








Incremental Learning Techniques Within a Self-updating Approach for Face Verification in Video-Surveillance

Eric Lopez-Lopez¹ , Carlos V. Regueiro¹ , Xosé M. Pardo² ,
Annalisa Franco³ , and Alessandra Lumini³ 

¹ Computer Architecture Group, CITIC, Universidade da Coruña, A Coruña, Spain
{eric.lopez, carlos.vazquez.regueiro}@udc.es

² Centro de Investigación en Tecnoloxías Intelixentes (CiTIUS),
Universidade de Santiago de Compostela, Santiago de Compostela, Spain
xose.pardo@usc.es

³ DISI - Department of Computer Science and Engineering,
Università di Bologna, Bologna, Italy
{annalisa.franco, alessandra.lumini}@unibo.it

Abstract. Data labelling is still a crucial task which precedes the training of a face verification system. In contexts where training data are obtained online during operational stages, and/or the genuine identity changes over time, supervised approaches are less suitable.

This work proposes a face verification system capable of autonomously generating a robust model of a target identity (genuine) from a very limited amount of labelled data (one or a few video frames). A self-updating approach is used to wrap two well known incremental learning techniques, namely Incremental SVM and Online Sequential ELM.

The performance of both strategies are compared by measuring their ability to unsupervisedly improve the model of the genuine identity over time, as the system is queried by both genuine and impostor identities. Results confirm the feasibility and potential of the self-updating approach in a video-surveillance context.

Keywords: Face verification · Video-surveillance · Incremental learning · Self-updating

1 Introduction

The aim of face verification in video-surveillance (FViVS) is to determine whether the faces captured in a sequence of video frames belong to a target (genuine identity). In addition to the general difficulties found in face verification using still photos, such as pose variations, illumination conditions, or occlusions, video face verification also incorporates its own issues (e.g. motion blur, low-resolution).

Depending on the treatment received by the video data three different scenarios emerge [13]. First, in Video-to-Video face verification (V2V) a system is

queried using a sequence of video frames in order to find the same target identity in another part of the video [23]. Second, in Still-to-Video (S2V) a system is queried with a still photo in order to find the same identity in a video [1, 5]. And finally, in Video-to-Still (V2S) verification tasks the goal is to find a target identity in a set of still images using a video query [4, 8].

Offline learning is commonly used when a large number of labelled face images are available for training in advance, and offers a good coverage of the (stationary) target domain. However, due to the high computational complexity required for retraining, it is not adequate for purpose where the regularly update of the classifier is needed. Conversely, online learning presents an efficient alternative by updating the classifier knowledge upon the arrival of new data. When the availability of labels is scarce, or relevant visual changes are expected in the target’s appearance after the modelling, online approaches are advantageous. It also learns to remove patterns that become extraneous and redundant over time. Thus, online learning has two components namely incremental and decremental learning, though here, we will only address the first one.

Hybrid approaches begin with an offline supervised learning, and then enhance the model over time in a semi-supervised or unsupervised way [4, 22]. Online approaches build and update their models in a semi-supervised or unsupervised way [20]. In biometry, both approaches are commonly referred as template updating. Here, we use self-updating to refer to methods where the decision whether to update or not is also driven by themselves.

Deserving a special mention, deep learning techniques have boosted face verification in terms of performance. Notwithstanding, their requirements of huge amount of training data hinder the applicability to real scenarios. In FViVS (either V2V or S2V/V2S) this limitation is due to the difficulty in curating and annotating such large video datasets as needed. In order to circumvent these difficulties, solutions like fine-tuning [21], or the utilisation of pre-trained networks as feature extractors have been proposed [19]. Nevertheless a general solution to transfer face recognition is still far from being reached [18]. Similarly, despite its growing interest in the scientific community, the adaptation of deep learning techniques to semi or unsupervised scenarios remains something pendent [24].

This work proposes a FViVS system capable of autonomously generates a robust model of a target identity, when starting with a minimum template. This template is unsupervisedly improved over time, as new samples of the target and different identities (impostors) are presented. To achieve this behaviour, incremental learning methods were selected. The scenario where a controlling agent selects one video frame that contains the target face, and the system is able to create the complete model by itself, epitomises an illustrative case of use. The main contributions are:

1. The application of a self-updating strategy to FViVS.
2. A comparison between two classification approaches designed for incremental learning (Incremental SVM [15] and Online Sequential ELM [16]) within a self-updating framework.
3. A study of the relevance of the initial template in the self-updating strategy.

The rest of the paper is organised as follows: Sect. 2 presents common strategies used to develop an unsupervised face verification system. Section 2.1 formally defines the hypothesis and the elements of a self-updating system. Sections 4 and 5 describe the experimental setting and the performed experiments. Finally, Sect. 6 exposes the conclusions of this work.

2 Unsupervised Face Verification in Videos

Traditionally, the use of template updating methods have been focused in two similar but different tasks: (i) the adaptation of a previously trained model to mitigate the impact of changes in either environments or facial appearance [4, 7], and, (ii) the gradual improvement of a template when the amount of labelled data is low [25]. This work try to provide insights on the second challenging task.

The absence of labels entails the necessity for somehow inferring this information and solving the dilemma of updating or not (Sect. 2.1). In the literature this is usually referred as the *stability-plasticity* or the *exploitation-exploration* dilemmas [9, 11]. In the specific case of videos, the possibility of exploiting a time series of images will help in the task of this inference (Sect. 2.2).

2.1 Self-updating

Firstly proposed in the scope of natural language processing [25], this approach relies on the output of the model to infer the labels to perform the template updating. The updating will be performed whenever the target identity is verified [6, 8]. Consequently, once the initial model is created (using a quite limited amount samples), the labelling (i.e. supervision) requirements is null.

In contrast, the main concern is how to determine the adequate **threshold** of the confidence value assigned to each label. Each new sample, labelled as belonging to the genuine identity and which confidence value is above the threshold, is used to update the template. While a high confidence threshold avoids the template corruption by outliers, it also prevents the system from accepting new valuable samples that differ from the ones contained in the template. Conversely, lower confidence thresholds can ease the acceptance and the subsequent addition of diverse information at the risk of corrupting the model with false positives.

2.2 Temporal Coherence

Often remarked as one of the keys to the actual development of an unsupervised learning method [8, 20, 24], the idea behind temporal coherence is something quite intuitive for humans. For example, if one of colleagues puts on a wig and sunglasses in front of you, it is natural to assume that the identity of this person is still the same despite his drastic look change. In videos, this idea is exploited assuming that successive frames tend to contain very similar information [2].

In FViVS, the exploitation of temporal coherence is performed with the help of a face tracker. This way, we can assume that an output video sequence of a face tracker belongs to the same identity despite changes in pose or illumination that could potentially damage the performance of a frame by frame recogniser.

3 Self-updating for FViVS

The idea of self-updating methods is to rely on the current model (M^t) at time t , to make the decision about updating itself when a video query has been identified (at time t) as belonging to the same identity. This way, unlabelled samples are gathered over time in order to improve the model without supervision.

Taking this into account the considered scenario assumes that initially ($t = 0$) a controlling agent selects one or a few video frames of the target identity (genuine) from a sequence given by a face tracker to create the *template*. It is also assumed the availability of a bunch of negative samples (impostors) from the domain of operation (in the literature this set is often called Universal Model, UM [4]) necessary to compare against the genuine information we are retrieving. The set of both the genuine template and the UM compose the set D^0 .

Over time ($t = 1, 2, 3, \dots$), the system is queried with new video sequences from unknown identities (both genuine and impostor) from the Cohort Model, CM [4]. If the model M^t accepts the query sequence, the sequence will be added to D^t in order to generate D^{t+1} and create the model M^{t+1} . In the opposite case, D^{t+1} remains the same so as M^{t+1} . The hypothesis of self-updating systems consist on assuming that this procedure will allow to improve performance.

3.1 Decision Rules for Self-updating

Since using a self-updating strategy gives to the models the power of deciding the label of a video sequence, we need to define three different rules:

- The *Frame Decision Rule (FDR)*. This rule assigns a score to every frame of the query video sequence. It corresponds to the outcome provided by the selected model (Sect. 3.2) and, consequently, dependant on it.
- The *Sequence Decision Rule (SDR)*. This rule is the actual implementation of the exploit of the temporal coherence described in Sect. 2.2. It is assumed that even if some frames of the sequence are not recognised we could still use the fact that the whole sequence belongs to a same identity in order to reject or accept it.

In practical terms, this rule assigns an unique score to every query video sequence based on the individual scores given by the FDR to each frame of their frames. It is computed using the **median** of the scores assigned by the FDR to each frame of the sequence (which is the equivalent of a majority voting). Identities will be verified by fixing a **threshold**. The cautiousness or greediness in this fixation is directly related with the *stability-plasticity* dilemma.

- The *Update Rule (UR)*. This rule marks how and when the model will be updated. In our case, whenever the identity is verified. Since it is planned to use only incremental methods, the update will consist in perform a partial fit using the actual query sequence.

3.2 Selected Incremental Learning Methods

Any classification method can be used within a self-update strategy. A self-updating method is used as a ‘wrapper’ one that in practice converts a supervised classification method into an unsupervised one.

In this work we compare two different incremental learning techniques within this strategy. The advantage of the incremental methods is that they provide a natural way of performing the template updating:

- **Incremental Support Vector Machines (I-SVM)**. [15] Solves the widely known problem of the Linear Support Vector Machines [3] by using the Stochastic Gradient Descent approach to incrementally find the hyper-plane parameters of the solution:

$$\mathbf{w}_0 \cdot \mathbf{x} + b_0 = 0$$

where \mathbf{w}_0 and b_0 are the parameters of the hyper-plane and \mathbf{x} represents a vector in the feature space.

- **Online Sequential Extreme Learning Machine (OS-ELM)**. [16] Derived by the well known ELM neural network classifier [12], this approach is specifically adapted to be able to compute and update the weight values sequentially as more data is becoming available (‘chunk-by-chunk’ or one-by-one).

In our case, a sigmoid function is used as activation function and the number of hidden nodes is empirically fixed at $\tilde{N} = 80$.

4 Experimental Setting

In this section, the experimental setup is explained. First, the database and the face detection algorithms are described. Then, the protocol for testing and the metrics used are presented.

4.1 FACE COX Database

The FACE COX database [13] gathers video frames of a total of 1000 users. There are 3 video sequences captured by 3 different cameras (**cam1**, **cam2** and **cam3**) and a high quality still photo of each user. The faces of the subjects, who walked along a S-path, were captured on fixed cameras with varying pose, illumination, scale, and amount of blur. Each camera recorded a part of the path, without temporal overlapping between them.

As it has been explained in Sect. 3, in a self-update strategy the update is performed after each video query. In this dataset, the number of sequences of a same user is quite limited (3 sequences per user). Therefore, a priori, this dataset would allow a maximum of 3 updates (without taking into account the testing needs). In order to mitigate this limitation, each video sequence was divided in a number of sub-sequences, while being ware of their temporal order.

Table 1. FACE COX database user and camera division learning.

	Genuine				Impostor			
	still	cam1	cam2	cam3	still	cam1	cam2	cam3
Train	0	0	0	0	300	300	300	300
Gallery	0	700	700	0	0	0	0	0
Probe	0	0	0	700	0	0	0	700

4.2 Face Detection and Feature Extractor

A face detection over each frame of the sequence is performed in order to isolate and correctly align the region of the face from the rest of the background using the tool provided in the Dlib library [14]. After that, we use the power of the pre-trained ResNet Convolutional Neural Network [10] implementation provided by the Dlib library [14] for feature extraction. This implementation achieves an accuracy of 99.38% in the LFW dataset and has shown to have very good properties in terms of robustness to non-identity related variations [17].

4.3 Training and Testing Protocol

Inspired by the protocol proposed by FACE COX database, we have created different subsets (Table 1):

- The **train subset** contains face images used as negative samples to train each method. In our experiments this subset is conformed by the images of 300 users taken from each 3 cameras.
- The **gallery subset** that contains the images used to create the initial template as well as the ones used to simulate the video queries (both genuine and impostor). To build this set in our experiments we will use the images from the other 700 users taken from **cam1** and **cam2**. Each video sequence taken from each camera were divided in **5** different sub-sequences given a total of 10 possible queries.
- The **probe subset** contains the images used to perform the testing of the system. The testing will be performed after each query of the learning phase to follow the evolution of the model. To build this subset we will use images taken from **cam3** from the same 700 users used to build the *gallery subset*. In this case we have divided each user sequence into **10** sub-sequences.

The identities that are present in the *train subset* will not be present in the other two subsets. This way, the identities of the training subset will conform the Universal Model (UM) and the identities in the *gallery* and the *probe subset* will make up the Cohort Model (CM). In the experiments, each identity will have a specific CM that will contain its data and the data of the 10 ‘most similar’ (using a SVM metric [17]) impostors.

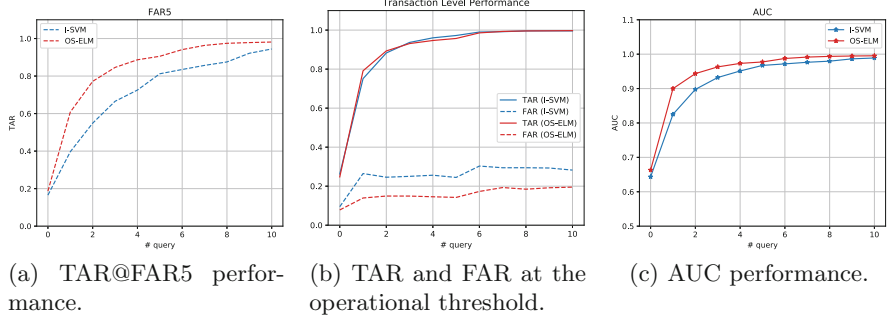


Fig. 1. Supervised performance of I-SVM and OS-ELM. Performance is measured after each query (in this case only genuine ones) presented to the system.

It is important to note that the *train subset* will be used as the validation set that will help us to fix the decision threshold of the SDR (see Sect. 3.1). The value of this *operational threshold* will be set to 10% FAR of the model created using the initial template.

4.4 Metrics

The metrics used to evaluate our system were the Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC) generated by the True Acceptance Rate (TAR) and False Acceptance Rate (FAR) when we vary the decision threshold. Besides, measurements of TAR at FAR of 5% (TAR@FAR5) and Transaction Level performance are also provided [4].

Performance is measured using using the *probe subset*, with a distribution of 10 sub-sequences per genuine identity and 1 sub-sequence per impostor identity. Then, the results obtained for each identity are averaged between the 700 identities from the *probe subset*.

5 Experiments and Results

The high degrees of freedom of the self-updating strategy forced us to be cautious during the testing. The first step we have taken is to establish a baseline or ‘upper-limit’ in the achievable performance. That is the case where the system is updated in a supervised manner (Sect. 5.1).

Afterwards, we have moved to measure the performance evolution by actually using the self-update strategy in unsupervised conditions (Sect. 5.2). Both genuine and impostor sequences are presented. The system needs to distinguish between them and update or not consequently. Finally, we highlight the importance of a good initial template for achieving good final performance (Sect. 5.3).

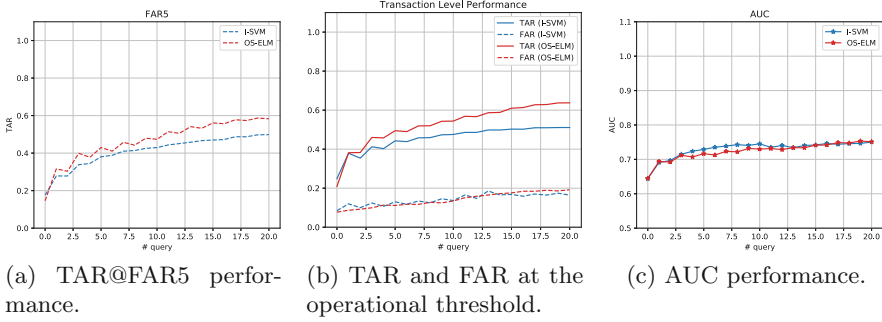


Fig. 2. Unsupervised performance of I-SVM and OS-ELM using a 1 frame initial template. Performance is measured after each query presented to the system.

5.1 Supervised Learning (Baseline)

The supervised case represents the upper-limit since labels query labels are provided to the system. Consequently, it does not have to decide between accepting or not. The initial template is composed by just one frame, the first one of the sequence which is used to create M^0 . From this point, the system is queried with 10 different queries from the genuine identity.

Performance is measured on the *probe subset* using the members of the CM of each identity, at the initial step ($t = 0$) and after each query ($t = 1, 2, \dots, 10$). This means that the CM is conformed by the genuine identity and the 10 most similar impostors (see Sect. 4.3). This has been done in order to have comparable results of this experiment with the following made under unsupervised conditions.

As it can be seen in Fig. 1a and c, both methods are able to achieve quite high performance (+0.90 TAR@FAR5 and +0.95 AUC) showing an overall good supervised modelling. However, it is important to note that the OS-ELM method shows the best behaviour in two important aspects. First, Fig. 1a shows a quicker improvement in performance, proving that this method is able to build a more robust model with the same data. This effect is specially visible during the first steps, when the genuine information is more limited.

Second, when the performance is measured for a given operational threshold (Fig. 1b), OS-ELM shows a more steady FAR over time than I-SVM. This can be specially relevant in the unsupervised learning due to the fact that an increasing FAR means that the probability of accepting impostors during the training will increase as well, and thus the risk of corrupting the model.

5.2 Unsupervised Learning (1 Frame Template)

Here we start testing the unsupervised capabilities of the two methods. The philosophy of the experiment is similar to the former one, requiring now the use of the SDR to distinguish between genuine and impostor identities. Thus, after the generation of M^0 using the initial template (1 frame), the system is queried

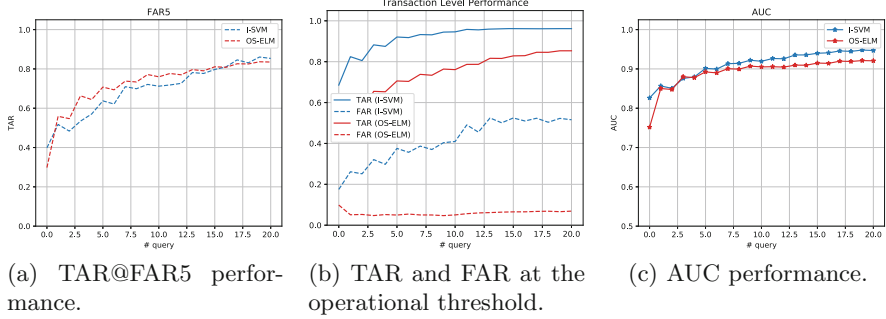


Fig. 3. Unsupervised performance of I-SVM and OS-ELM using a 5 frames initial template. Performance is measured after each query presented to the system.

by 10 genuine sequences and 10 impostor sequences, both of them belonging to the *gallery subset* (identities of the CM). For each genuine query (odd query, $t = 1, 3, \dots, 19$) we will have an impostor query (even query, $t = 2, 4, \dots, 20$) afterwards. Each impostor query belongs to a different identity.

It can be noted in Fig. 2a and c an important drop in performance of both methods with respect to the supervised cases. An explanation can be found on the fact that initial performance (≈ 0.2 TAR) is too poor to make a reliable decision, as Fig. 2b seems to proof. While FAR at the operational threshold remains mostly the same, TAR drops compared with the supervised case. Nevertheless, the self-updating strategy stills manages to achieve an important improvement specially in TAR@FAR5, $+0.32$ in I-SVM and $+0.39$ in OS-ELM (see Table 2). Overall, OS-ELM shows a slightly better performance respect I-SVM in every performance measurement.

One explanation for the moderate performance showed by both models in this experiment, could be found on the high requirements that were demanded. Specially because the template which was built with just one video frame. It could be affirmed that such a poor initial performance does not allow a system to start accepting/rejecting the right information. In the next section, we will repeat the same experiment with the difference that the template is built with 5 frames instead of just one.

5.3 Unsupervised Learning (5 Frame Template)

This experiment is the same that the previous one but changing the initial template from 1 frame to the 5 first frames As it can be seen in Fig. 3a and c, performance is significantly improved respect to the one-frame template. In both cases, TAR@FAR5 reaches values of 0.84.

Nevertheless, despite having a pretty similar performance, Fig. 3b shows an important difference between both methods. While OS-ELM maintains a steady FAR during all the experiment, I-SVM increases it over time. This could possibly means that I-SVM has a more unstable performance for a given threshold during

Table 2. Summary of TAR@FAR5 performances values obtained.

Model	Template	Initial	Final Superv.		Final Unsuperv.	
			Value	Improv.	Value	Improv.
I-SVM	1 frame	0.170	0.926	+0.756	0.498	+0.328
	5 frames	0.415	0.946	+0.531	0.846	+0.431
OS-ELM	1 frame	0.187	0.981	+0.794	0.583	+0.396
	5 frames	0.297	0.983	+0.686	0.848	+0.551

the online training. Nevertheless, since every impostor query has a different identity, this malfunctioning is not reflected too much in TAR@FAR5. Unlike the accepted genuine sequences, the impact of the accepted impostors is not constructive. This may have greater impact in the case where a same impostor is repeatedly querying the system.

5.4 Discussion

In Table 2, a summary of the experiments conducted in this work is presented. We have added the case of an initial template of 5 frames in supervised conditions in order to see the complete picture. Overall it can be said that the self-updating strategy is able to improve performance in every experiment. It is important to remark the extremely low labelled conditions (1 or 5 low quality video frames) in which our experiments were conducted have not been able to avoid this improvement. On the other hand, our experiments show that the self-updating strategy is quite sensible to the initial performance of the model (which in our case is expressed in the necessity of more genuine data to create the initial template). This fact makes our systems move from about 0.50 to 0.85 TAR@FAR5.

One final appointment to mention is how the systems are affected by our decisions when defining the self-updating strategy. In this case, for the sake of simplicity, a fixed threshold in the SDR was established in order to decide whether to update or not. This threshold was fixed selecting the point of M^0 ROC curve that corresponds to a 10% FAR. Nevertheless, we cannot assure that this ROC point will be stable with time. Therefore, the fixed threshold benefits the classification methods that preserve (or even decrease) the ROC point of functioning (as it can be seen in Fig. 3b where, unlike OS-ELM, I-SVM's FAR constantly increases).

6 Conclusions

In this work, the unsupervised FViVS problem using a self-update strategy has been explored. The case of study starts with a surveillance agent selecting one frame from a video sequence, and then the autonomous video-surveillance system

try to detect the same identity within the same or a different video sequence, while incrementally build a robust model of the target identity.

Experiments showed that a self-updating strategy seems to be viable to build the identity model without the necessity of labels, or at least capable of improving initial performance. In addition, the importance of a correct decision rule is highlighted during the online training as well as its correlation with the classification method at hand. This fact makes OS-ELM performance stands out respect to I-SVM.

For future work, the aim is to perform a deeper study including more classification techniques and an extended experimental assessment. It would also be interesting to explore the behaviour of this approach in life-long learning conditions in order to study its robustness to unwanted drifts.

Acknowledgements. This work has received financial support from the Spanish government (project TIN2017-90135-R MINECO (FEDER)), from The Consellería de Cultura, Educación e Ordenación Universitaria (accreditations 2016–2019, EDG431G/01 and ED431G/08), and reference competitive groups (2017–2020 ED431C 2017/69, and ED431C 2017/04), and from the European Regional Development Fund (ERDF). Eric López had received financial support from the Xunta de Galicia and the European Union (European Social Fund - ESF).

References

1. Bashbaghi, S., Granger, E., Sabourin, R., Bilodeau, G.A.: Dynamic ensembles of exemplar-SVMs for still-to-video face recognition. *Pattern Recogn.* **69**, 61–81 (2017). <https://doi.org/10.1016/j.patcog.2017.04.014>
2. Becker, S.: Implicit learning in 3D object recognition: the importance of temporal context. *Neural Comput.* **11**(2), 347–374 (1999). <https://doi.org/10.1162/089976699300016683>
3. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995). <https://doi.org/10.1007/BF00994018>
4. De la Torre, M., Granger, E., Radtke, P.V., Sabourin, R., Gorodnichy, D.O.: Partially-supervised learning from facial trajectories for face recognition in video surveillance. *Inf. Fusion* **24**, 31–53 (2015). <https://doi.org/10.1016/j.inffus.2014.05.006>
5. Dewan, M.A.A., Granger, E., Marcialis, G.L., Sabourin, R., Roli, F.: Adaptive appearance model tracking for still-to-video face recognition. *Pattern Recogn.* **49**, 129–151 (2016). <https://doi.org/10.1016/j.patcog.2015.08.002>
6. Didaci, L., Marcialis, G.L., Roli, F.: Analysis of unsupervised template update in biometric recognition systems. *Pattern Recogn. Lett.* **37**, 151–160 (2014). <https://doi.org/10.1016/j.patrec.2013.05.021>
7. Ditzler, G., Roveri, M., Alippi, C., Polikar, R.: Learning in nonstationary environments: a survey. *IEEE Comput. Intell. Mag.* **10**(4), 12–25 (2015). <https://doi.org/10.1109/MCI.2015.2471196>
8. Franco, A., Maio, D., Maltoni, D.: Incremental template updating for face recognition in home environments. *Pattern Recogn.* **43**(8), 2891–2903 (2010). <https://doi.org/10.1016/j.patcog.2010.02.017>

9. Grossberg, S.: Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Netw.* **1**(1), 17–61 (1988). [https://doi.org/10.1016/0893-6080\(88\)90021-4](https://doi.org/10.1016/0893-6080(88)90021-4)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778 (2016). <https://doi.org/10.1109/CVPR.2016.90>
11. Hoens, T.R., Polikar, R., Chawla, N.V.: Learning from streaming data with concept drift and imbalance: an overview. *Prog. Artif. Intell.* **1**, 89–101 (2012). <https://doi.org/10.1007/s13748-011-0008-0>
12. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: theory and applications. *Neurocomputing* **70**(1), 489–501 (2006). <https://doi.org/10.1016/j.neucom.2005.12.126>
13. Huang, Z., et al.: A benchmark and comparative study of video-based face recognition on cox face database. *IEEE Trans. Image Process.* **24**(12), 5967–5981 (2015). <https://doi.org/10.1109/TIP.2015.2493448>
14. King, D.E.: Dlib-ml: a machine learning toolkit. *J. Mach. Learn. Res.* **10**, 1755–1758 (2009). <https://doi.org/10.1145/1577069.1755843>
15. Kivinen, J., Smola, A.J., Williamson, R.C.: Online learning with kernels. *IEEE Trans. Signal Process.* **52**(8), 2165–2176 (2004). <https://doi.org/10.1109/TSP.2004.830991>
16. Liang, N., Huang, G., Saratchandran, P., Sundararajan, N.: A fast and accurate online sequential learning algorithm for feedforward networks. *IEEE Trans. Neural Netw.* **17**(6), 1411–1423 (2006). <https://doi.org/10.1109/TNN.2006.880583>
17. López-López, E., Pardo, X.M., Regueiro, C.V., Iglesias, R., Casado, F.E.: Dataset bias exposed in face verification. *IET Biom.* **8**(4), 249–258 (2019). <https://doi.org/10.1049/iet-bmt.2018.5224>
18. Masi, I., Wu, Y., Hassner, T., Natarajan, P.: Deep face recognition: A survey. In: *Conference on Graphics, Patterns and Images (SIBGRAPI)*, pp. 471–478 (2018). <https://doi.org/10.1109/SIBGRAPI.2018.00067>
19. Pernici, F., Bartoli, F., Bruni, M., Del Bimbo, A.: Memory based online learning of deep representations from video streams. In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 2324–2334 (2018). <https://doi.org/10.1109/CVPR.2018.00247>
20. Pernici, F., Bimbo, A.D.: Unsupervised incremental learning of deep descriptors from video streams. In: *International Conference on Multimedia Expo Workshops (ICMEW)*, pp. 477–482 (2017). <https://doi.org/10.1109/ICMEW.2017.8026276>
21. Sohn, K., Liu, S., Zhong, G., Yu, X., Yang, M., Chandraker, M.: Unsupervised domain adaptation for face recognition in unlabeled videos. In: *International Conference on Computer Vision (ICCV)*, pp. 5917–5925 (2017). <https://doi.org/10.1109/ICCV.2017.630>
22. Villamizar, M., Sanfeliu, A., Moreno-Noguer, F.: Online learning and detection of faces with low human supervision. *Vis. Comput.* **35**(3), 349–370 (2019). <https://doi.org/10.1007/s00371-018-01617-y>
23. Wang, R., Shan, S., Chen, X., Gao, W.: Manifold-manifold distance with application to face recognition based on image set. In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8 (2008). <https://doi.org/10.1109/CVPR.2008.4587719>

24. Wang, X., Gupta, A.: Unsupervised learning of visual representations using videos. In: International Conference on Computer Vision (ICCV), pp. 2794–2802 (2015). <https://doi.org/10.1109/ICCV.2015.320>
25. Yarowsky, D.: Unsupervised word sense disambiguation rivaling supervised methods. In: Annual Meeting on Association for Computational Linguistics (ACL), pp. 189–196. Association for Computational Linguistics, Stroudsburg, PA, USA (1995). <https://doi.org/10.3115/981658.981684>