

Template Update Methods in Adaptive Biometric Systems: A Critical Review

Ajita Rattani, Biagio Freni, Gian Luca Marcialis, and Fabio Roli

Department of Electrical and Electronic Engineering,
University of Cagliari, Italy

{ajita.rattani,biagio.freni,marcialis,roli}@diee.unica.it

Abstract. Template representativeness is a fundamental problem in a biometric recognition system. The performance of the system degrades if the enrolled templates are un-representative of the substantial intra-class variations encountered in the input biometric samples. Recently, several template updates methods based on supervised and semi-supervised learning have been proposed in the literature with an aim to update the enrolled templates to the intra-class variations of the input data. However, the state of art related to template update is still in its infancy. This paper presents a critical review of the current approaches to template updating in order to analyze the state of the art in terms of advancement reached and open issues remain.

1 Introduction

Biometrics refers to the use of physiological or biological characteristics to recognize the identity of an individual. These characteristics are unique to each individual and remain unaltered during the individual's lifetime [1].

A typical personal biometric authentication system operates in two distinct modes: enrolment and verification. In enrolment mode, individual's biometric sample(s) are captured, processed, features extracted and labels are assigned to them to establish identity, representing the template(s) of an individual. Verification mode, compares query biometric samples of the respective individual with the enrolled template(s) to verify an identity. It is quite difficult to have a perfect match ('100%') between template and a genuine query image, due to the presence of several factors: 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 change in facial expression or affine transformation in fingerprints. These factors introduce variations in the query image, resulting in performance degradation and making enrolled templates, usually acquired in controlled environment, un-representative [2].

The solution lies in making the biometric system "adaptive" to the intra-class variations of the input data. The standard approaches to deal with this problem are acquiring multiple impressions of each individual, making the threshold parameters adaptive, or adopting multi- biometrics [1].

Recently, novel solutions have been introduced in the form of template updating where the classifiers adapts itself to the intra-class variations based on learning

methodologies like supervised or semi-supervised learning. Although these methods are promising, the state-of-art related to them is still in its infancy. A unified taxonomy is missing along with the lack of clear statement of the problem [3]. Therefore, to further stimulate research on this important topic of template unrepresentativeness and their solution as update procedures, this paper critically reviews the template update methods to biometrics. The aim of this survey is to highlight current state-of-the-art with respect to current approaches, in particular, the learning methodology adopted and experimental evaluation followed. Our goal is also to give an insight into the open issues that still remain to be addressed.

Thus, we propose a taxonomy in which the current template update methods can be classified (Section 2). We discuss and compare some of the key features of these methods (Section 2.1 and 2.2), and the usual template update and management procedure adopted (Section 3). We have also analysed different protocols followed for the performance evaluation with respect to the threshold parameter set, the number of unlabelled data used for updating the templates, properties of database used, and percentage recognition accuracy enhancement reported in the literature, traits covered so far etc. As all these factors are very vital information to be taken into consideration for a fair evaluation of each update method (Section 4). Section 5 concludes the paper and discusses some open issues that still remain to be considered.

2 Biometric Template Update Methods: A Taxonomy

Template representativeness is a crucial problem in the biometrics, as the input biometric data is subject to on-going changes due to presence of intra-class variations thus making the initial enrolled templates non representative of them. As a result, performance degradations arise. To deal with this issue, adaptive biometric systems, based on appropriate “template update” methods, have been proposed [3]. The basic step in these methods are the assignment of identity labels, considering which a possible taxonomy is proposed in Figure 1. Identity label assignment can be done either in a completely supervised manner (i.e. by human expert intervention), or by using automatic “learning” methodology (“semi-supervised”). The key difference between “supervised” [2, 4-5] and semi-supervised learning [6-13, 15] is the technique followed for the data labelling. In supervised template update methods, the label assignment is manual, whilst in semi-supervised methods, it is automatic.

Supervised methods (Fig. 1, Vertex 1) proposed so far are usually offline [2, 4-5] as they operate on a batch of collected samples. On the other hand, semi-supervised methods (Fig. 1, Vertex 2) are automated methods that assigns identity labels to the unlabelled data on the basis of their own knowledge, derived through current enrolled templates without the intervention of human supervisor [6-13, 15]. These methods avoid the cost related to manual assignment of labels to unlabelled data used for updating. Therefore, they can also operate online.

The second basic step is, which is also followed in the taxonomy, is the template selection procedure. Among all templates, manually or automatic labelled, it is necessary to select the most “representative” ones, in terms of intra-class variations. Accordingly, supervised methods can be divided on the basis of template selection techniques (Fig. 1, Vertex 1.1) employed to reduce the gallery size, that is, clustering

(Fig. 1 Vertex 1.1.1) [2], and editing [4] (Fig. 1 Vertex 1.1.2). On the basis of the same selection principle, semi-supervised methods (Fig. 1 Vertex 2) can be further subdivided into: single modality [6-9, 12] (Fig. 1 Vertex 2.1) and multiple modality (Fig. 1 Vertex 2.2) [10, 11, 15] on the basis of whether the selection and update process operates in a uni-modal or multimodal environment. The single modality methods are further categorized as based on self-training [6-9] and graph-based [12]. Methods based on self-training (Fig. 1 Vertex 2.1.1) updates themselves iteratively using the highly confident input data [6-9] and graph based methods [12] (Fig. 1 Vertex 2.1.2) work by analysing the structure of the input data in the form of a graph [7]. The self-training based methods are further categorized as online (Fig. 1 Vertex 2.1.1.1) [6-8] or offline methods (Fig. 1 Vertex 2.1.1.2) [9] on the basis if the update is performed as soon as the input data arrives or after the batch of data has been collected during the system operation. Feature editing has also been employed for online methods [6-7] (Fig. 1 Vertex 2.1.1.1.1). The proposed graph based [12] and multimodal update methods [10, 11, 15] are offline methods (Fig. 1 Vertex 2.1.1.2). According to the presented taxonomy, the proposed template update methods in the scientific literature have been critically reviewed.

Sections 2.1 and 2.2 discuss methods in the proposed taxonomy in more detail.

2.1 Supervised Methods

Supervised methods (Fig. 1 Vertex 1) to template update have been introduced in [2, 4-5]. These methods update the system to the batch of unlabelled data collected during the system operation. They require the intervention of supervisor for assigning identity labels to the collected data which are used for updating. The labels assigned during the normal system operation are stripped off and the positively labeled samples by the human expert are added to the template set for updating. These are offline methods as the updating task is performed when the batch of input data has been collected after a fixed time interval. The supervisor should have a knowledge of the correct intra-class variations as noisy biometric data. For example, a noisy fingerprint image due to dirt accumulation on fingerprint sensor is also a difficult sample, but does not represent a genuine intra-class variation. Therefore, supervised methods which exploit properties of fingerprints in terms of features used for their comparison (e.g. minutiae), perform automatic template selection (Fig.1 Vertex 1.1.).

These methods can be classified on the basis of the type of techniques used for selecting the representative templates: clustering-based [2] (Fig.1 Vertex 1.1.1) and editing-based [4] (Fig.1 Vertex 1.1.2). The former are based on applying standard clustering algorithms to each updated template gallery, and selecting an appropriate template from each cluster. To this aim, two algorithms, namely, MDIST, based on finding the templates which minimize the intra-distances among samples in the gallery, or DEND, based on the so-called “dendogram” [17], in order to select the templates exhibiting large intra-class variations, have been proposed. In latter [4], each gallery is viewed as a training set of a NN classifier. Accordingly, if a large batch of samples is available, the aim is to “edit” it in order to find the subset exhibiting the minimum size and also keeping the same classification accuracy of the training set. In [4], several existing editing algorithm have been applied with promising results. In particular, it has been noticed that editing based approaches are

not dependent on the “number of clusters” (i.e. number of templates), an input parameter for clustering-based approaches, set manually. Moreover, it has been argued that editing approach could help in retaining variable number of templates for each individual, by detecting easy and difficult clients, for which a small number of templates are enough or which require large number of templates, for a good classification accuracy [4].

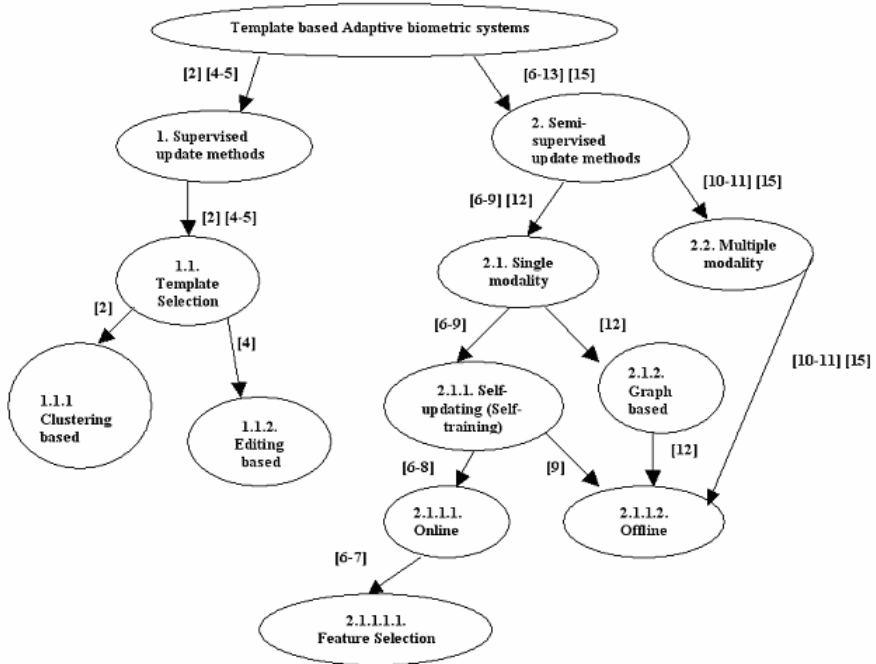


Fig. 1. The proposed taxonomy for the template update methods. The edges are labeled by the reference number of the proposed methods.

2.2 Semi-supervised Methods

As an alternative to supervised approaches, semi-supervised methods (Fig.1 Vertex 2) have been introduced [3, 6-13, 15]. “Semi-supervised” methods derive their name from the machine learning approaches related to the joint use of labeled and unlabelled data for improving the classification accuracy, named as “semi-supervised learning” [18]. In this topic, term “labeled data” is used to denote initial training samples, and “unlabelled data” to the samples available during system operation and “pseudo-labeled” by the classifier. In biometric applications, the labeled data are the initial enrolled templates and the unlabeled ones are the input biometric data collected during the system operation. Since classification approach is threshold-based in biometrics, labeled and unlabelled data are jointly used to modify such threshold and the composition of galleries by adding “pseudo-labeled” data to the related client’s gallery.

Semi-supervised methods can be subdivided into “single-modality” (Fig. 1 Vertex 2.1) and “multi-modality” (Fig. 1 Vertex 2.2) based approaches.

2.2.1 Single Modality

Single modality based methods update the biometric system without using any external help or knowledge [2-5, 7]. They can be based on “self-training” (Fig.1 Vertex 2.1.1) or graph-mincut (Fig.1 Vertex 2.1.2). These methods can be either online [6-8] or offline [9-13, 15-16], depending on whether the templates are updated as soon as the input data arrives or after a batch of data has been collected over a specific time-period.

a) Self-Updating “Self-training” based methods

These methods to template update, update themselves iteratively (Fig.1 Vertex 2.1.1) [6-9]. Specifically, these methods iteratively add “highly genuine” unlabelled data to the template set of the respective client. Highly genuine data is recognized if matching score between input data and the nearest template in the gallery, is over a fixed “updating threshold”. In online updating (Fig.1 Vertex 2.1.1.1), templates are updated as soon as the input data is available [6-8]. Some online approaches [6-7] also proposed an appropriate method for managing templates into the gallery (see Section 3). Offline methods (Fig.1 Vertex 2.1.1.2) [9-13, 15] to template update operate on the batch of unlabelled data (labels assigned during the system operations are stripped off). It is worth mentioning that both of these types of method are dependent on the threshold value selected for updating, as only the input data whose matching score is above the selected threshold is added for updating. They 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 it has been shown that these methods can exploit only the patterns very similar (“near”) to the enrolled templates which leads to non-exploitation of many difficult and informative intra-class variations but relaxing the threshold results in more probability of impostor’s introduction [11]. They are also dependent on the representativeness of initial enrolled templates. Online methods are also dependent on the order of the sequence of input data as it affects the matching score [7]. Ref. [21] have compared the performance of offline methods with online methods and no significant differences were reported in the performance.

b) Graph-based methods

Graph based methods [12] (Fig.1 Vertex 2.1.2) try to overcome the limitation of self update, which is only able to exploit data very near to initial templates, by organizing labeled and unlabelled data in a graph-like structure, where nodes are all the samples and edges are labelled with similarity values among them. The aim is assign labels to input data by studying the underlying structure between the labeled and unlabelled data.

Among the various graph based methods in semi-supervised learning [18], graph-mincut optimization has been proposed in [12]. The aim of graph mincut is to find, minimum weight set of edges, if cut, partitions the graph in two parts related to two opposite classes, the unlabelled data belonging to genuine class partition are labelled positive [20]. This method has the advantage of bypassing the selection of the optimum threshold, the method is not iterative hence not time consuming. Preliminary

experiments on comparison of graph-mincut with self update states that the former results in better assignment of labels and also reduces the creep in of impostors. On the other hand, these methods can only be offline (Fig. 1 Vertex 2.1.1.2).

2.2.2 Multi-modality

These methods utilize multimodal biometrics for updating (Fig.1 Vertex 2.2) [10,11,15]. They are based on the “co-training” concept of semi-supervised learning technique [19]. Specifically, two classifiers help each other in the updating process on the assumption that they work on i.i.d independently and identically distributed data [10-11]. The main intuition is that, if matchers are complementary, one matcher operating at high confidence helps other to identify “difficult patterns” [10-11, 15].

The offline version of this method (Fig.1 section 2.1.1.2) has been proposed in [10] with preliminary experiments. These showed that co-update is better than self-update in capturing large intra-class variations of the input data without relaxing acceptance threshold. Another work [11] investigated the conditions under which the co-update can update the system better than self-update specifically. Reported results pointed out that co-update can outperform self-update when the initial enrolled templates are non-representative of the input data as can happen in an uncontrolled environment. However, in a controlled environment when the input data is representative, performances of both self-update and co-update are comparable [11]. This work also presented for the first time in literature the effects of impostor’s introduction and studied the performance degradation of self-update due to this problem [11].

3 Template Management Strategies

In this section, it is discussed the strategy of managing selected input samples in the template gallery. An appropriate strategy is necessary since, in real applications, all possible collected intra-class variations are not virtually retainable, due to the constraints of the available memory size or verification time required, which obviously increases with the number of templates in the gallery. With regard to this problem, the procedure of updating of template galleries so far are: (1) super template based, in which the input data is always fused to a common single template called “super template” embedding all the information together [6-8], or (2) instance based, in which the input data is always added as a separate instance to the gallery set of the respective client [9-13, 15]. Most of the self-update online methods (Fig.1 Vertex 2.1.1.1) have followed super template based approach. However, the offline based self-update (Fig.1 Vertex 2.1.1.2), template co-update (Fig.1 Vertex 2.2) and graph-based methods (Fig.1 Vertex 2.1.2) have followed instance based approach. Recently, an intermediate approach has been followed too by always fusing the data which is similar to enrolled templates and adding intra-class variations as a separate instance to the template gallery set [11].

The above mentioned approaches, does not take into account that, in real applications, the available memory space is quite limited. Thus, due to space and time constraints may be some fixed number, for example, only M templates (with M small) could be stored. In order to take advantage of benefits of template update in constraint environment, template replacement criteria [5] have been proposed. They

treat templates as as a set of samples where obsolete templates can be “replaced” by novel and more representative ones. Accordingly, approaches like FIFO, LFU, and clustering algorithms, used for different contexts [2], have been modified [5] in order to be employed for template replacement. This approach has been so far tested online, and has shown the property of strongly decreasing misclassification errors during updating, by limiting the problem of impostors insertion in client’s galleries [5].

4 Experimental Protocols and Performance Evaluation

The effectiveness of a template update method is dependent on the operational factors such as the number of verification attempts, the prior probability of an impostor attempt and on the availability of intra-class variation over the operational timescale, thus the experimental protocol adopted for the evaluation of these methods are worthy to be examined.

An experimental protocol defines the number of initial images to be used for training the system, how is the threshold set, is it re-estimated or kept fixed, how many samples are used for updating the templates, what is the stopping criteria of these update methods, how is the performance enhancement evaluated for each method. The evaluation of these methods are also affected by the properties of the dataset in respect of the number of samples per client, more the better, and the period of the collection span of the database to account for temporal intra-class variations too.

Different protocols have been adopted in the literature of template update methods. Also a separate paper proposing flow chart of template-update methods and a protocol for performance evaluation has also been introduced in the literature [16]. The aim of this protocol is to evaluate the performance of the update procedure ir-respective of the order of sequence of input unlabelled data arrival. Accordingly, three ordered sequence of unlabelled data are used for updating the templates i.e.,(i) impostor matching first, in which the impostor unlabelled samples are followed by genuine ones, (ii) random matching, random presentation of genuine or impostor samples, (iii) genuine matching first, in which genuine samples are followed by impostor ones. The performance is evaluated accordingly for each sequence individually and is averaged, giving overall enhancement. Some self-update online methods (Fig.1 Vertex 2.1.1.1.) [7] have followed this protocol. Other self-update methods have only operated only at random matching scenario [6, 8]. All the self-update methods have operated at fixed threshold which is not re-estimated after each update procedure by an unlabelled input data.

However, most of the template self-update offline methods [9] (Fig.1 Vertex 2.1.1.2) and multi-modality based co-update methods [10-11, 15] (Fig.1 Vertex 2.2) have followed the protocol of training using an initial template and have operated at threshold fixed at 0 or 1% FAR and is re-estimated after each iteration of updating [7-10, 12] [10-11, 15]. The performance of online update procedures are reported to progress as the input data are used for updating without evaluation on a separate test set [6-8]. However, offline methods have used separate test set [9, 10-11, 15] for the performance evaluation due to updated templates. The graph based methods [12] (Fig.1 section 2.1.2) have also followed the protocol of offline methods but the difference is in the technique, as these methods assign labels to all the unlabelled data together, the performance on the test set has been evaluated after all the unlabelled samples being labeled and the template gallery updated, instead of progressive

Table 1. Description of the dataset used, their sizes, number of unlabelled data and if separate test set used, the size of it is mentioned for fair evaluation and understanding. Multiple entries denote multiple datasets used and accordingly multiple information are mentioned.

Ref.	Dataset & Traits used(dataset, trait)	Size of the database	#Unlabeled samples	% Performance enhancement	Separate Test set
[2] *	Home-made, finger	50 x 100	25	9.99% 31.39%	75
[4]	Equinox, face	50 x 100	supervised	97.6 %	50
[5]	FVC2002 Db1-4 , finger	100 x 8	6	53%	50
[6]	FVC2000 DB1-4, finger	100 x 8	6	31.39 %	No
	Home-made, finger	12 x 200	199	50	
[7]	Home-made, finger	41	10	32.81%	No
[8]	Home-made, face	20 x 220	9	75 %	210
		30 x 75	5	80%	70
		7 x 24	3	60%	21
[9]	AR, face	100 x 8	6	74 %	7
[10]	AR/FVC-2002 DB2, face & fingerprint	100 x 8	7	46%	No
[11]	Equinox and D1EE, face & fingerprint	42 x 20	9	59 %	11
[12]	Equinox, face	57x129	50	40 % - 27%	71

performance evaluation. A separate study in template update has evaluated the impact of template self-update at different threshold conditions and security level [13] and reported results indicated more performance enhancement at user-specific threshold with re-estimation at security level of 1% FAR [13].

As mentioned above, efficiency of an update method is also a function of the properties of dataset employed for its experimental evaluation. Table 1 gives the database description, the size of the database (number of people x samples per person), number of unlabelled data (samples used for template updating), the % performance enhancement reported and test set size, consisting of genuine and impostors samples (to evaluate the performance of the updated system), if separate test set is used for performance evaluation (Table 1 reports information only for the references proposing different update methods). Note that mostly the systems are trained with single initial enrolled template before the application of update procedures. Worth mentioning, stop criteria too, plays an important role in offline methods as it effects the number of unlabelled data used for the template updating thus effecting the gained performance. At present, some methods have operated on fixed number of unlabelled data used for updating [7][9] and some self update offline [13] (Fig.1 Vertex 2.1.1.2) and co-update methods [10-11] (Fig.1 Vertex 2.2) are based on updating the templates till no more samples from unlabelled batch can be used for updating. As the number of samples used for updating effects the performance reached, making the stop criteria very important issue.

All these information together gives complete detail about the proposed template update methods in literature and results in fair comparison of them. It can be seen that on an average the performance enhancement is significant but the size of unlabelled data and fixed number of iterations for self-update offline methods [9] (Fig.1 Vertex 2.1.1.2) does not provide a fair evaluation of these methods. Thus a fair stop criteria is still needed to be formulated for these offline methods. For the graph-based methods (Fig.1 Vertex 2.1.2) [12] no stop criteria is required as they are based on partitioning the graph and assigning the labels.

Moreover so far only face and fingerprint traits have been studied for template updating using PCA or EBGM classifier for face and Minutiae/String based method for fingerprint [2-13, 15] respectively. May be template updating can be studied for other trait too. Since the protocol is different, the methods cannot be compared directly. In future, these methods can be evaluated on the common test bed for the better comparison of the proposed template update methods. * Table 1, Ref [2] denotes two performances improvement due to batch and augment updating.

5 Discussions and Conclusions

Template update is a very challenging topic. In this paper, we critically reviewed the main approaches proposed in the literature. To the best of our knowledge, this is the first survey about template update methods. We proposed a possible taxonomy for summarizing the current state-of-the-art, highlighting the advancements and limitations of the proposed methods.

Although template update methods have shown to be promising, some open issues still need to be addressed for their effective implementation. In particular, the existence, if any, of the tradeoff between performance enhancement and gallery size maintained due to updating. Worth mentioning, all the template update methods are prone to impostors introduction and the attraction of more samples of it which may gradually lead to “creep in” of identity, when the genuine person loses its identity. Very recently, the problem of impostor’s introduction has been stated in [11]. Ref. [13] has reported that even with operation of update procedures at stringent threshold condition, the introduction of impostor cannot be avoidable. As apart from factors like incorrect estimation of threshold or basic FAR of the system, these methods are very much prone to impostors introduction due to the presence of “very difficult” clients, “wolves and lambs” according to the Doddington’s zoo [13]. Wolves are the clients having the ability to imitate others irrespective of stringent threshold conditions and lambs are the clients vulnerable to impostors attack and the presence of these characteristic clients result in impostor introduction. To model the early stoppage of impostor introduction due to these client is still an open issue.

References

1. Ross, A., Nandakumar, K., Jain, A.K.: Handbook of Multi-biometrics. Springer, Heidelberg (2006)
2. Uludag, U., Ross, A., Jain, A.: Biometric template selection and update: a case study in fingerprints. *Pattern Recognition* 37(7), 1533–1542 (2004)
3. 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)

4. Freni, B., Marcialis, G.L., Roli, F.: Template Selection by Editing Algorithms: a case of Study in Face Recognition. 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. 745–754. Springer, Heidelberg (2008)
5. Freni, B., Marcialis, G.L., Roli, F.: Replacement algorithms for fingerprint template update. In: Campilho, A., Kamel, M.S. (eds.) ICIAR 2008. LNCS, vol. 5112, pp. 884–893. Springer, Heidelberg (2008)
6. Jiang, X., Ser, W.: Online Fingerprint Template Improvement. *IEEE Trans. PAMI* 24(8), 1121–1126 (2002)
7. Ryu, C., Hakil, K., Jain, A.: Template adaptation based fingerprint verification. In: Proc. of ICPR, Hong Kong, vol. 4, pp. 582–585 (2006)
8. Liu, X., Chen, T., Thornton, S.M.: Eigenspace updating for non-stationary process and its application to face recognition. *Pattern Recognition*, 1945–1959 (2003)
9. 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)
10. Roli, F., Didaci, L., Marcialis, G.L.: Template co-update in multimodal biometric systems. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 1194–1202. Springer, Heidelberg (2007)
11. Rattani, A., Marcialis, G.L., Roli, F.: Capturing large intra-class variations of biometric data by template co-updating. In: IEEE Workshop on Biometrics, Int. Conf. on Vision and Pattern Recognition CVPR 2008, Anchorage, Alaska, USA (2008)
12. Rattani, A., Marcialis, G.L., Roli, F.: Biometric template update using the graph mincut: a case study in face verification. In: 6th IEEE Biometric Symposium, Tampa, USA (2008)
13. 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)
14. Doddington, G., Liggett, W., Martin, A., Przybocki, M., Reynolds, D.: Sheeps, Goats, Lambs and Wolves: A Statistical Analysis of Speaker Performance in the NIST 1998 Speaker Recognition Evaluation. In: Proc. ICSLD, Sydney, Australia (1998)
15. Didaci, L., Marcialis, G.L., Roli, F.: A theoretical and experimental analysis of template co-update in biometric verification systems. 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. 745–754. Springer, Heidelberg (2008)
16. Ryu, C., Kim, H.: Fingerprint Verification Testing Scenarios for Multi-impresion Enrollment and Template Adaptation. In: Proc. of Biometric Symposium, pp. 19–21, 39–40 (2005)
17. Jain, A.K., Dubes, R.C.: Algorithms for clustering data. Prentice-Hall, Englewood Cliffs (1988)
18. Zhu, X.: Semi-supervised learning literature survey, Computer Sciences TR 1530 (2006)
19. Blum, A., Mitchell, T.: Combining labelled and unlabeled data with co-training. In: Proc. of the Workshop on Computational Learning Theory, pp. 92–100 (1998)
20. Blum, A., Chawla, S.: Learning from labeled and unlabeled data using graph mincuts. In: Int. Conf. on Machine Learning, pp. 19–26 (2001)
21. Freni, B., Marcialis, G.L., Roli, F.: Online and Offline Fingerprint Template Update Using Minutiae: An Experimental Comparison. In: Perales, F.J., Fisher, R.B. (eds.) AMDO 2008. LNCS, vol. 5098, pp. 441–448. Springer, Heidelberg (2008)