

# An Experimental Analysis of the Relationship between Biometric Template Update and the Doddington's Zoo: A Case Study in Face Verification

Ajita Rattani, Gian Luca Marcialis, and Fabio Roli

University of Cagliari – Department of Electrical and Electronic Engineering  
{ajita.rattani,marcialis,roli}@diee.unica.it

**Abstract.** The problem of biometric template representativeness has recently attracted much attention with the introduction of several template update methods. Automatic template update methods adapt themselves to the intra-class variations of the input data. However, it is possible to hypothesize that the effect of template updating may not be the same for all the clients due to different characteristics of clients present in the biometric database. The goal of this paper is to investigate this hypothesis by explicitly partitioning clients into different groups of the “Doddington’s zoo” as a function of their “intrinsic” characteristics, and studying the effect of state of art template “self update” procedure on these different groups. Experimental evaluation on Equinox database with a case study on face verification system based on EBGM algorithm shows the strong evidence of non-uniform update effects on different clients classes and suggest to modify the update procedures according to the client’s characteristics.

## 1 Introduction

Recently, template representativeness has attracted much attention in the field of biometrics [1]. Often, the online operation of the biometric system encounters substantial intra-class variations in the input biometric data, due to the presence of several factors; like human-sensor interaction, environmental conditions, temporal variations (short term like scars in fingerprint surface and long term like aging in face) and other temporary variations like changes in facial expressions or rotation in fingerprint [2]. These variations make the initial enrolled templates, often captured in controlled environment, non-representative, thus resulting in degradation of the performance of the system.

To deal with this issue, novel solutions in the form of template update procedures have been introduced. Their aim is to adapt the enrolled templates to the intra-class variations of the input biometric data, on the basis of some learning methodologies like supervised [3] and semi-supervised learning [4-9]. Among these, supervised methods require the intervention of human experts for updating, thus making update procedure time consuming and inefficient task [3].

On the other hand, semi-supervised methods are automated systems that adapt themselves to the intra-class variations of the input biometric data. They derive their

name from the machine learning technique of semi-supervised learning that adapt the classifier through the joint use of initial labelled (templates) and unlabelled input biometric data. These data are available during the online operation, without the intervention of human supervisor [4-9]. Among others, self update is based on the self-training concept of semi-supervised learning, in which the input data recognized as highly confident are added iteratively to the template gallery set of the respective client [4-7].

Recently, self update procedure has gained much focus and the overall effectiveness of it has been proved experimentally in [4-7]. Also a serious issue of impostor's intrusion or "creep-in" errors, causing counter-productive effect, as an open issue has been pointed out in [9]. This problem of "creep-in" errors has also been recently investigated in detail [10] where the relationship between update errors due to impostor's introduction and performance degradation has been established. Ref [10] also suggested that the existence of creep-in of errors is also a function of user population characteristics apart from the basic FAR of the system or incorrect estimation of threshold parameters. Specifically in [10], the intrusion of impostors has been also correlated to clients which are intrinsically easy to "imitate", and clients "capable" to confuse themselves with other clients. These two groups have been called "lambs" and "wolves", respectively, according to the well-known concept of "Doddington's zoo" [11]. But their "presence" has not been recognized by using the rigorous definition given by Doddington et al. [11]. The presence of wolves, for example, has been detected by their repeated presence in different client's galleries at varying thresholds. Lambs have been identified by being attacked repeatedly at varying acceptance thresholds [10].

The aim of this paper is to extensively investigate the hypothesis that different types of clients as lambs, goats, wolves and sheeps [11] result in different updating effect on the application of self update procedure on the database, thus suggesting adaptation of template update methodology for each group of clients.

To this aim, user population is explicitly partitioned into different animal groups i.e., sheeps, lambs, wolves and goats using the definition of Doddington Zoo [11]. Then, the effect of global application of self update procedure has been studied on these classes of clients. Experimental evaluations are done on Equinox Database with 57 clients and 129 frontal face images per client [12]. The self update based technique has been used as a case study on face verification system based on Elastic Bunch Graph Matching (EBGM) [13]. Results pointed out that the effect of template update is different for each group, which confirms our hypothesis. This paper also mentions the counter-measures which may help to avoid the non-uniform effect of template update on these clients.

In Section 2, Self Update and its relation with Doddington zoo is described in detail. Section 3 reports the experimental results, and preliminary conclusions are drawn in Section 4.

## 2 Template Self Update and Doddington's Zoo

In the "online" template self update algorithm [4-7], a matcher adapts itself to the variations of the input data, available during the normal system's operation. The aim

of these methods is to capture the temporal and temporary intra-class variations of the input data by modifying the templates, thus enhancing the generalization performance on the novel unseen data.

The general “online” self update method may involve two steps: 1. Initialization: where each user is enrolled with its templates to build the initial gallery and the initial system parameter (“updating threshold”) is set; 2. Updating: where the input data is compared with the template(s) of the claimed identity’s gallery to compute the matching score. If matching score is greater than the threshold, the template set of the respective client is updated by either fusing with the current template or adding that sample as another instance into the gallery set of the claimed identity. If the matching score is less than the threshold, that sample is rejected. The described process is repeated on each availability of input sample.

Self update techniques usually operate at stringent acceptance threshold and exploit (i.e. add to the clients’ galleries) only the confidently classified samples in order to avoid the introduction of impostors into the gallery set of the client. But avoiding this problem is very difficult in a realistic verification system, and the impostor’s introduction leads to the so-called “creep in” error which strongly decreases the effectiveness of the update procedure.

Apart from incorrect threshold estimation conditions, these creep-in errors are also due to the presence of variable clients, like wolves, lambs and goats, present in the Doddington zoo as identified and reported in [10]. The Doddington’s definition of Wolves, Lambs, Goats are stated as follows [11]:

- Goats are clients intrinsically difficult to recognize: they exhibit a very high FRR;
- Lambs are clients easily imitable by other users: they exhibit a very high FAR;
- Wolves are the users (not necessarily clients) which have the capability to imitate other clients: i.e., when a wolf try to access as a certain client, it has a very high probability of success causing impostor’s introduction [11].

Even with the operation at stringent threshold conditions, presence of wolves and lambs do result in impostor’s intrusion. Work reported in [10] have discussed the threat of impostor’s introduction into the client’s galleries and highlighted that the presence of wolves and lambs cause the introduction of impostors into the gallery set of the clients even at the stringent threshold conditions. However, the presence of wolves and lambs have been shown on the basis of experimental evaluations, where wolves were determined by their repeated presence in the gallery set of other clients and lambs were identified by their repeated vulnerability to impostors at various threshold conditions. It was concluded that even with the operation at stringent threshold conditions, the intrusion of impostors is unavoidable due to the presence of clients like lambs and wolves in the biometric database [10]. Even, the unseen impostor input sample may also have the wolf characteristic apart from their presence in the database.

In this paper, we extend the above study and show explicitly the effect of update procedures not only for lambs and wolves but also for sheeps and goats. It has been hypothesized that due to the presence of these characteristic people in the database and the operation of the uniform self update procedure on them, goats may not be updated at the same strict threshold and lambs and wolves may attract impostors even

at the strict threshold conditions. Thus the effect of self update is studied specifically for each group by the prior division of database into different animal groups according to the definition of Doddington zoo [11].

### 3 Experimental Results

#### 3.1 Data Set

Equinox Corporation Database [12] consisting of 56 individuals with 129 face images per person with significant intra-class variations like illumination changes and variations in facial expressions etc, are used for experimental evaluation. The time span of the collected data sets is over one year. The self update based mechanism is used as a case study on face verification system based on Elastic Bunch Graph Matching [13]. Other face matching algorithm could be used as well.

#### 3.2 Experimental Protocol

In a typical personal verification system, a different batch of unlabelled set ( $D_u$ ), owing to different access attempts, is collected for each client over a period of time. In order to respect this simple evidence, the following protocol has been adopted in the literature [9-10]. We have also followed this protocol for our study.

(1) 56 initial templates are selected (one template for each client  $T$ ). These are the initial template set consisting of a neutral image per client ( $T_{ca}$ ). The threshold for self updating is always evaluated on this set, being the only set available in real environments. Threshold is evaluated on this template set by comparing each template to the templates ( $T$ ) of all the other clients ( $T_{ca}$ ) thus estimating the impostor distribution and selecting a threshold value at 0%FAR. These stringent starting conditions simulate the real environment where very less labelled data is available to set the system parameters.

(2) Remaining client images are subdivided in three sets with 25 images per client in the prediction set and 25 in unlabelled set and remaining 78 face images in the test set. Prediction dataset is used to partition the database clients into different animal groups.

(3) The whole dataset (except for templates) is then randomly partitioned into 56 partitions, such that the  $c$ -th partition does not contain images of the  $c$ -th client. Each of these partitions, consisting of 128 images, represent the “impostor set” for the  $c$ -th client: 25 images are added to unlabelled set, 25 images to the prediction set and 78 face images in the test set. The same number of genuine and impostors are added to the unlabelled, prediction and test set to have equal priors for both the classes, i.e., genuine and impostor. The database is partitioned into different animal groups of Doddington’s zoo using the samples from the prediction set. Then the unlabelled data set is used to update the enrolled templates using the self update procedure irrespective of the database partition. Then the test set is used to evaluate the actual improvement in performance gained

by the template self update algorithm. Then the performance of the updated system is evaluated using the test set for each partitioned group.

### 3.3 Rules for Identifying Different Animal Groups of Doddington's Zoo

This section presents the rules followed using the prediction set to divide the database into specific animals groups. Each client is enrolled using a single initial template. Note Ref [14] also partitions the database clients into different groups exhibiting variability on the basis of "sheepiness index" (sheeps are "well behaved" clients in the Doddington's zoo). However, the technique does not explicitly identifies which partition belongs to which specific animal group. It just partitions the database into different groups sharing the common characteristics. By considering the aim of this paper, we partitioned the database clients into specific animal groups according to the definition given by Doddington [11]. Thus:

- **Goats** are identified by evaluating FRR for each client at zeroFAR, which guarantees that no impostors are accepted. zeroFAR is calculated by matching each client's enrolled template to the relative impostor images of the prediction set. For each client, genuine samples belonging in the prediction set (25 samples) are matched with the initial enrolled template and FRR is evaluated for each client. Then, clients whose FRR exceeds a threshold value more than 0.5 are considered as "Goats".
- **Lambs** are identified by evaluating their FAR under an operational point of zeroFRR (zeroFRR is calculated by matching each enrolled template to the genuine samples of that client in the prediction set), which guarantees that no genuine users are rejected. Then for each client, a batch of impostor samples (25) from the prediction set are matched with the enrolled template and FAR is evaluated accordingly. Then, clients whose FAR exceeds a threshold value more than 0.5 are considered as "Lambs".
- **Wolves** are identified as follows. Each potential wolf (client) attacks all genuine samples (all 25 samples) of all the other clients in the prediction set. The operational point for each client is estimated at zeroFAR, calculated similarly as that for Goats. Candidate clients which are accepted even under this very stringent condition for more than 50% of the users are considered as "real" wolf.
- **Sheeps** are the clients which do not belong to any of the above presented group i.e., Lambs, Goats and Wolves.

Estimating zeroFRR is very difficult because of the small sample size affecting the data. However, by considering the goal of our study, the database partitioning has been made by considering quite large number of samples in order to have a good estimation of wolves, lambs and goats.

### 3.4 Results

The goal of the paper is to study the effect of template update procedure for different clients termed as members of Doddington's Zoo. Accordingly, first the biometric database is partitioned into different characteristics group of clients on the basis of

definition of Doddington Zoo [11] (section 3.3) and then the effect of self update procedure (section 2), updating threshold is set at zeroFAR, is analyzed on these client's group.

Table 1 reports the statistics about the percentage of specific "animals" present in the database, found using the prediction set of the Equinox database (section 3.3).

**Table 1.** Percentage of different characteristic clients in the database

% Lambs	% Wolves	% Sheeps	% Goats
12	2	56	30

Table 1 shows that Equinox database comprises of 12% lambs, 2% wolves, 56% sheeps and 30% goats. It is worth noting that the most frequent class is that of "sheeps", that is, clients "easy" to recognize and also difficult to imitate. Goats appeared as the second most frequent class. This contradicts results found in other works [14-15] (Ref. 15 is about fingerprints), and pointed out how the frequency of these classes strongly depend on the variety of the user population, prediction rules adopted and cannot be referred to general "frequency rules". Lambs and wolves are the least frequent classes.

As second step, we evaluated the effectiveness of self update method for each group of clients. We computed the average % of unlabelled samples exploited from the batch of unlabelled set (section 3.2) and the average % of impostors introduced for each group (available in the same unlabelled set). These statistics, presented in Table 2, help to evaluate the behaviour of self update on different client's group.

**Table 2.** Average percentage of samples added to clients galleries, impostors introduced, and successful % of wolves attacks after self updating

Animal Type	(%) Samples Exploited (added to clients gallery)	(%) Impostors Introduced (among samples exploited)	(%) Wolves attacked (attack by all the detected wolves with 25 samples per wolf)
Wolves	56.0	37.0	-
Lambs	65.7	43.2	65.9
Goats	23.3	11.8	26.3
Sheeps	72.6	16.4	41.0

Table 2 shows the different percentage of impostors and available samples exploited, from the unlabelled batch, for each partitioned group. According to the statistics presented in Table 2, the following analysis can be done:

- 1) lambs, i.e. clients vulnerable to impostors, have introduced a high percentage of unlabelled samples, but about half of them are impostors. They are also strongly affected by detected wolves;
- 2) goats, i.e. clients difficult to be recognized, have exploited the lowest number of available samples, thus confirming their intrinsic characteristic according to the definition of Doddington. On the other hand, they are much less prone to impostors introduction (only 11.8% of added samples are impostors) than other classes of clients, and also are less prone to wolf attacks;
- 3) sheeps, i.e. well behaved clients, have exploited the largest number of samples and the % impostor introduction is also minimum for these clients. Even % of wolf attacks is minimum for these clients ;
- 4) wolves too are intruded by impostors. This class exhibits the additional feature that can attack other clients, namely, lambs, and can be confused with other impostors, even lambs. This confirms evidences in previous work [10], where it is stated that certain “lambs” may be “wolves” for other clients too [11].

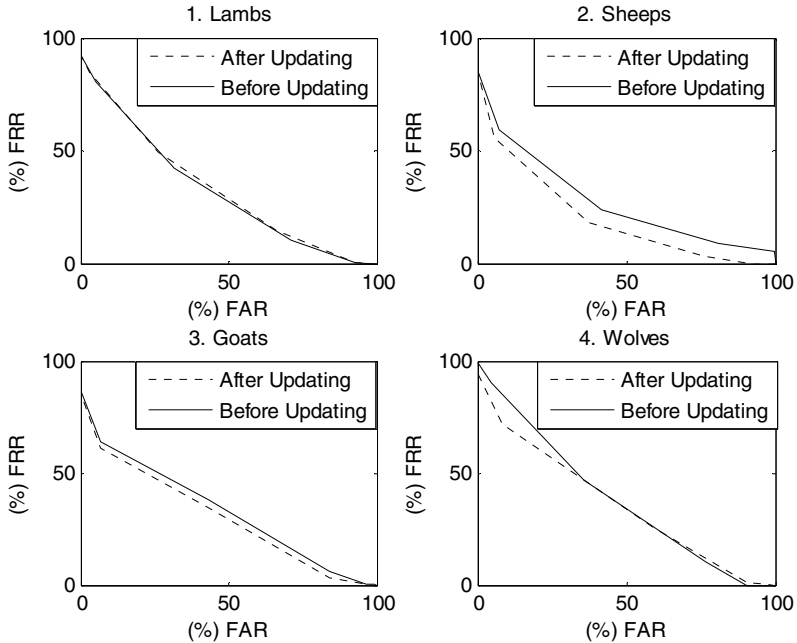
### 3.5 Performance Enhancement Reported for Each Group of Clients

According to the statistics reported in section (3.5), it is evident that the effect of self update is different for each partitioned clients group. To provide a further evidence, the average performance attained due to updated templates after the self update procedure for each group of clients is reported in Fig. 1. Fig. 1 presents the ROC curves on test set, before (straight line) and after updating process (dotted line), for each group of clients. In ROC curves, x axis is % FAR and y axis % FRR.

As can be seen from Fig. 1, initial performances of the verification system for each client group (straight lines) is different and after the update procedure, the impact is different for each group of clients (dotted lines). Lambs (subplot 1), the clients vulnerable to impostors, have shown no substantial improvement, because of the large amount of impostors introduction into the galleries (table 2, second row, column 1). Sheeps, as expected, being well behaved clients, have upgraded in the performance after updating process. Nevertheless, they too have attracted impostors into galleries, but because of much more percentage of genuine samples introduced, the impact of impostors is less, hence the performance is upgraded (table 2, fourth row, column 1&2). Goats, difficult to recognize clients, have shown slight improvement in performance due to less capture of genuine samples, although very less percentage of impostor samples are added to the related gallery (table 2, third row, column 1&2). Finally, wolves appears to benefit from self update only at threshold value set at high-security level, where low FAR is expected. Different performances after updating and the statistics presented in table 2 clearly showed the different behaviour of self update procedure for the investigated groups of clients.

These evidences suggest that adjustable update procedures have to be adopted, depending on the clients class. In the following, we give some preliminary guidelines: (1) firstly, a user-specific update threshold can be adopted for each group of clients [10]; (2) lambs and wolves could be updated by adopting supervised methods, as even with the operation at stringent acceptance threshold, they are prone to impostors introduction. Thus, the manual assignment of labels to the samples utilized for updating may minimize the probability of impostors introduction [3]. We believe that manual

assignment of label can be suitable, due to their small percentage in the user population: the cost of updating should be compensated by benefits in improving the performance; (3) goats may operate at relaxed updating threshold, say 1%FAR, for better capturing large intra-class variations [9], since they are less prone to impostors introduction.



**Fig. 1.** ROC Curves showing the initial performance (straight line) and after template updating (dotted lines), on the test set, for each partitioned class of clients

## 4 Conclusions

This paper studied the hypothesis of different behaviour of the self update algorithm due to the presence of different characteristics of clients in the database. Accordingly using the Doddington's classification, the database is partitioned and self update process is applied. Our hypothesis has been confirmed by reported experimental results.

These preliminary results pointed out that the suitability of self update must be carefully analysed on the target user population, although no significant performance degradation have been noticed on the four identified groups of users, especially for "anomalous" clients (lambs and wolves). On the other hand, performance improvement for the most frequent classes, namely, sheeps and goats, has been pointed out. Since they result in covering about 80% of user population, applying update algorithms to these user appears to be worth-while. On the other hand, adjustable updating procedures may be taken into account for wolves and lambs. We suggested some "guide-lines" by considering the consistence of wolves, lambs, goats and sheeps in the user population at hand.



In our opinion, with regard to above guide-lines and their suitability, the main problem to solve is to clearly understand if frequency of users classes, which in our study did not partially agree with other achievements at the state of the art, is general. It can be easily supposed, and maybe verified, that this frequency may change as function not only of users, but also of selected biometric and prediction rules and the set used for clients partitioning. But, although frequency of different classes may change, the behaviour of self update for each partitioned class may remain consistent.

## References

1. Roli, F., Didaci, L., Marcialis, G.L.: Adaptive Biometric Systems that Can Improve with Use. In: Ratha, N., Govindaraju, V. (eds.) *Advances in Biometrics: Sensors, Systems and Algorithms*, pp. 447–471. Springer, Heidelberg (2008)
2. Jain, A.K., Flynn, P., Ross, A.: *Handbook of Biometrics*. Springer, New York (2007)
3. Uludag, U., Ross, A., Jain, A.K.: Biometric template selection and update: a case study in fingerprints. *Pattern Recognition* 37(7), 1533–1542 (2004)
4. Jiang, X., Ser, W.: Online Fingerprint Template Improvement. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(8), 1121–1126 (2002)
5. Ryu, C., Kim, H., Jain, A.K.: Template Adaptation based Fingerprint Verification. In: 18th ICPR 2006, vol. 4, pp. 582–585. IEEE Computer Society, Los Alamitos (2006)
6. Roli, F., Marcialis, G.L.: Semi-supervised PCA-based face recognition using self-training. In: Yeung, D.-Y., Kwok, J.T., Fred, A., Roli, F., de Ridder, D. (eds.) *SSPR 2006 and SPR 2006*. LNCS, vol. 4109, pp. 560–568. Springer, Heidelberg (2006)
7. Liu, X., Chen, T.H., Thornton, S.M.: Eigenspace updating for non-stationary process and its application to face recognition. *Pattern Recognition*, 1945–1959 (2003)
8. Roli, F., Didaci, L., Marcialis, G.L.: Template co-update in multimodal biometric system. In: Lee, S.-W., Li, S.Z. (eds.) *ICB 2007*. LNCS, vol. 4642, pp. 1194–1202. Springer, Heidelberg (2007)
9. Rattani, A., Marcialis, G.L., Roli, F.: Capturing large intra-class variations of biometric data by template co-updating. In: *IEEE Workshop on Biometrics, International Conference on Vision and Pattern Recognition, Anchorage, Alaska, USA* (2008)
10. Marcialis, G.L., Rattani, A., Roli, F.: Biometric template update: an experimental investigation on the relationship between update errors and performance degradation in face verification. In: da Vitoria Lobo, N., Kasparis, T., Roli, F., Kwok, J.T., Georgiopoulos, M., Anagnostopoulos, G.C., Loog, M. (eds.) *S+SSPR 2008*. LNCS, vol. 5342, pp. 684–693. Springer, Heidelberg (2008)
11. Doddington, G., Liggett, W., Martin, A., Przybocki, M., Reynolds, D.: Sheep, goats, lambs and wolves: A statistical analysis of speaker performance, NIST 1998 speaker recognition evaluation. In: *Proceedings of ICSLP 1998* (1998)
12. <http://www.equinoxsensors.com/products/HID.html>
13. Wiskott, L., Fellows, J.-M., Kruger, N., von der Malsburg, C.: Face Recognition by elastic bunch graph matching, Tech. Report IR-INI 96-08 (1996)
14. Poh, N., Kittler, J.: A Methodology for Separating Sheep from Goats for Controlled Enrollment and Multimodal Fusion. In: *6th IEEE Biometric Symposium, Tampa, Florida, USA* (2008)
15. Micklin, A., Watson, C., Ulery, B.: The Myth of Goats: how many people have fingerprints that are hard to match?, Tech. Rep. NISTIR 7271, NIST (2005)