biometric reference over time in order to take into account the variation of the biometric data, and thus to reduce the performance decreases during the use of the biometric system. For example, static keystroke dynamics (which recognizes an individual considering its way of typing a predefined password) is a behavioral modality highly subject to template ageing [3, 6]. That is why it is interesting to use template update methods for this modality.

Wang *et al.* has recently analysed the behaviour of the biometric reference in the context of an online template update system attacked with the frog-boiling attack [13] for keystroke dynamics. The frog-boiling attack consists in presenting a query built with artificially generated keyboard events in a way specified by the attacker. The attacker generates keyboard events similar to the ones of the user and progressively modifies them in order to make them similar to a chosen target. This way, the template update makes drift the template which matches the target with False Acceptations. After the attacks, the Equal Error Rate (EER) of the various tested verification systems increased from between 9.9% and 18.9% to between 19.1% and 63.6%. However, although the attack is efficient and realistic in an operational context, it has been evaluated on a very weak template update system: each accepted query is used by the updating procedure. So, a lot of impostor samples can be used while updating, and attack results are over-estimated in comparison to how a smarter template update system would behave. Indeed, more sophisticated template update methods have been proposed quite recently. Rattani *et al.* use a dual-staged classification-selection approach for offline template update systems [11]. In the first stage, the unlabeled collected samples are labelled (client/impostors) with a graph based label propagation method. During the second stage, the informative genuine samples are selected based on a minimization of the estimated expected risk. The method has been validated on a fingerprint dataset collected during 9 sessions. In their protocol, 33% of impostors samples are randomly chosen and included in the query sets. Their update technique outperforms the self-update [9] by including more samples, the graph-min-cut [9], and the supervised scenario by including less erroneous or non informative samples. Giot *et al.* use several biometric references to represent a user, and each biometric reference evolves by using a different template update system [5]. The method has been validated on a keystroke dynamics dataset collected over 8 sessions. It outperforms the self-update [9] by reducing the number of update errors (inclusion of impostors samples, or forgetting of genuine samples). In their protocol, 30% of impostors samples are randomly chosen and included in the query sets.

Despite these previous studies, there is not yet any consensus on the way of evaluating template update systems. Usually, the number of impostors samples included in the sets of queries is low [11, 5] (less than 50%), and although one study analyses the recognition performance of template update systems under various attack rates [3], so far to our knowledge, there is not any study which analyses the evolution of the update errors depending on the attack rate for non biased template update systems (*i.e.* update decisions different from the authentication decision). Our objective in this paper is to propose such an analysis. In particular, the novelties of this paper are: (i) the proposition of a new method for creating pools[1] of genuine and impostors samples, depending on the strength (ratio of impostor queries against total number of queries) of the attack, for the evaluation of template update system. This new method must provide a better estimate of the performance with varying attack rates; and most important (ii) the analysis of the behavior (both in terms of authentication error and update error) of an existing template update scheme (only evaluated so far with 30% of attacks), under different attack rates, using the proposed method for the pool construction.

The paper is organized as follows. Section 2 presents our pool construction method. Section 3 presents the experimental protocol we adopted. Section 4 presents the experimental results obtained with this protocol as well as their comparison with the pool construction method of [3]. Section 5 discusses these results, and section 6 concludes this paper.

2. Proposed Method of Pool Construction

We make the assumption that the evaluation is done with a joint adapt and test strategy [8] adapted for a scenario with several sessions. So, the scores serve both for update process and for performance evaluation. As we think that variability can be correlated with the age of the template, the genuine samples are presented in a chronological order [2].

The proposed evaluation procedure is inspired from [3], but we have chosen to use a different method to build the pools of verification samples. In [3], authors build pools of *fixed* size N_F^Φ: whatever the ratio Φ of impostors included in the queries is, the total number of samples remains constant (with a high ratio of impostors, the number of genuine samples is very small): $N_F^\Phi = G_F^\Phi + I_F^\Phi = N$,with N the number of samples acquired during one session for one user, $G_F^\Phi = N * (1 - \Phi)$ the number of genuine samples and $I_F^\Phi = N * \Phi$ the number of impostor samples. We think it is not a correct way of building the pools as the number of genuine samples is very limited when there are many impostors samples. Few genuine samples means few correct update possibilities. That is why we propose a new construction method. We use pools of *variable* size N_V^Φ: the number of samples in the pool depends on the ratio of impostors: $N_V^\Phi = G_V^\Phi + I_V^\Phi$,with $G_V^\Phi = N = N_F^\Phi$ the number of genuine samples and $I_V^\Phi = N * \Phi/(1 - \Phi)$ the number of impostor samples. We think this way of building the pool is more appropriate as in the worst case there are always N genuine samples able to update the biometric reference instead of $N * (1 - \Phi)$ which can be very small and avoid to really update the biometric reference. Table 1 presents the method of sample choosing.

```
Require: $N$ the number of samples per session
Require: $S$ the number of the selected session
Require: $\Phi$ the probability of using an impostor value
Require: $i$ selected user
Mark all samples of all impostors available
$pool \leftarrow \emptyset$
$clients \leftarrow$ Get the $N$ samples of user $i$ for session $S$
$impostors \leftarrow \emptyset$
{Select the impostors samples}
for $nbimp = 0$ to $N * \Phi/(1 - \Phi)$ do
    $selected\_imp \leftarrow$ Randomly select an available impostor
    $selected\_sample \leftarrow$ Randomly select an available sample of $selected\_imp$
    Append $selected\_sample$ to $impostors$
    Mark $selected\_sample$ of $selected\_imp$ unavailable
    if All samples of $selected\_imp$ are unavailable then
        Mark $selected\_imp$ unavailable
    end if
end for
{Order the pool but keep client chronology}
$choice \leftarrow [\underbrace{0, \cdots, 0}_{N}, \underbrace{1, \cdots, 1}_{\frac{N*\Phi}{(1-\Phi)}}]$
Shuffle $choice$
for $i$ from 1 to $N + \frac{N*\Phi}{(1-\Phi)}$ do
    if $choice[i] == 1$ then
        $sample \leftarrow$ Pop the first element of $impostors$
    else
        $sample \leftarrow$ Pop the first element of $clients$
    end if
    Append $sample$ to $pool$
end for
return pool
```

Table 1: Pool construction for user i, session S and impostor rate Φ

[1] a pool is a set of ordered queries, which belong to the genuine user and to impostors, to be tested against the biometric reference

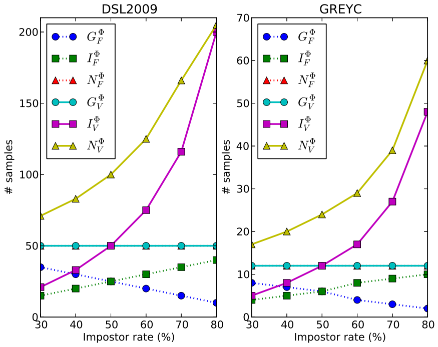


Figure 1: Number of genuine and impostor samples in each pool depending on the ratio of impostors Φ used to create the pool of [3] and this work. There is one pool constructed for each couple of session and user. In our pool construction method, there are always more samples used, which allow to have more accurate results.

3. Experimental Protocol

The evaluation protocol is inspired from [3] and we use the template update method presented in [5]. We consider the keystroke dynamics modality as it is a behavioral modality with a lot of temporal variations. However a similar protocol can also be used for other modalities.

Keystroke Dynamics Datasets There are few datasets publicly available for static keystroke dynamics. We evaluate the proposed study with the two largest public datasets of the literature. The GREYC dataset [4] consists of 5 sessions for 100 users. Each user typed 12 times per session the password “greyc laboratory” on two different keyboards on the same machine. The DSL2009 dataset [7] consists of 8 sessions for 51 users providing 50 samples per session, and each user has provided 400 samples. The password is “.tie5Roanl”.

Figure 1 presents the number of samples manipulated in each pool by using the two pool creations methods and the two datasets.

Keystroke Dynamics Authentication Method We have chosen a method presented in [1] as it is one of the best performing method when few enrollment samples are available. Each sample is composed of the duration of the key press of each key (H), the delay between each key press (DD), and the delay between a key release and the next key press (UD). So, the jth sample of user i, $\mathbf{x_j^i}$ is encoded as following $\mathbf{x_j^i} = [H_{j,0}^i, DD_{j,0}^i, UD_{j,0}^i, H_{j,1}^i, DD_{j,1}^i, UD_{j,1}^i, ...]$. As each user types the same password, all the $x_\bullet^\bullet$ have the same length. To build the biometric reference $\mathbf{r}^i$ of user i, we need a gallery of several samples: $\mathbf{g^i} = [\mathbf{x}_1^i, \mathbf{x}_2^i, ..., \mathbf{x}_M^i]$, where M is the number of samples in the gallery. We use all samples of the first session for enrollment and all samples of the other sessions for verification, update, and evaluation. Then, we compute the following information: the mean value $mean$ of the gallery; the median value $median$ of the gallery; the standard deviation std of the gallery; the maximum value $maximum$, and the minimum value $minimum$ between $mean$ and $median$.

The comparison score between the query $\mathbf{q}$ and the reference $\mathbf{r}^i$ is computed by counting the number of time values which verify the following equation:

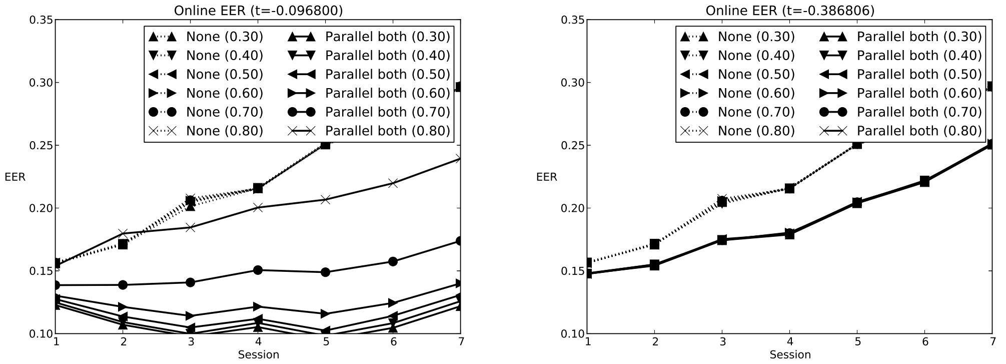
$$res[k] = \begin{cases} 1 & \text{, if } min[k] * \left(0.95 - \dfrac{std[k]}{mean[k]}\right) \le \mathbf{q}[k] \\ & \text{and } \mathbf{q}[k] \le max[k] * \left(1.05 + \dfrac{std[k]}{mean[k]}\right) \\ 0 & \text{, otherwise} \end{cases} \tag{1}$$

res is an array of 0 and 1. The first 1 is replaced by a 1.5, and the final score is computed as follows $score = 1 - mean(res)$.

Template Update Method We use the best performing template update method (Double Parallel) presented in [5]. This method uses a double threshold scheme: τ_{auth} serves for authenticating a user, while τ_{upd} serves for updating the biometric reference (if the user has been accepted). Each user i is represented by two biometric sub-references $\mathbf{r}_1^i$ and $\mathbf{r}_2^i$ computed with two distinct galleries $\mathbf{g}_1^i$ and $\mathbf{g}_2^i$. During the enrollment, $\mathbf{g}_1^i = \mathbf{g}_2^i$ as they contain exactly the same enrollment samples, and $\mathbf{r}_1^i = \mathbf{r}_2^i$. During the verification process, the query $\mathbf{q}$ is compared to each sub-reference $\mathbf{r}_1^i$ and $\mathbf{r}_2^i$ in order to obtain two scores $s_1^{i,\mathbf{q}}$ and $s_2^{i,\mathbf{q}}$, while the final score $s^{i,\mathbf{q}}$ is obtained by computing their average. If $s^{i,\mathbf{q}}$ is below the authentication threshold τ_{auth}, the individual is accepted, otherwise he is rejected and the procedure stops here. If $s^{i,\mathbf{q}}$ is below the update threshold τ_{upd}, $\mathbf{q}$ is used to update $\mathbf{g}_1^i$ and $\mathbf{g}_2^i$, and to recompute $\mathbf{r}_1^i$ and $\mathbf{r}_2^i$ (note that $\mathbf{q}$ can belong to an impostor). The *sliding window* (release of the oldest sample and addition of $\mathbf{q}$) is used to update $\mathbf{g}_1^i$, while the *growing window* (addition of $\mathbf{q}$) is used to update $\mathbf{g}_2^i$. The modification of a gallery implies the re-computing of its associated biometric reference. This way $\mathbf{r}_1^i$ and $\mathbf{r}_2^i$ give two different views of user i, which allows to reduce the template update error rate and to obtain well performing template update system.

Samples Order As there is a typing stability effect over sessions with keystroke dynamics due to the learning process [3, 12] it is necessary to keep the chronology of genuine samples (see section 2).

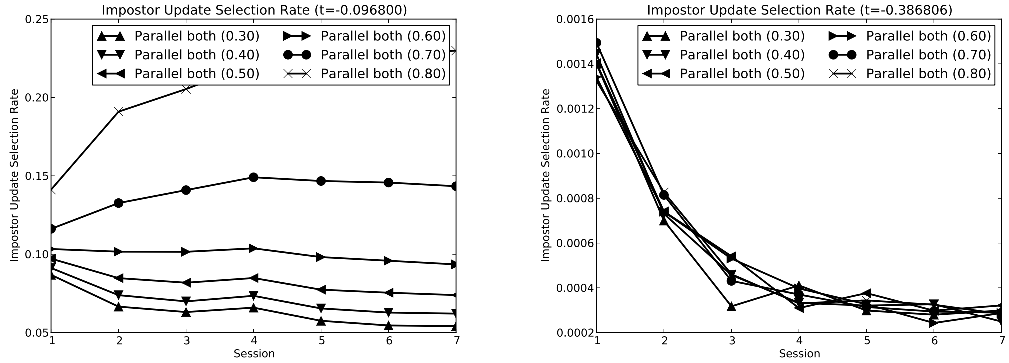
Threshold Configuration We have chosen to set τ_{upd} with the help of the enrollment samples under two distinct configurations: (i) *Permissive*: τ_{upd} is equal to the threshold giving the EER with the enrollment samples (session 0). (ii) *Stringent*: τ_{upd} is equal to the threshold allowing obtaining a False Match Rate (FMR) of 1% with the enrollment sam-



(a) Permissive threshold

(b) Stringent threshold

Figure 2: Equal Error Rate (EER) for various attack ratios (30%,40%,50%,60%,70%,80%), for two different update thresholds with the DSL2009 dataset



(a) Permissive threshold

(b) Stringent threshold

Figure 3: Impostor Update Selection Rate (IUSR) for various attack ratios (30%,40%,50%,60%,70%,80%), for two different update thresholds with the DSL2009 dataset

ples (session 0). A leave one out method is used to compute the scores. As in several works [11, 10, 3], performance is given by computing the EER (so τ_{auth} is not fixed).

4. Experimental Results

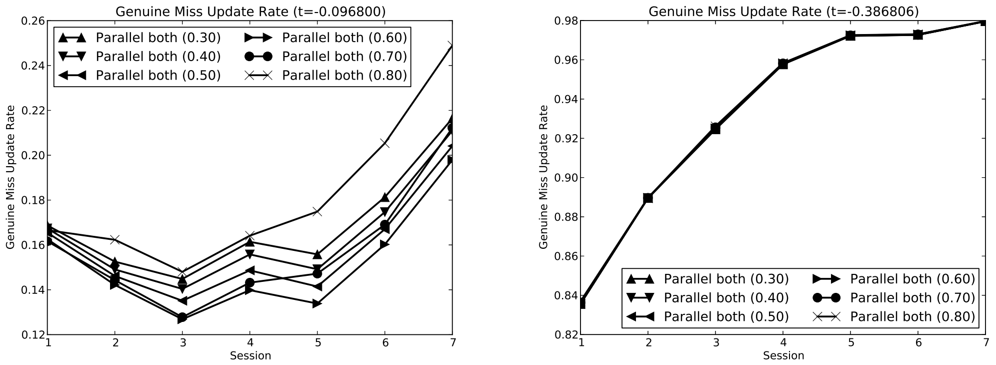
Although results are given for the DSL2009 dataset which contains more sessions than the GREYC dataset, a comparison between the two datasets is provided in the discussion section. As the pool creation method is stochastic, the evaluation procedure has been run 50 times, and we present the averaged results.

4.1. Comparison Between the Two Pools Creation Methods

Table 2 gives the difference of performances obtained between the two pool creation systems. For each update threshold and type of error, the performance results of each attack rate are stored in an ordered list. This table shows that the EER provided by our pool construction method is lower than the one provided by the method of [3] as well as the GUMR. More generally, the results provided by the two methods are different. We verify if the results obtained with the two pool creation methods are equal with the student t-test of paired samples. Figure 5 visually describes the performance difference of the EER for the two methods.

4.2. Performances Obtained With the New Pools Creation Method

Figure 2 presents the Equal Error Rate (EER) over sessions for a system with no template update, and a system with the hybrid template update tested under various attack rates. As expected, the EER over the sessions is the same



(a) Permissive threshold

(b) Stringent threshold

Figure 4: Genuine Update Miss Rate (GUMR) for various attack ratios (30%,40%,50%,60%,70%,80%), for two different update thresholds with the DSL2009 dataset

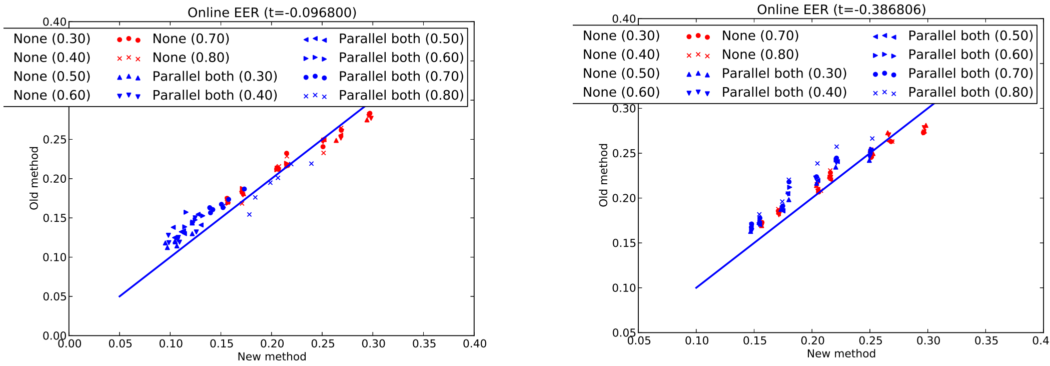
Table 2: Difference of results obtained by using the pool creation method of [3] (noted Old) and the proposed method (noted New). The p-value of the student paired test is given, and $p - value < 0.05$ are highlighted in bold (they represent a difference of result between the old and new method at 95%)

	EER			IUSR			GUMR		
Dataset	Old	New	P-value	Old	New	P-value	Old	New	P-value
Permissive update threshold									
DSL2009	0.19(0.05)	0.18(0.06)	**3.5e-06**	0.12(0.03)	0.11(0.05)	0.007	0.18 (0.03)	0.16(0.03)	**4.03e-07**
GREYC	0.15(0.02)	0.14(0.02)	**1.21e-0.8**	0.04(0.01)	0.03(0.01)	**8.43e-10**	0.36(0.02)	0.32(0.01)	**7.3e-10**
Strict update threshold									
DSL2009	0.20(0.05)	0.19(0.06)	**2.5e-14**	0.06(0.06)	0.06(0.07)	**0.001**	0.55(0.37)	0.55(0.39)	0.58
GREYC	0.15(0.02)	0.14(0.02)	**4.0e-16**	0.02(0.02)	0.01(0.02)	**4.83e-08**	0.65(0.29)	0.63(0.32)	**0.02**

whatever is the attack rate when using no template update system (the slight differences in the error rates come from the randomisation procedure). When the template update scheme is parameterized with a permissive threshold, and when there are between 30% and 50% of attacks, the EER are quite similar and far better than using no template update. However, when there are more than 50% of attacks, the performances decrease over sessions (the more important the impostor rate is, the more important the performances decrease is) and are closer to the ones when using no template update scheme. But these performances remain still better than when using no template update system. With the stringent threshold, the EER value over the sessions is similar whatever the ratio of impostor samples is. Performances remain still better than when using no update system, but are not far different.

Figure 3 presents the Impostor Update Selection Rate (IUSR) [5] over sessions for the system using template update with various attack rates. It consists of the ratio of impostor samples wrongly included during the update process. With the permissive threshold, the ratio decreases over sessions when there are between 30% and 50% of attacks, and is still lower than 10%. When there are more than 60% of attacks, the ratio of impostors samples included during the update increases over sessions. With the stringent threshold, whatever the ratio of attacks is, the IUSR decreases over sessions and is always inferior to 0.1%.

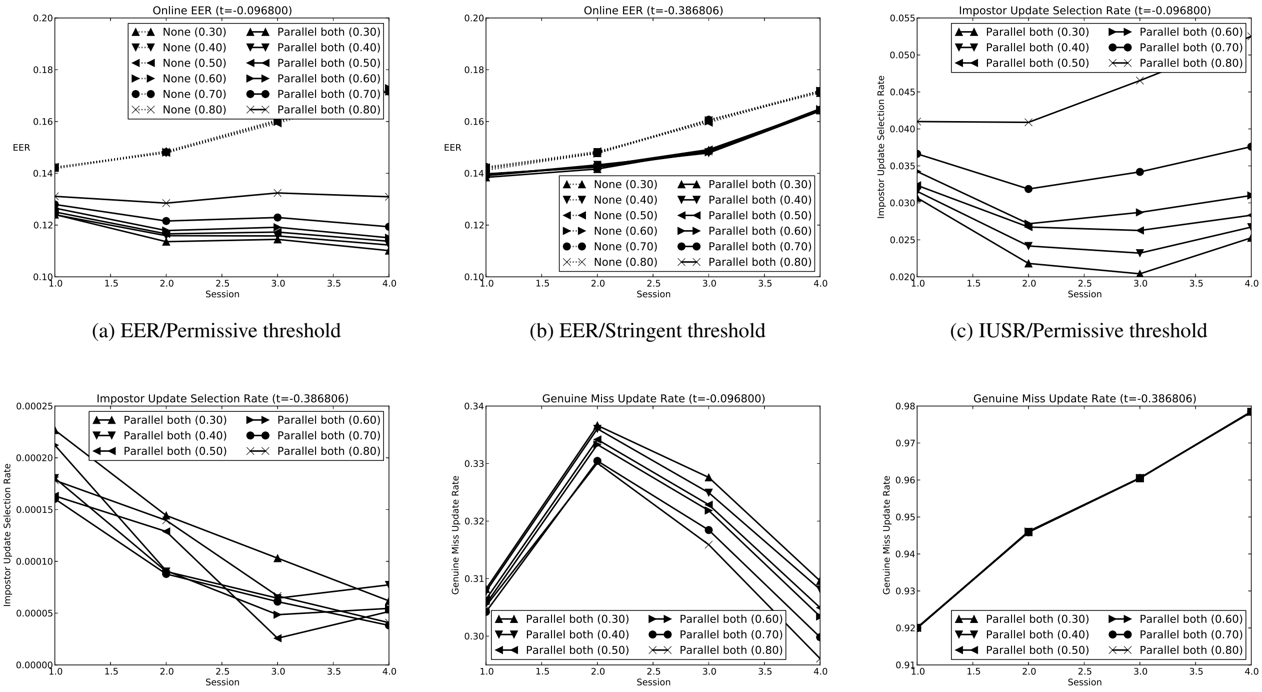
Figure 4 presents the Genuine Update Miss Rate (GUMR) [5] over sessions for the template update system with various attack rates. For both template update thresholds, the GUMR increases over sessions; however, the increase rate is far slower for the permissive threshold (with a GUMR always inferior to 26%) than for the stringent threshold (with GUMR always superior to 84%). For the system configured with the permissive threshold, the miss error rate is smaller for higher attack rates when the rates are between 30% and 50%. However, it is the opposite when the attack rates are between 60% and 80%. As for the IUSR, there are no fundamental differences of error rate between the different attacks rate and the stringent threshold.



(a) Permissive threshold

(b) Strict threshold

Figure 5: Difference of EER of various update systems and session computed with the method of [3] and the proposed method



(a) EER/Permissive threshold

(b) EER/Stringent threshold

(c) IUSR/Permissive threshold

(d) IUSR/Stringent threshold

(e) GMUR/Permissive threshold

(f) GMUR/Stringent threshold

Figure 6: Performances for the GREYC dataset which contains 5 sessions instead of 8

5. Discussion

The results provided by Table 2 are what we expected as more genuine samples are tested against the biometric reference. Thus, more samples can be used to update the biometric reference which better reflects the real biometric data of the user.

The template update system is more stable when configured with a stringent threshold in comparison to a permissive threshold. Indeed, when there are less than 50% of attacks, the performances of a system configured with the permissive threshold are far batter than for a system configured with the stringent threshold. However, the attack rate does not impact the update and authentication performances with the stringent threshold, whereas there is a great impact for the permissive threshold.

Even if for the permissive configuration more impostor samples are included, the ratio of clients missed is lower than for the no update system. It seems to cancel the fact of including impostors' samples. For this reason, even when there are 80% of attacks, the authentication performances with the permissive threshold are better than with the stringent threshold.

Figure 6 presents the results of the same experiments on the GREYC dataset. Globally the same behavior is observed for the various error rates, except for the GUMR which seems to decrease over time, instead of increasing for the permissive threshold. We note that the GUMR is higher for the GREYC dataset than for the DSL2009 (so we could expect a wrong model of the user as time goes on), but the EER always decreases over sessions when using a permissive threshold.

Note that in these scenarios, the attackers do not try to mimic the typing behavior of an impostor and do not try to alter the biometric reference in order to match a particular user, as for the Frog-Boiling attack [13]. In this reference, the authors test the robustness of the template update system when various different impostors try to impersonate one user. It is necessary to implement additional mechanisms to present update system to avoid frog-boiling attacks.

Although it is still an open problem for keystroke dynamics, liveness detection should be used within template update systems in order to avoid the inclusion of artificial samples.

6. Conclusion

Template update mechanisms are important modules for biometric behavioral authentication systems, as they allow reducing the performance decrease over time. However, the update is subject to errors when impostors try to impersonate a client by potentially allowing these impostors to alter the biometric reference of the client. That is why it is interesting and important to analyse the behavior of template update systems among various attack rates.

We have proposed a protocol which allows to test attacks at various rates while always keeping an important amount of genuine samples in order to more accurately evaluate the template update system. We have applied this protocol to an existing template update scheme of the state or the art which is evaluated on two update configurations (permissive and strict thresholds) at various attack rates. Results show that the proposed protocol allows giving better estimate of the performance and that the resistance to the attack depends on the severity of the threshold configuration of the template update system.

Although the method has been evaluated on a behavioral modality, it can also be used for morphological modalities. Future works should emphasize on the creation of a global error metric useful for the comparison of template update systems.

References

[1] T. de Magalhaes, K. Revett, and H. Santos. Password secured sites: stepping forward with keystroke dynamics. In *International Conference on Next Generation Web Services Practices*, 2005.

[2] A. Drygajlo, W. Li, and K. Zhu. Q-stack aging model for face verification. In *Proc. 17th European Signal Processing Conference (EUSIPCO 2009)*, 2009.

[3] R. Giot, B. Dorizzi, and C. Rosenberger. Analysis of template update strategies for keystroke dynamics. In *Computational Intelligence in Biometrics and Identity Management (CIBIM), 2011 IEEE Workshop on*, pages 21–28, 2011.

[4] R. Giot, M. El-Abed, and R. Christophe. Greyc keystroke: a benchmark for keystroke dynamics biometric systems. In *IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS 2009)*, pages 1–6, 2009.

[5] R. Giot, C. Rosenberger, and B. Dorizzi. Hybrid template update system for unimodal biometric systems. In *IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS 2012)*, pages 1–7, 2012.

[6] P. Kang, S.-s. Hwang, and S. Cho. Continual retraining of keystroke dynamics based authenticator. In *Proceedings of ICB 2007*, volume 4642, pages 1203–1211, 2007.

[7] K. S. Killourhy and R. A. Maxion. Comparing anomaly-detection algorithms for keystroke dynamics. In *39th Annual International Conference on Dependable Systems and Networks (DSN-2009)*, pages 125–134, 2009.

[8] N. Poh, R. Wrong, J. Kittler, and F. Roli. Challenges and research directions for adaptive biometric recognition systems. In *Advances in Biometrics*, pages 753–764, 2009.

[9] A. Rattani. *Adaptive biometric system based on template update procedures*. PhD thesis, University of Cagliari, 2010.

[10] A. Rattani, G. Marcialis, and F. Roli. Self adaptive systems: An experimental analysis of the performance over time. In *Computational Intelligence in Biometrics and Identity Management (CIBIM), 2011 IEEE Workshop on*, pages 36 –43, 2011.

[11] A. Rattani, G. L. Marcialis, E. Granger, and F. Roli. A dual-staged classification-selection approach for automated update of biometric templates. In *International Conference on Pattern Recognition (ICPR)*, 2012.

[12] M. M. Seeger and P. Bours. How to comprehensively describe a biometric update mechanisms for keystroke authentication. In *3rd International Workshop on Security and Communication Networks (IWSCN 2011)*, 2011.

[13] Z. Wang, A. Serwadda, K. Balagani, and V. Phoha. Transforming animals in a cyber-behavioral biometric menagerie with frog-boiling attacks. In *The IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS 2012)*, 2012.