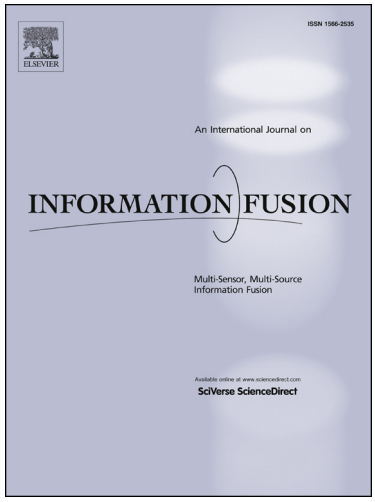


Partially-Supervised Learning from Facial Trajectories for Face Recognition in Video Surveillance

Miguel De-la-Torre, Eric Granger, Paulo V.W. Radtke, Robert Sabourin, Dmitry O. Gorodnichy

PII: S1566-2535(14)00070-0
DOI: http://dx.doi.org/10.1016/j.inffus.2014.05.006
Reference: INFFUS 654

To appear in: *Information Fusion*

Received Date: 20 August 2013
Revised Date: 17 May 2014
Accepted Date: 26 May 2014

Please cite this article as: M. De-la-Torre, E. Granger, P.V.W. Radtke, R. Sabourin, D.O. Gorodnichy, Partially-Supervised Learning from Facial Trajectories for Face Recognition in Video Surveillance, *Information Fusion* (2014), doi: http://dx.doi.org/10.1016/j.inffus.2014.05.006

Available online at www.sciencedirect.com

ScienceDirect

Information Fusion

Information Fusion, (XXXX) XX (2014) 1–38

Partially-Supervised Learning from Facial Trajectories for Face Recognition in Video Surveillance

Miguel De-la-Torre[a,b], Eric Granger[a], Paulo V. W. Radtke[a], Robert Sabourin[a], Dmitry O. Gorodnichy[c]

[a]*Laboratoire d'imagerie de vision et d'intelligence artificielle, École de technologie supérieure, Université du Québec, Montréal, Canada*
[b]*Centro Universitario de Los Valles, Universidad de Guadalajara, Ameca, México*
[c]*Science and Engineering Directorate, Canada Border Services Agency, Ottawa, Canada*

Abstract

Face recognition (FR) is employed in several video surveillance applications to determine if facial regions captured over a network of cameras correspond to a target individuals. To enroll target individuals, it is often costly or unfeasible to capture enough high quality reference facial samples a prior to design representative facial models. Furthermore, changes in capture conditions and physiology contribute to a growing divergence between these models and faces captured during operations. Adaptive biometrics seek to maintain a high level of performance by updating facial models over time using operational data. Adaptive multiple classifier systems (MCSs) have been successfully applied to video-to-video FR, where the face of each target individual is modeled using an ensemble of 2-class classifiers (trained using target vs. non-target samples). In this paper, a new adaptive MCS is proposed for partially-supervised learning of facial models over time based on facial trajectories. During operations, information from a face tracker and individual-specific ensembles is integrated for robust spatio-temporal recognition and for self-update of facial models. The tracker defines a facial trajectory for each individual that appears in a video, which leads to the recognition of a target individual if the positive predictions accumulated along a trajectory surpass a detection threshold for an ensemble. When the number of positive ensemble predictions surpasses a higher update threshold, then all target face samples from the trajectory are combined with non-target samples (selected from the cohort and universal models) to update the corresponding facial model. A learn-and-combine strategy is employed to avoid knowledge corruption during self-update of ensembles. In addition, a memory management strategy based on Kullback-Leibler divergence is proposed to rank and select the most relevant target and non-target reference samples to be stored in memory as the ensembles evolves. For proof-of-concept, a particular realisation of the proposed system was validated with videos from Face in Action dataset. Initially, trajectories captured from enrollment videos are used for supervised learning of ensembles, and then videos from various operational sessions are presented to the system for FR and self-update with high-confidence trajectories. At a transaction level, the proposed approach outperforms baseline systems that do not adapt to new trajectories, and provides comparable performance to ideal systems that adapt to all relevant target trajectories, through supervised learning. Subject-level analysis reveals the existence of individuals for which self-updating ensembles with unlabeled facial trajectories provides a considerable benefit. Trajectory-level analysis indicates that the proposed system allows for robust spatio-temporal video-to-video FR, and may therefore enhance security and situation analysis in video surveillance.

Keywords: Semi-Supervised Learning, Multiple Classifier Systems, Adaptive Biometrics, Incremental Learning, Video-to-Video Face Recognition, Video Surveillance.

1. Introduction

In video surveillance applications, automated face recognition (FR) systems are increasingly employed to match facial regions of interest (ROIs) captured across a network of video cameras to individuals of interest enrolled to the system. These applications range from watchlist screening, which involves still-to-video FR, to person re-identification (for search and retrieval), which involves video-to-video FR. Regardless, systems for FR in video surveillance (FRiVS) must operate under semi- and unconstrained capture conditions, where scale, pose, occlusion, blur/resolution, expression and illumination vary over time.

scores produced by ROIs in a trajectory [5]. Tracking information as also been used to model the joint posterior distribution of the motion and identity for the individual in the scene [6].

This paper concerns system for video-to-video FR, where facial models for matching are defined as a statistical model. Facial models are usually designed during enrollment, ideally using several high quality reference ROIs captured for the target individual under controlled conditions. In video-to-video FR, these reference ROIs are extracted along one or more reference trajectories. This requirement is rarely fulfilled in practical applications, and enrollment of individuals often relies on a limited number of lower quality ROIs. FR performance tends to decline since facial models are not representative of the faces to be recognized during operations. Both abrupt and gradual changes in capture conditions (due to, e.g., aging and variations in pose and lighting) also lead to a decline in FR performance due to a growing divergence between these facial models and faces captured during operations. Several adaptive classifiers have been proposed in literature for supervised incremental learning of labeled samples [2, 7, 8, 9]. These can be used to update facial models after enrollment, as new reference data becomes available, allowing to maintain or increase matching performance. Adaptive multiple classifier systems (MCS) have been successfully applied for FRiVS [2, 10]. In these systems, the facial model of each individual is encoded using an ensemble of 2-class classifiers or detectors (EoD), trained to discriminate between samples of a target individual and non-target individuals.

An issue with the supervised update of classifiers is the analysis and extraction of labeled reference samples from operational videos. A domain expert must isolate target faces manually or semi-automatically in video surveillance footage, which involves undesirable costs and delays. Instead of relying on a human expert, the system may self-update face models with operational videos. Several semi-supervised learning approaches have been proposed to update biometric models using a combination of labeled and unlabeled samples [11, 12, 13]. In the area of adaptive biometrics, two representative approaches for semi-supervised learning are the self-update and co-update techniques [14]. The first applies an update threshold (higher than the detection threshold) to each matching scores to select input biometric samples as new templates, and the second seeks corroboration of scores from two or more matchers for cross-updating.

To the authors' knowledge, a FR system that allows for self-updating facial models in video surveillance applications has not been proposed in literature. An issue encountered with self-updating is the reliable selection of operational samples from the target individual to adapt facial models. A high level of confidence is required to avoid updating models with non-target data. In contrast, a facial model should also be adapted with a diversified set of reference samples to improve the generalization performance. Given an adaptive MCS proposed in [2, 10], information from a face tracker and individual-specific ensembles may be integrated to provide a variety of high confidence reference samples.

In video surveillance, an abundance of reference samples may be extracted from non-target facial trajectories acquired in the scene during routine system operation. Two databases may be formed with samples extracted (1) from trajectories of other individuals of interest besides the target individual (known as the cohort model, CM), and (2) from unknown people appearing in scene (known as the universal model, UM) [1, 2, 10, 15]. This imposes the

need to sub-sample non-target data in order to design accurate facial models, using an ensemble of 2-class classifiers. Moreover, adaptive MCSs require reference data to be stored in memory for validation [2, 9]. Practical memory limitations impose the need for a method to rank and select the most relevant validation samples for each individual (EoD).

In this paper, an adaptive MCS is proposed for video-to-video FR in semi- and unconstrained video surveillance environments. Within the adaptive MCS, an EoD encodes and updates the facial model of each individual of interest. This novel system allows for spatio-temporal recognition and self-update of facial models based on high-confidence trajectories. During operations, a face tracker defines facial trajectories for different individuals that appear in a video. Track ID numbers are integrated with predictions of individual-specific ensembles at a decision-level for enhanced video-to-video FR. The proposed system relies on tracker quality to regroup ROIs into facial trajectories, and applies a double thresholding scheme to curves produced by accumulating positive EoD predictions for a trajectory. An individual of interest is recognized if the number of positive predictions accumulated over some time window of a trajectory surpass a *detection* threshold for an EoD.

A second (higher) *update* threshold is applied to select high-confidence trajectories that are suitable for self-updating a facial model. If the number of positive predictions surpasses this threshold for an EoD, then all samples extracted from the target ROIs of the trajectory are combined with non-target samples (selected from the CM and UM) to update the corresponding face model. Since a trajectory may contain target ROIs that were incorrectly classified by the EoD, facial models are adapted with a diversified set of reference samples that may refine the decision boundary between target and non-target distributions, and thereby improve the generalization performance. A sub-sampling technique based on condensed nearest neighbor (CNN) [16] is employed to select non-target samples along this boundary. The data for EoD update is comprised of diverse facial regions associated with target and non-target trajectories, and is employed to generate a new pool of 2-class classifiers, and to update the fusion function of the user specific EoD. To avoid issues related to knowledge corruption in incremental learning classification systems, the self-update of EoD employs a learn-and-combine strategy [2]. Finally, a long term memory (LTM) is maintained over time with a fixed number of reference validation samples per individual. A memory management strategy based on the Kullback-Leibler (KL) divergence criteria [17] is proposed to rank and select the most relevant target and non-target reference samples samples. This criteria seeks to preserve the highest relative entropy of ensemble over time. In other words, the KL divergence becomes higher for samples that contain a higher level of information according to the knowledge previously acquired by the individual specific EoD.

Video sequences from the Carnegie Mellon University Face in Action (FIA) dataset for video FR was used for proof-of-concept validation. Video sequences were captured from 180 subjects with an array of 6 cameras over three sessions separated by a three-month interval. In this dataset, video of individuals were captured under semi-controlled conditions in a security check point scenario. When a sequence is presented to the proposed system during operations, trajectories are employed for spatio-temporal recognition, and high-confidence trajectories are used for self-update. Three levels of performance evaluation are considered – transaction-based analysis (in the ROC and *precision-recall*

spaces), subject-level analysis (Doddington zoo characterization), and trajectory-based analysis (of the overall system for video sequences).

This paper is organized as follows. Sections 2 and 3 provide a brief overview of techniques employed for FRiVS and adaptive biometrics, respectively. The adaptive MCS proposed for self-update from facial trajectories is described in Section 4, including specialized individual-specific strategies for management of reference data, for fusion of tracking and classification responses, and for self-update of facial models (EoDs). Section 5 describes the experimental methodology – protocol, video data set and measures used in performance evaluation. Finally, results are presented and discussed in Section 6.

2. Video-to-video Face Recognition

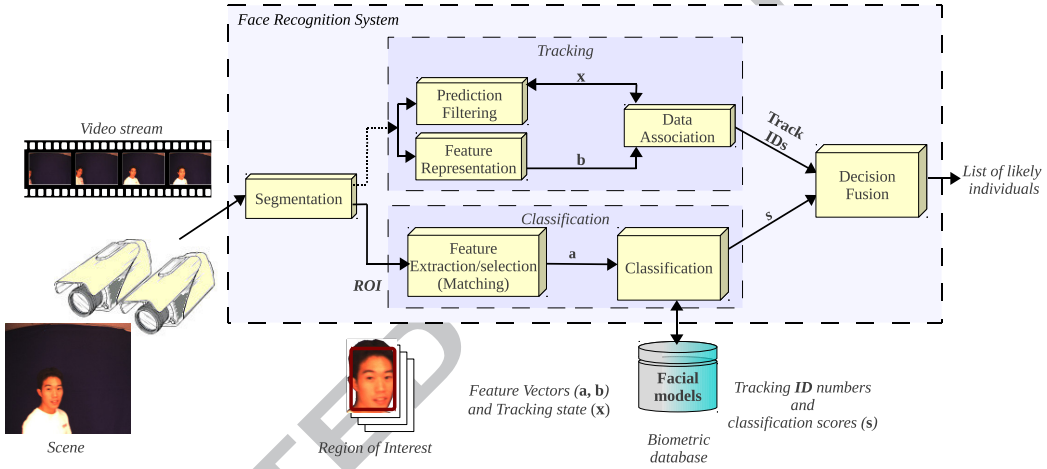


Figure 1. Block diagram of a system for video face recognition.

Assume that video streams are captured using one or more video cameras (see Fig. 1). The segmentation process isolates the facial regions of interest (ROIs) from successive frames, and discriminative features are extracted to represent faces for tracking (vector **b**) and classification (vector **a**). A new track is typically initialized when an emergent face is captured far from others, and is defined over consecutive frames using the state of the facial region being tracked **x** (appearance, scale, position, track number, etc.) and a vector of tracker-specific features **b**. Classification features extracted from each ROI (vector **a**) are often image-based (using e.g., Local Binary Patterns) or pattern recognition-based (using e.g., Principal Component Analysis). The tracking module follows the movement or expression of distinct faces across video frames, while the classification module matches ROIs captured in video to the system's facial models. Finally, the decision fusion combines track **ID**s and classification scores **s** in order to predict it target individuals appear before a camera.

2.1. Face Tracking:

Facial tracking (FT) techniques allow to follow the movement of each of individual and to regroup facial regions of a same person (without knowing his identity). The input of the tracker is the stream of frames acquired with video cameras, and the initial face ROIs to be tracked, while the output defines as a set of facial regions with the same **ID** for which the track has high tracking quality Q_T. Note that only the first ROI in a trajectory (ROIs used for classification) may be equivalent in a track (state of facial regions from the tracker) [18].

The basic tracking steps are face representation, prediction filtering and data association. In face representation, the tracked facial region is represented with distinctive features (tracking feature vector **b**) in order to allow tracking from one frame to the next. Commonly used features are color histogram, skin color probability map and active contours, just to mention a few. Predicting the next state with Kalman and Particle filters seeks the new state **x** (appearance, scale, location, and/or velocity, etc.) of the facial region to be tracked in the current frame, based on the information in the previous frames and some underlying model for state transitions. The objective of the prediction filtering is to avoid drift and reduce the search space by using a probability framework, although some methods perform data association heuristically instead (e.g. Mean-shift and Cam-shift). Finally, in the data association step, the tracker associates a feature vector of the facial region extracted from the previous frame with the feature vector in the current frame. Tracking methods are categorized according to the type of descriptor used for face representation: holistic, contour-based, and hybrid information. Most face-tracking methods in literature rely on holistic representations due to their robustness.

2.2. Specialized Classification Architectures:

In the literature, FR in video surveillance (FRiVS) is addressed as an open set problem, considering that the number of individuals of interest is highly outnumbered by other persons in the scene. Multi-class classifiers have been used, which apply a rejection threshold for unknown individuals. A multi-class classifier designed for video FR is the Open Set TCM-kNN [1]. It uses transductive inference to produce a classification score based on randomness deficiency. Tax and Duin also proposed a technique to combine one-class classifiers in a multi-class classifier. Their heuristic allows to adjust a class-specific outlier rejection threshold, and combine non-generative class models [19].

Similarly, modular architectures with one detector per individual have been proposed to address the problem with individual-specific 1- or 2-class classifiers. The convenience of these modular approaches has been widely studied in the literature, setting individual- (or user-) independent parameters [20]. For instance, the approach proposed by Kamgar and Parsi, that identifies the decision region(s) in the feature space for each individual face by training a dedicated feed-forward neural network for each individual of interest [21]. Another example is the SVM-based modular system proposed by Ekenel et al., applied to a visitor interface scenario [5].

Finally, modular approaches have been extended to train an ensemble of classifiers per individual. An example of such a system is the ensemble of detectors (EoD) designed for each person in a watch list. Non-target samples are retrieved from the CM (database maintained with trajectories from non-target individuals of interest) and the UM

(database with training samples from unknown people appearing in scene). Base classifiers are co-jointly trained using a training strategy based on DPSO. It allows for the generation of a diversified pool of ARTMAP neural networks, and trained detectors are then selected and combined using Boolean combination (BC) [10].

2.3. Decision Fusion:

Approaches for FR in video can be categorized according to those that neglect temporal information and those that propose strategies to exploit it. Algorithms that neglect temporal information have been proposed for still image recognition, and exploit only physiological information on the face. Examples of these approaches include Eigenfaces, Fisherfaces and Active Appearance Models. Alternatively, approaches that exploit temporal information present the advantage of increased contextual knowledge and data in video, allowing the use of physiological and behavioral information. Discriminant analysis of facial optical flow, Hidden Markov Models (HMMs), and the sequential importance sampling (SIS) algorithm are just some approaches in this category [3].

Spatio-temporal approaches for FR merge spatial information (e.g. face appearance) with the sequential variations presented over time (e.g behavior). Zhang and Martinez use probabilities accumulated by matching ROIs to the individual-specific Gaussian mean estimated from gallery reference samples, and normalize to produce posterior probabilities. This temporal analysis is independent of the matching or tracking algorithm [22]. Liu and Chen used HMMs to model the appearance and dynamics of a person, obtaining high confident results on sequences that were then used to adapt the models. A potential problem with the modeling of probability distributions of the motion is the assumption that the movement will be very similar, regardless of the new scenario [23]. Accumulating classification responses over time eliminates the assumption, and still takes into account the time information. For instance, the work of Ekenel et al. evaluates a video-to-video FR system for individuals entering into a room, which progressively combines confidence scores of the matchers using a sum rule over the full sequences to estimate the identity in video [5]. In their approach, they use a k-NN classifier on a DCT representation of face images, and use min-max normalization on the distance-based output scores, and then compare their proposed approaches: distance-to-model, distance-to-second-closest and a combination of both. *Score* and *quality* driven fusion methods were used to combine responses from frames in video sequences, within a border control system [4]. In the first method, matching scores are compared to a predetermined threshold, whereas the second compares the intrinsic quality of the image intrinsic to the predefined threshold. Finally, a joint sparse representation has been used to simultaneously take into account correlations and coupling information among video frames [24]. Sub-dictionaries for distinct partitions are aligned using majority voting, and decisions are made under the minimum class reconstruction error criterion.

2.4. Challenges of Facial Modeling:

One of the main challenges of FRiVS is that facial models lose their representativeness over time because they are designed *a priori* design using a limited number of reference samples captured under semi- and uncontrolled

conditions. Facial captures incorporate considerable variations because of the limited control over operational conditions in the scenes – changes in illumination, pose, facial expression, orientation, occlusion, etc. Furthermore, the physiology of enrolled individuals may change over time, either temporarily (e.g., hairstyle, cosmetics, glasses, etc.) or permanently (e.g., aging, surgery, etc). These factors result in facial models that are not representative of faces to be recognized. However, new information may emerge during operations to update of re-enrollment, and formerly collected data may eventually become obsolete in a changing environment. As described in Section 3, several adaptive biometric techniques have been proposed to update biometric models over time, and maintain or improve a high level of performance.

3. Adaptive Biometric Systems

The internal structure of biometric models dictates the most effective strategy for adaptation. In general, it involves (1) the *selection* of diversified, relevant reference samples to update a template gallery or an LTM of reference validation samples, and (2) the actual *update* of template galleries or classifier parameters using supervised or semi-supervised learning schemes.

3.1. Selection of Representative Samples:

In this paper, adaptive MCS are considered for FRiVS, where an ensemble of detectors (EoD with 2-class classifiers trained on target vs. non-target samples) is used to design the facial models of individuals of interest [2]. The level of informativeness of an input sample **a**, may be estimated using selection techniques based on the data itself, or using information retrieved from the ensemble. Examples of selection techniques used for FR include editing algorithms such as the CNN, used to manage a gallery of templates in template matching systems [25].

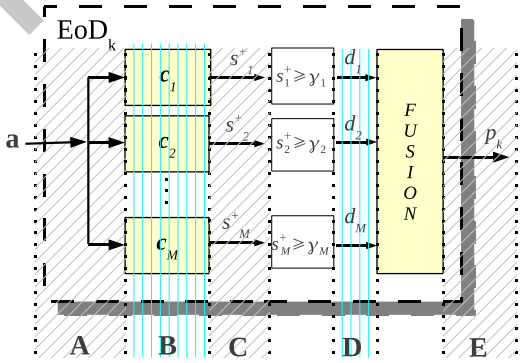


Figure 2. Ranking levels that are relevant for an ensemble of 1- or 2-class binary classifiers, e.g., for individual k.

Fig. 2 presents the levels of selection that are relevant for ensembles of 1- or 2-class binary classifiers. The *input data level* **(A)** allows to use the dataset itself to filter out redundant samples. At this level, the estimation of the data distribution of samples is not required in the filtering process, which makes the methods at level **(A)** dependent

only on the reference samples. Filtering methods here do not use a ranking, but rather, the geometric relationship between samples in feature space. At the *classifier level* **(B)**, the relevance measure of samples is retrieved from the internal response of the classifier to an input sample **a**. At the *classifier score level* **(C)**, the output scores $s_m^+(\mathbf{a})$ of the M classifiers in the ensemble are combined to produce a measure of relevance. When probabilistic classifiers are used as base classifiers, the relevance measure computation is based on the combined estimated posterior probability (classification scores s_m^+). At the *classifier decision level* **(D)**, the decisions $d_m(\mathbf{a})$ of the classifiers in the ensemble are combined. Voting strategies can be used to generate a relevance measure such as vote entropy. Finally, at the *ensemble decision level* **(E)**, the global output of the ensemble can be used as a measure of informativeness of the input sample.

Table 1. Sampling techniques for the selection of representative samples according to the five ranking levels from Fig. 2.

Technique	**A**	**B**	**C**	**D**	**E**
Uncertainty sampling (from Active Learning)					
Less confident [26]					✓
Surprise [27]					✓
Margin Sampling [28]		✓			✓
Entropy Sampling [29]					✓
Query by Committee (from Active Learning)					
Average surprise [27]			✓		
Average Margin Sampling [28]		✓	✓		
Vote Entropy [30]				✓	
Kullback-Leibler divergence [17]			✓		
Other measures inspired in diversity of ensembles					
Margin (voting) [31]				✓	
Less confident (voting) [26]				✓	
Surprise (voting) [27]				✓	
Resampling techniques [32, 33]					
Condensed Nearest Neighbor rule [16]	✓				
Random Undersampling	✓				
SPIDER	✓				
One-Sided Selection	✓				
Wilson's Edited Nearest Neighbor rule	✓				
Neighborhood Cleaning Rule	✓				
Tomek links [34]	✓				
Boosting weighting	✓				
Budget-sensitive, progressive-sampling	✓				

Table 1 presents sampling techniques from the literature according to the five ranking levels. Techniques that operate at level **A**, are suitable when the distribution of the new incoming data is unknown, e.g., before the samples are used in the design/update process. Using data dependent techniques to select reference samples avoids any bias produced by the knowledge already embedded in the system. At level **B**, information from the internal components of the classifiers are used to estimate the relevance of test samples. However, given that such information is incompatible from one classifier to another, such ranking techniques usually suffer from poor representativeness of the informativeness of a sample.

Levels **C** and **D** are independent of the classification algorithm used in the ensemble and in the combination

strategy. The only constraint imposed at level **C** lies in the compatibility of scores produced by classifiers, a limitation that can be overcome by using normalization strategies. Alternatively, probabilistic base classifiers can be used, taking advantage of their output estimated posterior probabilities, and avoiding the need for normalization. Level **D** is also a good candidate for combining decisions from (crisp) classifiers; however, the resolution is limited by the number of classifiers in the ensemble. Finally, level **E** estimates the informativeness of an input sample using information from ensemble members and the fusion function. Crisp decision functions, such as the weighted majority voting or Boolean combination, provide a decision that can produce a binary relevance measure. Otherwise, it must be converted to a score in order to be used as a multiple-valued relevance measure (e.g. using the ROC space [35]). In that case, an extra validation set may be required, which is impractical in many real applications.

Given a set of positive target samples, and the availability of abundant non-target samples in the application (the CM and UM), the selection of a representative subset of representative training samples becomes essential for practical implementations. Level **A** in Fig. 2 provides a wide spectrum of techniques, in which different approaches allow for the selection of samples from distinct regions of the data distributions. For instance, the CNN finds the borderline samples, whereas using Tomek Links allows to remove both noisy and borderline samples from the set of data. On the other hand, one sided selection allows to remove noisy and borderline samples from the majority class by combining Tomek Links followed by CNN. Due to the complexity of the non-target distribution (e.g. it holds samples from all non-target individuals), non-target borderline samples are important for classifier training. These samples allow for a fine tuning of the decision frontier between classes. In this paper, the CNN has been used to select borderline samples between target and non-target data distributions, providing more relevance to the samples closer to the overlapping area [16]. In Section 4, a CNN-based strategy is proposed to consider representative samples from the target and non-target distributions, and especially those samples in their overlapping zone.

Different from uninformed selection (level **A**), an informed selection of validation samples considers the responses of the base classifiers in the ensemble, and takes advantage of the current state of knowledge of the classification system. From the rest of the (informed) ranking levels, level **B** is not considered because of the incompatibility of the internal information between classifiers. And level **E** is not considered given that the information is reduced to a single decision, and an extra validation set may be required to produce a multi-level ranking. After this reasoning, ranking measures from levels **C** and **D**) are chosen as best candidates. The graphs of the measures at these levels were analyzed (see Appendix A), and it can be seen that average margin sampling (AMS), Kullback-Leibler (KL) divergence and vote entropy (VE) present a peak in the overlapping region between target and non-target distributions. These samples in the overlapping region are of special interest for validation given that they provide a higher level of information. From the aforementioned measures, VE shows a lower resolution than KL and AMS, and the smoothness of the KL divergence curve shows a better representation of the overlapping area. Furthermore, the KL divergence takes advantage of the posterior probabilities estimated by the base classifiers, and allows to select the samples that provide the highest level of information, which appear in the overlap areas between classes, close to the decision boundaries. In this paper, the KL divergence is employed to implement a strategy for assessing the relevance of

reference samples in managing a fixed size memory of validation samples.

3.2. Update of Biometric Systems:

In the literature, several approaches allow for supervised adaptation providing reliable results [2, 9, 19], and yet obtaining labeled reference samples is costly or impractical. To overcome this difficulty, some *semi-supervised* methods have been introduced for automatic template updates [12, 13, 14, 36, 37, 38, 39]. This paper focuses on the semi-supervised updating of biometric models. *Self-training* and *co-updating* are two well-known algorithms for semi-supervised adaptation using template matching.

In *self-update* methods [14], the biometric models are first designed storing samples from a labeled data set D_L in a template gallery $\mathcal{G}$. Prediction is possible by applying a decision threshold γ^d to the similarity score produced after template matching. Then, during operations, similarity scores are produced for the unlabeled samples, and those with a high degree of confidence (surpassing an updating threshold $\gamma^u \geq \gamma^d$), are integrated to the gallery $\mathcal{G}$, thereby updating the corresponding biometric models. The notion of "high degree of confidence" is subjective, and depends on both the matching algorithm and the application domain, but an update threshold higher or equal than the prediction threshold is commonly used. This procedure is detailed in Algorithm 1.

Algorithm 1: Self-update algorithm to adapt a gallery for template matching.

Input :
$\mathcal{G} = \{t_1, ..., t_N\}$ //*
Gallery with initial templates
Unlabeled adaptation set
$\mathcal{G}' = \{t_1, ..., t_N, ..., t_M\}, M \geq N$ //*
Updated template gallery
$\mathcal{G}' \Leftarrow \mathcal{G}$ //*
Initialize with $\mathcal{G}$
for $l = 1, ..., L$ **do**
  //For all templates in the gallery $t_l \in \mathcal{G}$
  for $n = 1, ..., N$ **do**
    $s_{n,l} \Leftarrow similarity\_measure(d_l, t_n)$ //*
    Compute score against all samples in $\mathcal{G}$
  $s_l \Leftarrow \max\{s_{n,l} : n = 1, ..., N\}$
  if $s_l > \gamma^u$ **then**
    $\mathcal{G}' \Leftarrow \mathcal{G}' \cup d_l$ //*
    Include the sample surpassing γ^u in the new data set

$D = \{d_1, ..., d_L\}$ //*
Output :
Estimate threshold $\gamma^u \geq \gamma^d$ for the templates in $\mathcal{G}$;
//For all samples $d_l \in D$

Co-update is a semi-supervised learning strategy adapted for use with two diversified matchers with galleries specialized on distinct biometric traits, which are designed to improve performance mutually [14]. For example, in [14], authors propose the use of fingerprints and the face, using co-training for semi-supervised updates of the facial and fingerprint models. Algorithm 2 presents the co-training algorithm. The procedure starts with the design of the two matchers with the labeled templates in galleries $\mathcal{G}_1$ and $\mathcal{G}_2$, and selecting ad-hoc the thresholds for decision (γ_1^d and γ_2^d) and update (γ_1^u and γ_2^u). Once the unlabeled sets D_1 and D_2 are collected, both matchers are used to label the samples, and those with high degrees of confidence (at least in one of the matchers) are added to the updated galleries $\mathcal{G}'_1$ and $\mathcal{G}'_2$. Also the decision and update thresholds are be updated over time in accordance with the newly acquired

data. A potential advantage of the co-update algorithm is that it can retrieve update samples that are not typical of the distribution of target data from a single trait, allowing adaptation to diverse, possibly abrupt changes.

The advantages of adapting a biometric system using operational data carries an inherent risk. There exists a trade-off between the false updates and false rejections that affect of performance. A conservative threshold (or other parameters in the biometric model) may allow a system without false updates, but also a system that is never adapted to changes in the environment. Conversely, a less conservative threshold may contribute to increase in the number of false updates and the inherent deterioration of biometric models. Following this reasoning, we can easily see that a good selection of adaptation criteria (decision threshold) is crucial in the design of the system.

Algorithm 2: Co-update algorithm to adapt a gallery for template matching.

Input :
$\mathcal{G}_1 = \{t_1^1, ..., t_{N_1}^1\}$ and $\mathcal{G}_2 = \{t_1^2, ..., t_{N_2}^2\}$ //*
Galleries with initial templates $D_1 = \{d_{1,1}, ..., d_{L,1}\}$ and $D_2 = \{d_{1,2}, ..., d_{L,2}\}$ //*
Unlabeled adaptation sets, $d_{l,1}$ corresponds to $d_{l,2}$ **Output** :
$\mathcal{G}'_1 = \{t_1^1, ..., t_{N_1}^1, ..., t_{M_1}^1\}$, $M_1 \geq N_1$ //*
Updated galleries for both modalities $\mathcal{G}'_2 = \{t_1^2, ..., t_{N_2}^2, ..., t_{M_2}^2\}$, $M_2 \geq N_2$ *Estimate thresholds* $\gamma_1^u \geq \gamma_1^d$ *and* $\gamma_2^u \geq \gamma_2^d$ *for the* $\mathcal{G}_1$ *and* $\mathcal{G}_2$ *respectively* ;
//For each gallery G_i, $i = 1, 2$
for $i = 1, 2$ **do**
  $\mathcal{G}'_i \Leftarrow \mathcal{G}_i$ //*
  Initialize with templates in the gallery i //For all samples $d_{l,i} \in D_i$
  for $l = 1, ..., L$ **do**
    //For all templates in the gallery $t_{n,i} \in \mathcal{G}_i$
    for $t_{n,i} \in \mathcal{G}_i$, $n = 1, ..., N_i$ **do**
      $s_{n,l,i} \Leftarrow$ *similarity_measure*$(d_{l,i}, t_{n,i})$ //*
      Compute score for all $d_n \in D_i$
    $s_{l,i} \Leftarrow \max\{s_{n,l,i} : n = 1, ..., N_i\}$;
    if $s_n^i > \gamma_i^u$ **then**
      $j \Leftarrow$ mod $(i + 1, 2) + 1$ //*
      Samples added to the complementary gallery $\mathcal{G}'_j \Leftarrow \mathcal{G}'_j \cup d_{l,j}$

Other semi-supervised approaches take advantage of neural or statistical classifiers in the construction of biometric models. For instance, in [37], a view representation that combines facial and torso-color histograms was used with bunch graph matching for adaptive person recognition. The system is capable of updating existing biometric models and to automatically enroll unknown individuals based on a double thresholding strategy. Update was performed on operational video streams that provide high sequence-to-entry similarity, measure of confidence. The sequence-to-entry similarity is the average of maximum frame-to-entry similarity values, which in turn was defined as the maximum similarity value over all facial representations in a database entry [37]. Bayesian networks were also used to recognize facial expression and detect faces using a stochastic structure search algorithm [40]. This approach combined labeled and unlabeled samples to train the Bayesian networks, and seek for the Bayesian network structure that provided the minimum probability of error, using maximum likelihood estimation. SVMs with locality preserving projections have also been combined to update facial models, by incorporating information from operational ROIs taken from video [41]. The algorithm first builds a data model of a video sequence, and then uses semi-supervised

locality preserving projections to assemble a graph with the geometrical structure of the feature space of faces.

MCSs have also been used in conjunction with the co-training and self-training. In [42], for instance, an ensemble of five classifiers was trained with two different diversity generation techniques (bootstrap and the training of different classifiers). These techniques are based on a re-training schema for biometric model updates, and improve accuracy by 18% using the product rule for combination. Another modification of the co-training algorithm for MCS was proposed for updating only unlabeled samples that produced high confidence [43]. The five patterns with highest probability of belonging to the specific person, were selected as the most confident. This system was tested with 3 non-homogeneous classifiers in the ensemble, and provided the highest performance with a voting combination scheme. Finally, a semi-supervised classification schema based on random subspace dimensionality reduction was proposed for graph-based semi-supervised learning. In this approach, a kNN graph is built in each processed random subspace, and semi-supervised classifiers are trained on the resulting graphs, using majority voting rule for combination [44].

MCSs for semi-supervised learning in the literature have provided improved accuracy, and show the utility of unlabeled samples. In this paper, an adaptive MCS is proposed for spatio-temporal FR, that allows for semi-supervised learning from facial trajectories defined by the face tracker. It exploits the two thresholds (γ^d and γ^u) from the self-update algorithm, and the quality of tracking as a second source of confidence, characteristic borrowed from the co-update algorithm. The tracking quality allows to regroup facial regions from the same individual, and the accumulation of the predictions from the user-specific ensembles over time allow for high confident decisions.

3.3. Adaptive Face Recognition:

In the literature, adaptive FR systems have traditionally incorporated new training data to update the selection of templates from a facial database, using clustering and editing techniques. Processing thus allows an improved representation of intra-class variations to be obtained using a sole template. These systems were proposed to improve facial models considering the intra-class variations from input samples [36].

Recent work on the *supervised update* of facial models includes an FR system formed from an adaptive MCS. A DPSO based incremental learning strategy has been proposed for video-based access control. It allows the evolution of an ensemble of heterogeneous multi-class classifiers from new data, using an LTM to store validation samples for fitness estimation and to stop training epochs. This approach reduces the effect of knowledge corruption [9]. Another adaptive MCS for designing and updating facial models is composed of an EoD per individual, an LTM and a dynamic optimization based training module. When a new data block becomes available, a diversified pool of ARTMAP neural networks is generated by a DPSO based learning strategy. The combination function is updated using Boolean combination (BC) [2]. Learn++ is another ensemble-based incremental learning technique that has been tested on FR problems [7]. It performs supervised incremental learning by training and integrating a new batch of weak classifiers to the ensemble when new reference samples become available. These weak classifiers are generated using a bagging strategy inspired in the AdaBoost algorithm.

Semi-supervised approaches for facial model update are generally based on the classification similarity. For

instance, in [12], semi-supervised learning has been applied to FR with self-training, using an Euclidean distance-based measure of similarity. In each iteration, the PCA-based feature space is updated with the newly acquired soft-labeled samples. In [45], the authors propose a method for combining tracking and recognition to build a facial model based on co-training. This method is used to label face samples and thus to build a learning dataset for each user. Their initial facial model consists of a single manually selected frontal face picture, and the extraction of new face samples is done off-line. In order to identify informative training samples, they replace the second classifier with a tracker. An extension to the self-update algorithm named the Graph Mincut [38], has been proposed to update templates. This approach analyzes the underlying structure of operational data, and a pair-wise similarity measure between operational data and existing templates is used to draw a graph that relates these samples.

A representative example that exploits not only the classification similarity, but also video information, is presented in [13]. The authors propose an update strategy called incremental template update. It is based on the similarity between input samples and gallery templates. It exploits the frequency of detection on the complete sequences for the individuals in front of the camera, and combines this frequency with the coordinates of the detection within the last frame in the sequences.

4. A Self-Updating System for Face Recognition in Video Surveillance

In this paper, an adaptive MCS is proposed for spatio-temporal FRiVS that allows for partially-supervised learning from facial trajectories. As shown in Fig. 3, the proposed system is comprised of a segmentation module for face detection, a face tracker, a modular classification system with one EoD per individual of interest, a decision fusion system, a design/update system, and a sampling selection system.

During operations, informations from a tracker and modular classifiers (user-specific EoDs) are integrated at a decision fusion level for enhanced video-to-video FR. A *highly confident* trajectory[2] T is associated with an individual of interest k when the number of accumulated positive predictions of a EoD over a fixed-size window of ROIs surpasses a predefined *detection* threshold (γ_k^d).

The MCS allows for self-update of facial models over time, based on diverse ROIs captured within trajectories. When an individual of interest k is detected by the system within a high quality trajectory T, and the number of positive predictions surpasses a second higher *updating* threshold, $\gamma_k^u \geq \gamma_k^d$, all the corresponding facial ROIs are combined (as target samples) with selected non-target samples from the CM and UM to produce a labeled training data set D to update a facial model. User-specific EoDs are updated using a *learn-and-combine* strategy, thereby avoiding knowledge corruption [2]. A new pool of detectors (2-class classifiers) is generated with D, and combined with previously learned detectors to adapt the EoD. For an accurate estimation of a fusion function and selection of an operations point, the LTM stores and updates a representative set of validation samples. Finally, a strategy based on

[2]The notation T_k is reserved for trajectories assigned to an individual of interest k, for a design-update phase, e.g. labeled trajectories, whereas T is used for unlabeled operational trajectories.

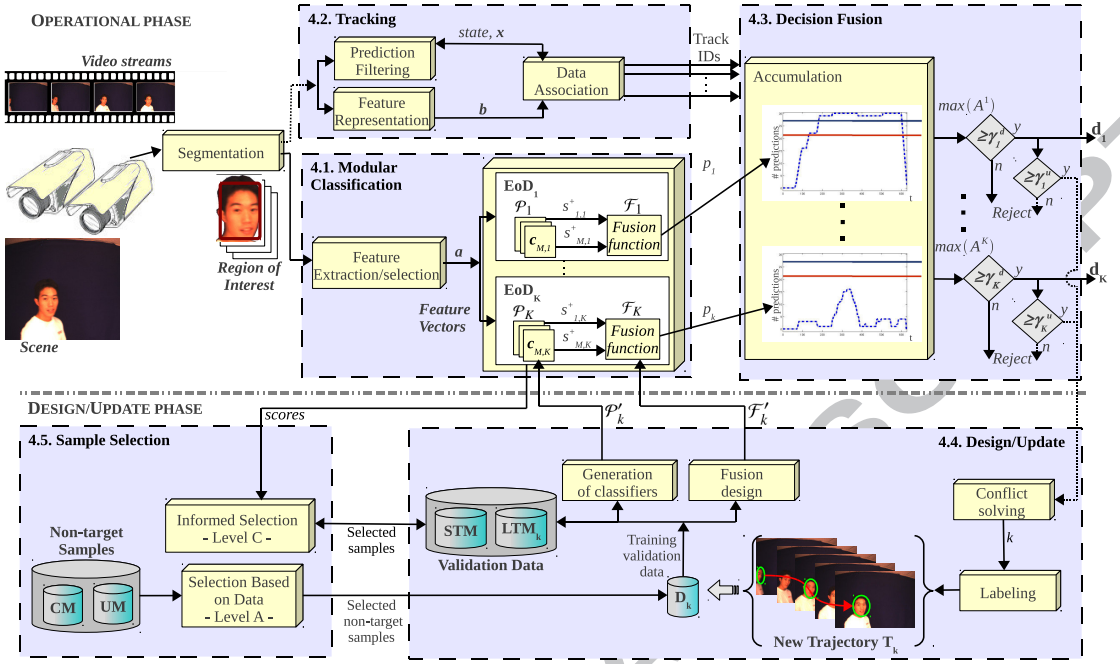


Figure 3. Block diagram of the proposed self-updating system for spatio-temporal FRiVS.

Kullback-Leibler divergence is employed to rank and store only the most representative facial samples from the LTM. It combines ROI matching scores of user-specific ensembles within high quality facial trajectories captured with a tracker, for efficient self-updating of facial models over time. The set of ROIs associated with trajectories provide diversity for robust EoDs design.

4.1. Modular Classification System:

A modular classification architecture is proposed in this paper. Individual-specific EoD allow for enhanced classification accuracy when only a limited number of training samples is available for system design [10]. Accordingly, each EoD estimates discriminant bounds between the target (individuals of interest) and non-target (the rest of the world) classes. Each ensemble EoD_k is comprised of a pool of 2-class classifiers $\mathcal{P}_k = \{c_{1,k}, ..., c_{M,k}\}$, and a fusion function $\mathcal{F}_k$ that is designed using a validation set D_k^c, for $k \in \{1, ..., K\}$.

During operations, each ensemble member $c_{m,k}$ produces an output score $s_{m,k}^+(\mathbf{a})$ for a given feature vector $\mathbf{a}$ corresponding to an input ROI. The scores are then combined using $\mathcal{F}_k$. Each individual-specific EoD_k produces an output prediction $p_k(\mathbf{a})$. Positive predictions are then accumulated over time in the decision fusion system to produce a composed decision (see Fig. 3).

The fusion function $\mathcal{F}_k$ holds a set of operations points. Each point is comprised of classifier specific thresholds and combination functions (e.g. a Boolean combination or voting scheme). Depending on the strategy used for the estimation of the fusion function, a subset of the classifiers in the pool $\mathcal{P}_k$ is selected to maximize performance.

The evaluation of the operations points on a selection set D_k^s allow to select a specific operations point in the ROC space, given a predefined acceptable fpr. Given that the system seeks to maximize the tpr under a constraint of the amount of false positives, the convex hull is selected in order to consider only the points with highest tpr. If there is no operations point for a specific fpr, a virtual classifier is produced by interpolating the closest adjacent operating points [46].

Finally, the self-update is achieved by using adaptive EoDs, each one is capable of supervised incremental learning. A *learn-and-combine* strategy is employed to maintain performance even after several adaptations, yet avoid knowledge corruption associated with many incremental learning classifiers [2].

4.2. Tracking System:

As shown in Fig. 4, the face tracker initializes a new trajectory with the first facial ROI captured by the segmentation system in a different area of the scene. As the tracker follows the facial region through the scene, the segmentation system captures high quality facial ROIs for some of the frames, allowing to produce a trajectory (a trajectory T is defined over consecutive frames). Note that the segmentation module does not retrieve a facial region from all frames. The diverse set of facial ROIs belongs to the same individual is defined by the tracker. When the tracking quality Q_T falls under a (manually) pre-defined overall quality threshold ($Q_T < \gamma^T$), its trajectory is dropped.

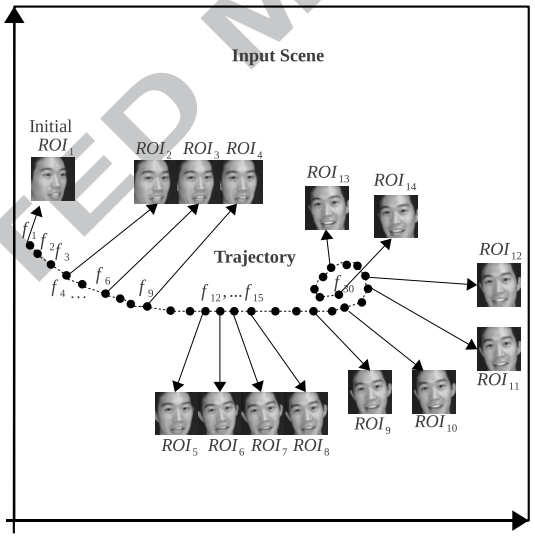


Figure 4. Illustration of the trajectory formation process within 30 frames of a FIA video. The tracker is initialized with ROI_1 and follows the face of an individual (person with ID 2), through the scene (capture session 1). f_i represents the position of the face in the camera view for frame i. The ROIs in the trajectory are produced by segmentation at $f_1, f_4, f_6, ..., f_30$, and the track is dropped at f_30. The trajectory is $T = \{ROI_1, ROI_2, ..., ROI_{14}\}$.

4.3. Decision Fusion System:

The adaptive MCS detects the presence of individuals of interest based on the number of positive EoD_k predictions over trajectories. Given a high quality trajectory T, each EoD_k generates a prediction $p_k(\mathbf{a}_n)$ for each sample $\mathbf{a}_n$

associated with a ROI in the trajectory. Output predictions from EoD_k over the ROI samples of a trajectory T, at the selected operations point, are defined by the set $\mathbf{P}_k = \{p_k(\mathbf{a}_1), ..., p_k(\mathbf{a}_N)\}$, associated with each input ROI sample $\mathbf{a}_n$. Negative predictions set $p_k(\mathbf{a}_n) = 0$, and positive ones set $p_k(\mathbf{a}_n) = 1$. The decision fusion system accumulates the number of positive predictions A_k of each EoD_k on fixed size window W according to:

$$A_k = \sum_{i=0}^{W-1} p_k(\mathbf{a}_{(W-i)}) \quad \in [0, W] \tag{1}$$

For instance, a window of size $W = 30$ accumulates the last 30 positive predictions from the same trajectory. Each EoD_k accumulates a sequence of positive predictions that range from 0 (EoD_k made only negative predictions for W), to a maximum of W (EoD_k made only positive predictions for the last W ROIs).

Based on these accumulations A_k, for $k = 1, ..., K$, the system produces decisions. If A_k surpasses threshold γ_k^d, the system detects the presence of individual k and alerts the operator. Furthermore, if A_k surpasses the update threshold γ_k^u, the trajectory is suitable for self-updating of the corresponding EoD_k. Given the negative effects on performance caused by false updates, threshold γ_k^u is greater or equal to γ_k^d.

For each EoD_k, the detection threshold γ_k^d is estimated using a validation set composed of one positive and several negative trajectories. In this way, a single target trajectory is required for design and update of the facial model. An accumulation curve is computed for each trajectory in the validation dataset. The *higher negative envelope* (*hne*) is defined as the curve formed from the highest A_k values of the negative accumulation curves. The detection threshold for EoD_k is computed as the maximum value in the *hne* plus the maximum difference between the *hne* and the *positive accumulation curve* (*pac*) for the corresponding individual k:

$$\gamma_k^d = \max\{hne(f_i) : i = 1, ..., |T_k|\} + \left(\frac{\max\{pac(f_i) - hne(f_i) : i = 1, ..., |T_k|\}}{2}\right) \tag{2}$$

where f_i is the frame number i in the trajectory. By considering the presentation order of the target (positive) and non-target (negative) facial regions, the time information is included in the threshold estimation for specific facial models. The adaptation threshold γ_k^u is set to a value equal to or greater than γ_k^d:

$$\gamma_k^u = \gamma_k^d + \Gamma_k \tag{3}$$

where Γ_k is a user-defined real value between 0 and $(W - \gamma_k^d)$. Fig. 5 illustrates the measures used in the threshold estimation strategy, presenting the *pac* and the *hne*. The reliability of γ_k^d and γ_k^u estimates grows with the number of non-target trajectories present in the validation set.

When the accumulative curve corresponding to an operational trajectory T surpasses the detection threshold γ_k^d for one or more EoDs, the system outputs the corresponding decision signals. The output to the decision support system lists all individuals of interest that are detected in the scene.

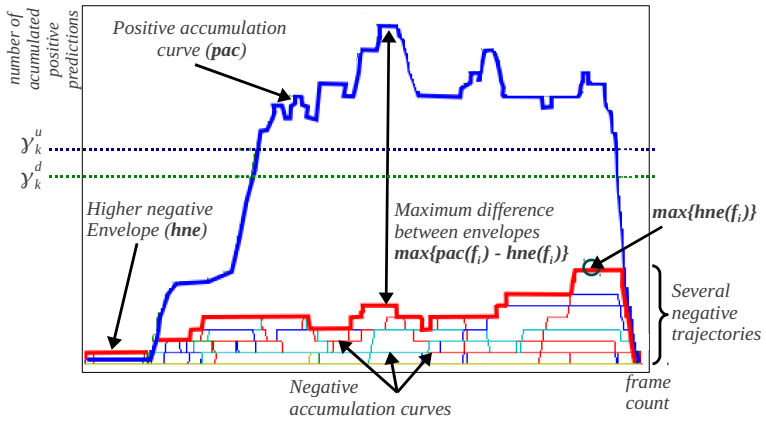


Figure 5. Detection and update threshold estimation on validation trajectories at the decision level.

4.4. Design/Update System:

Given a trajectory T, if the number of accumulated positive predictions from the EoD_k surpasses the update threshold, $A_k \geq \gamma_k^u$, the *design/update* system assigns the corresponding label to the trajectory. If conflict occurs (two or more EoDs detect the same trajectory as suitable for update), the EoD_k with highest A_k value is selected. If two or more trajectories present the same A_k value, the system is prevented from updating, and these conflicting trajectories are stored for further analysis by a human expert.

Once the trajectory has been successfully tested for conflicts, the system assigns the label k to all the patterns corresponding to the facial ROIs of the trajectory T, and it becomes a labeled trajectory T_k. An advantage of the proposed system is the incorporation of diversified information into facial models of detected individual. Self-updating provides EoDs with a greater diversity of samples captured under various conditions (pose, lighting, etc). These samples allow for a more accurate definition of the boundaries between target and non-target individuals in accordance with the most recent facial samples.

When a new trajectory T_k is detected and labeled for update, it is divided into three subsets in order to follow a *learn-and-combine* strategy. A CNN based selection algorithm allows to retrieve borderline and distinctive samples from the negative distribution, by selecting negative samples from the CM and UM (see Section 4.5). The CM database is comprised of a set of trajectories from the individuals of interest, excluding individual k; and the UM database is comprised of trajectories from other non-target individuals that represent the rest of the world, e.g. random individuals that appear frequently in the scene. The subset D^t is used for training[3], D^e for validation on the number of training epochs, and D_k^f for optimization of classifier hyperparameters. Then, some ensemble generation strategy (e.g. random subspace methods, boosting and bagging, [47]) allows to generate a diversified pool of classifiers, and add them to the previous pool $\mathcal{P}_k$. The samples from the validation sets (D^e and D^f) are then mixed with samples from the LTM_k[4],

[3]For simplicity of notation, the k has been omitted from all design data blocks, e.g. $D_k^t \equiv D^t$.
[4]Note that the LTM_k is initially empty, and filled with positive and negative samples after the initial design.

stored to a short term memory (STM_k), randomized and divided into two subsets (D^c and D^s). The classifiers from the pool $\mathcal{P}_k$ and the fusion function $\mathcal{F}_k$ are selected and combined using D^c, and the operations point is selected using D^s. The process is repeated for all the EoDs. In summary, each EoD_k is updated with new ROIs from a trajectory T_k by generating new base classifiers, adding these to a pool $\mathcal{P}_k$, and updating the fusion function according to the old and new validation samples.

Algorithm 3: Design and update of a user-specific ensemble of detectors, EoD_k.

Input : T_k, $EoD_k = \{\mathcal{P}_k, \mathcal{F}_k\}$, LTM_k, UM, CM
Output : EoD'_k, LTM'_k //*
Updated $EoD'_k = \{\mathcal{P}'_k, \mathcal{F}'_k\}$ and LTM'_k
Divide T_k in D^t, D^e, D^f evenly //*
T_k keeps only positive samples — $D^t \Leftarrow CNN\_NEG\_SEL(D^t, UM, CM)$ //*
Form 2-class data sets with target (+) vs. — $D^e \Leftarrow CNN\_NEG\_SEL(D^e, UM, CM)$ //*
non-target (-) samples (see Algorighm 4) — $D^f \Leftarrow CNN\_NEG\_SEL(D^f, UM, CM)$;
$P'_k \Leftarrow \{c'_{1,k}, ..., c'_{M,k}\}$ //*
Generate a pool $\mathcal{P}'_k$ using D^t, D^e and D^f — $\mathcal{P}_k \Leftarrow \mathcal{P}'_k \cup \mathcal{P}_k$ //*
Combine old and new classifiers in the pool — $STM_k \Leftarrow D^e \cup D^f \cup LTM_k$ //*
Store old and new validation samples in STM_k — Divide STM_k in D^c and D^s evenly ;
$\mathcal{F}'_k \Leftarrow FUSION(D^c, D^s, fpr)$ //*
Estimate fusion function given a predefined fpr — $EoD'_k \Leftarrow \{\mathcal{P}'_k, \mathcal{F}'_k\}$ //*
Updated selection of classifiers and fusion function — $LTM'_k \Leftarrow KL\_SEL(STM_k, \lambda_k)$ //*
Use KL to replace samples in LTM_k with most — //*
informative in STM_k

If the size of the LTM_k for EoD_k is λ_k, the size of the STM_k is chosen to be $2\lambda_k$ in order to store enough new and old validation samples. This follows the assumption that old (from LTM_k) and new samples are equally relevant. Then, the validation samples in the STM_k are ranked according to Eq. 4 (see Section 4.5), and the λ_k samples with the highest values are stored in the LTM_k.

4.5. Sample Selection:

Sample Selection for Training. Positive samples from the aforementioned design/update trajectory T_k are coupled with negatives from the CM and UM to form the learning set D. Negative samples from the CM and UM are stored in a single global fixed size memory capable of storing recent facial captures from non-target individuals. The size of this memory should be determined according to system requirements, but it should be large enough to store trajectories from several non-target individuals. In practice, the UM can be regularly updated with trajectories from random or selected individuals (e.g. employees or frequent clients), and the CM is updated every time the system receives update trajectories. The CNN subsampling strategy [16] is employed to reduce the bias of training 2-class classifiers with imbalanced data sets (limited positive vs. abundant negative samples). This method selects those samples from both classes that lie on the area of overlap or are difficult to classify (outliers). Nevertheless, these samples are complemented with distinctive samples from the underlying distributions. Distinctive samples are selected by storing all available positive references as well as a uniform sampling of negative ones from UM and CM after CNN selection. This CNN negative selection strategy resembles one sided selection in the application of CNN

selection, however the CNN negative selection does not discard borderline samples, and includes distinctive samples through random selection. This permits the update of the ensemble considering not only the most relevant past and present samples close to decision bounds, but also typical samples distinctive of the most recent states of distributions of data.

The CNN negative selection strategy is detailed in Algorithm 4. When a trajectory T_k is provided to the system for training/update, the corresponding ROIs are used to build dataset of positive samples D^+, and a set D^- is formed with samples from the UM and CM. The CNN algorithm is then applied to $D^+ \cup D^-$ to select a consistent subset for design of the binary base classifiers. The resulting dataset D comprises three parts of equal size: (1) the complete set of positives D^+, (2) the negative samples selected by CNN (close to the decision boundaries) $D^{-'}_{cnn}$, and (3) a uniform random selection of non-borderline negatives D^-_d. In this way, D contains all target samples and twice more non-target samples. Algorithm 4 makes no assumptions concerning the probability distribution of the positive and negative samples, and permits an unbiased selection of negative samples, based solely on the distribution of the new samples.

Algorithm 4: *CNN_NEG_SEL*. Select negative samples to design the system.

Input : D^+, UM //*
Positive and negative samples from UM and CM data bases
Output : D //*
Design dataset with all positive and selected negative samples
$D^- \Leftarrow UM \cup CM$ //*
Consider all negative samples from UM and CM — $[D^+_{cnn}, D^-_{cnn}] \Leftarrow CNN(D^+, D^-)$ //*
Samples selected by CNN — $np \Leftarrow |D^+|$ //*
Number of positive samples — $D^{-'}_{cnn} \Leftarrow RAND\_SEL(D^-_{cnn}, np)$ //*
Select np negatives from D^-_{cnn} belonging to UM and CM evenly — $D^-_d \Leftarrow RAND\_SEL(D^-, np)$ //*
Select np distinctive negatives from D^-, not selected by CNN — $D \Leftarrow D^+ \cup D^{-'}_{cnn} \cup D^-_d$

Management of LTM_k. Level **C** ranking measures (see Section 3.1) permit the selection of samples from the LTM_k that are difficult to classify by the ensemble members (in Fig. 2). These samples are distinctive of the decision bound between the target and non-target classes, as estimated with the base classifiers in the EoD. The disagreement of base classifiers on a determined validation sample is proportional to its difficulty, give a degree of information for border specification when the fusion function is estimated. This is also valid for the accurate selection of operations points. Among ranking measures available in the literature, the Kullback-Leibler divergence produces a continuous measure of the disagreement between the ensemble members that covers the overlapping area between class distributions (see analysis in Appendix A). Accordingly, the KL divergence permits the exploitation of the knowledge from base classifiers to select the validation samples that provide the highest level of information. Even more, its continuous ranking values permit the discrimination between two samples that appear very close to each other in the feature space. The KL divergence of an input sample **a** is computed using:

$$KL(\mathbf{a}) = \frac{1}{M}\sum_{m=1}^{M}\left(\sum_{i\in\Omega} s_m^i(\mathbf{a}) \log \frac{s_m^i(\mathbf{a})}{\hat{P}^i_{EoD_k}(\mathbf{a})}\right) \quad (4)$$

where M is the number of classifiers in the ensemble EoD_k, and $\hat{P}^i_{EoD_k}(\mathbf{a})$ given by (5) is the consensus probability that the class $i \in \Omega$ is the correct label for sample $\mathbf{a}$, given the scores $s_n^i(\mathbf{a})$ produced by the base classifiers:

$$\hat{P}^i_{EoD}(\mathbf{a}) = \frac{1}{M}\sum_{n=1}^{M} s_n^i(\mathbf{a}) \quad (5)$$

The value of KL divergence is proportional to the level of information provided by a sample $\mathbf{a}$. The most informative samples present the largest average difference between scores of any single committee member and the consensus.

Algorithm 5 details the selection process that considers all the validation samples in the STM_k. Given an EoD_k, the KL_SEL algorithm selects the λ_k most challenging samples from the validation set, providing those samples lying on the overlapping area according to the agreement of the ensemble members. When a validation dataset D is presented to the algorithm, all samples are ranked according to the KL divergence using the scores produced by all the base classifiers in the pool $\mathcal{P}_k$. The λ_k highest ranked samples are retained, while the less informative ones are discarded, maintaining the proportion of target and non-target samples. Thus, the ranking method is based on past and present information on samples that are difficult to classify, according to older and newer classifiers.

Algorithm 5: Subsampling using the KL divergence, $KL\_SEL(input = \{D, s_k(a_i), \lambda_k\}, output = \{Dr\})$.

Input : D, $s_k(a_i)$, λ_k //*
Data block, scores $s_k(a_i)$, $a_i \in D$ produced by EoD_k //*
and size of the LTM_k **Output** : Dr //*
Data block with λ_k representative samples from D
//For each sample in the data block
for $a_i \in D$ **do**
  $relevance_i = KL(s_k(a_i))$ //*
  Compute the KL divergence according to Eq. 4
$D \Leftarrow SORT(D, relevance, dec)$ //*
Sort D in decreasing order, according to $relevance_i$ $Dr^+ \Leftarrow FIRST\_POSITIVES(D, \lceil \frac{\lambda_k}{2} \rceil)$ //*
Positive samples with highest KL divergence $Dr^- \Leftarrow FIRST\_NEGATIVES(D, \lceil \frac{\lambda_k}{2} \rceil)$ //*
Negatives with highest KL divergence $Dr \Leftarrow Dr^+ \cup Dr^-$

5. Experimental Methodology

Some methodologies for performance evaluation of adaptive biometric systems divide the design-update data into subsets, and use a same independent test set to show the evolution of performance [8, 12, 13, 23]. Others divide the unlabeled data set into subsets, and progressively update on a subset while testing on the next subset [14, 38]. This last approach is followed in this paper. The main task under evaluation is detecting the presence of

individuals of interest in semi-constrained environments, and the experimental protocol was designed to study the evolution of system performance in a changing classification environment. The adaptive MCS is first trained using design trajectories from an enrollment session (D), then the updating process was performed on three different capture sessions D_t, $t = 1...3$ in a video-to-video recognition scheme. The system is adapted after the presentation of each session D_t with the trajectories detected as positives, and the performance is evaluated using ROC and PROC spaces after the presentation of a different capture session D_{t+1}.

5.1. Video Surveillance Database:

The proposed system was characterized in a video surveillance scenario using the Carnegie Mellon University Face in Action (FIA) database [48]. The FIA database contains 20-second videos that capture the faces of 180 participants that simulate a passport checking scenario. Capture speed is fixed to 30 frames per second, with a resolution of 640×480 pixels. An array of 6 cameras was positioned at the face level to capture the scene. However only the 2 frontal cameras are considered here. They are positioned at 0^o (frontal) and $\pm 72.6^o$ angle with respect to the individual. Three of the cameras were set at a zoomed focal-length (8-mm), resulting in face areas over 300×300 pixels. The other three cameras were set at an unzoomed focal length (4-mm), resulting in face areas over 100×100 pixels. Data was captured in three sessions separated by a three-month interval for each individual. Facial regions of interest (ROIs) were detected in videos using the Viola-Jones algorithm [49]. Visual tracking was also applied on video sequences, initializing the Continuously Adaptive Mean Shift (CAMSHIFT) [50] with the first face detected. All images were scaled to 70×70 pixels, which is the maximum resolution of the smallest face detected by the Viola-Jones algorithm. The Multi Scale Local Binary Patterns (MS-LBP) [51] feature extractor was used with three block sizes (3x3, 5x5 and 9x9), in conjunction to pixel-intensities features. These features were stacked, and the 32 principal characteristics (PCA) were selected to form the feature vectors.

Ten individuals of interest were selected, and one EoD was designed for each of them. Variants in expression, aging, pose, haircut, whiskers and beard made the problem more challenging (see Fig. 6). From the remaining individuals, 88 were selected to build the universal model (UM), and the rest were considered as unknown individuals and only appeared on the test datasets. Note that samples from individuals belonging to the UM do not appear in the test set, thus avoiding a positive bias.

One trajectory was retrieved from each individual in each capture session, and organized in four datasets. The total number of ROI samples contained in the trajectories from each design/update datasets is summarized on Table 2. As shown in the table, the CM is comprised of 9 trajectories from non-target individuals in the cohort, and the number of ROI samples is different for each EoD. For instance, the CM of individual with FIA ID=2 is comprised of 1,746 reference ROI samples. ROI samples in the UM are 6,167, 2,966, and 3,188 are retrieved from each data block D_1, D_2 and D_3 respectively, divided in 88 non-target trajectories per block. Finally, the total number of ROI samples in the trajectories from unknown individuals is 10,240, 10,967, and 5,104 for D_1, D_2 and D_3 respectively. The fixed size memory containing the UM and CM is maintained with a first-in-first-out strategy, and it stores up to 12,000 facial

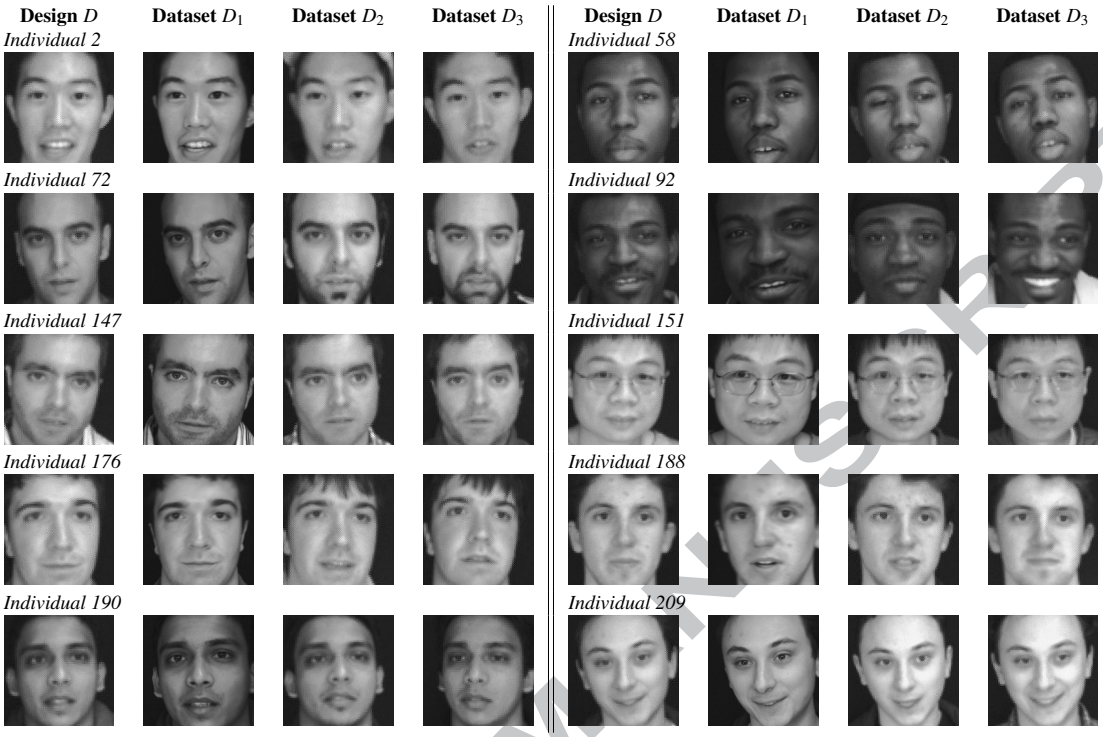


Figure 6. Sample images from individuals of interest detected in video sequences from the FIA database.

regions belonging to the most recent trajectories from non-target individuals.

Table 2. Number of ROI samples in design and test trajectories for each individual of interest.

FIA Individual (k)	$\|T_k\|, T_k \in D$	$\|T_k\|, T_k \in D_1$	$\|T_k\|, T_k \in D_2$	$\|T_k\|, T_k \in D_3$
ID 2	149	114	109	119
ID 58	202	176	215	172
ID 72	223	144	184	151
ID 92	180	125	125	167
ID 147	235	128	163	161
ID 151	216	80	187	135
ID 176	113	90	210	126
ID 188	148	118	172	192
ID 190	190	132	92	88
ID 209	239	121	162	137

5.2. Implementation of the Proposed MCS:

For proof-of-concept, the adaptive MCS proposed in Section 4 is implemented in the following way. The classification system in the MCS is formed of an adaptive EoD per individual [2]. The base classifier for the EoD is the Probabilistic Fuzzy ARTMAP (PFAM) [52], which combines Fuzzy ARTMAP density estimation for learning category prototypes, with a non-parametric posterior probability distribution procedure inspired by the Probabilistic Neural Networks during the operational phase. A diversified pool of base classifiers is generated through a dynamic particle swarm optimization (DPSO) learning strategy [9]. The DPSO learning algorithm was initialized with a swarm

of 60 particles, 6 sub-swarms of maximum 5 particles, and a maximum of 30 iterations (+5 to ensure convergence). The classifier corresponding to the global best particle, as well as the 6 local best classifiers from each sub-swarm are added to the ensemble. Finally, new classifiers are combined with previously trained ones ($\mathcal{P}_k$) using the Boolean combination (BC) that operates in the ROC space [53]. BC starts by regrouping classifiers according to performance and then combines all pairs of operations points for the two best classifiers, according to their representation in the ROC space. Then, the convex hull of the new operations points is successively combined with the next best classifiers, until the overall convex hull stops improving.

The CAMSHIFT is a well known kernel-based tracking algorithm that uses region-based features representation [50]. It uses a combination of a weighting kernel and a histogram to represent the target and attain frame-to-frame object tracks, using the probability distribution of faces in video. It dynamically handles the changing distributions by adjusting the size of the search window according to the area under such a window. The internal face representation consists of the skin probability histogram of the face, and the kernel is a simple step function. During data association, two histograms q_1 and q_2 corresponding to the predicted and actual facial regions respectively are compared with the Bhattacharyya coefficient given by:

$$Q_T \equiv \hat{B}(q_1, q_2) = \sum_{u=1}^{m} \sqrt{q_1(u) q_2(u, y)} \tag{6}$$

where u varies over all histogram bins, and y is the target position. Coefficient Q_t expresses the quality of a trajectory from one frame to another in terms of the similarity between predicted and actual face regions.

5.3. Experimental Protocol:

Prior to computer simulations, four datasets were prepared using frontal videos of the FIA database. The design dataset D is comprised of the positive trajectories in the zoomed capture session 1. The adaptation datasets D_1 to D_3 are constructed with tracks from the unzoomed view of capture sessions 1 to 3, respectively. This capture scenario corresponds to an environment with gradual changes of face models due to aging. Negative samples are independently selected for each of the training/validation sets using Algorithm 4, by selecting samples from the CM and UM. Three different scenarios were prepared, with different design-update schemes.

- *Supervised learning on D only.* Considered a static system, designed on the first dataset D_1 only. The test is performed on the other D_1 to D_3 datasets, but no update (additional learning) is performed. The performance in this scenario establishes the lower bound for the semi-supervised strategy, e.g., when no update is performed by the semi-supervised system. The approaches considered in this scenario include the TCM-kNN, a single PFAM, Learn++(PFAM) and EoD (PFAM).

- *Supervised incremental learning.* The system is first designed on D, and new reference samples become available (D_1 to D_3), and are incorporated after the test is performed. It is assumed that an expert has analyzed the

video sequences of individuals enrolled to the system, and manually labels them in order to update the system. Adaptive approaches (PFAM$_{inc}$, Learn++(PFAM) and EoD$_{sup}$ (PFAM) $LTM_{KL,\lambda=\infty}$) were updated with only the new labeled data, and TCM-kNN is trained on batch mode, learning the past and new samples from scratch[5].

- *Partially-supervised learning.* Similarly to the supervised incremental learning scenario, the system is designed on D, and new information on test sessions D_1 to D_3 is incorporated when a trajectory T yields an accumulation curve that surpasses the update threshold, γ_k^u. The approaches considered in this scenario include the EoD$_{ss}$ (PFAM) with 6 different sizes of LTM: $\lambda = \{0, 25, 50, 75, 100, \infty\}$.

Learning is performed following 2x5-fold cross-validation for 10 independent experiments. Positive samples from the incoming trajectory are randomly and evenly split in 5 folds of the same size. The folds are first distributed in three different design sets, including two folds for training (D^t), $1\frac{1}{2}$ fold to stop training epochs (D^e), and $1\frac{1}{2}$ fold for fitness evaluation (D^f). Once the classifiers are trained, D^e and D^f are combined, randomized and divided into two equally distributed subsets to produce a validation data to estimate a fusion function (D^c), and to select the operations point (D^s). Negative samples are chosen from the UM as well as the CM according to CNN selection (Algorithm 4). In each training/validation dataset, 33% of positives is accompanied by approximately 58% of negatives from the UM, and the remaining 9% from the CM. About 87% of the negatives correspond to samples taken from the UM and 13% are from the cohort. This is expected, given that the superset D^- is composed of close to 13.63% of samples from the CM, and 86.37% of samples from the UM. The folds are distributed between the training/validation sets for each replication of the experiment, and average performance measures are produced with five different assignments. At replication 5, the sample order is randomized for each class and the five folds are regenerated. The procedure followed in each trial of the experiment is summarized in Algorithm 6.

Algorithm 6: Experimental protocol to evaluate each EoD_k, on a single 2×5 cross-validation trial.

```
D⁻ ⇐ UM ∪ CM //*
Trajectories in the CM and UM   EoD_k ⇐ DESIGN(T_k ∈ D, EoD_k ≡ ∅, LTM_k ≡ ∅, D⁻) //*
Design the EoD_k with Algorithm 3   Estimate γ_k^d and γ_k^u using T_k and trajectories in D⁻;
for t = 1...3 do
    Evaluate performance of the EoD_k on D_t //*
    Classifier and decision levels   D⁻ ⇐ UM ∪ CM //*
    Trajectories from CM and UM in D_t //For every trajectory in the new data block D_t
    for T ∈ D_t do
        //If the accumulated predictions surpass the update threshold
        if (A_k(T) ≥ γ_k^u) then
            T_k ⇐ T //*
            Label the trajectory with tag k   EoD_k ⇐ UPDATE(T_k, EoD_k, LTM_k, D⁻) //*
            Update with T_k (Algorithm 3)   Update γ_k^d and γ_k^d with T_k and trajectories in D⁻
```

The proposed adaptive MCS was compared to other classifiers for FRiVS. The TCM-kNN was trained with a fixed $k = 1$ on a batch learning scheme, as followed in [1]. The Learn++ algorithm was initialized to generate 7 PFAM base

[5]For a new block D_n, TCM-kNN must be trained from scratch using a data superset $D_{batch} = D \cup D_1 \cup ... \cup D_n$.

classifiers on every incremental learning step, and weighted majority voting was validated on D^c. PFAM classifiers used in all other approaches were trained using a DPSO based learning strategy to optimize their hyperparameters.

5.4. Performance Analysis:

The analysis of simulation results has been divided into three levels. First, *transaction-based analysis* shows the performance of the system based on classification decisions on each ROI. Then, a *subject-based analysis* allows a focus on specific individuals, which in turn allows for levels of performance depending on particular characteristics. Finally, a *trajectory-based analysis* shows the overall performance of the system (shown in Fig. 3), viewed by accumulating system predictions over input trajectories.

Transaction-based performance analysis is used to assess the performance of the system for matching ROI samples to facial models. The true positive rate (tpr) and false positive rate (fpr) are estimated for different (fpr, tpr) operational points, and connected to draw a receiver operations characteristic (ROC) curve. When equal priors and costs are assumed, the closest operations point to the upper-left corner corresponds to the optimal decision threshold. In applications with fpr constraint, the selection of the operations point is obtained from the graphical representation. The operations point is estimated on a validation subset used for operational predictions, providing a test (fpr, tpr) pair that reveals the generalization performance of the system at the selected point. The AUC (area under the curve) summarizes the performance depicted in a ROC graph, and the partial AUC (pAUC) focuses on a specific region of the curve, e.g. pAUC (5%) for an $fpr \leq 0.05$.

For different priors and costs of errors, the *Precision-Recall Operating Characteristic* (*PROC*) curve constitutes a graphical representation of detector performance where the impact of data imbalance is considered. The precision between positive predictions ($precision = TP/(TP+FP)$) is combined with the tpr (or *recall*) to draw a PROC curve. In general, the tpr is increased when the amount of positive (minority class) samples augments. On the contrary, the *precision* decreases with this amount. To combine precision and recall at a particular operations point, the scalar F_1 produces a single performance indicator:

$$F_1 = 2 \cdot \frac{precision \cdot tpr}{precision + tpr} \tag{7}$$

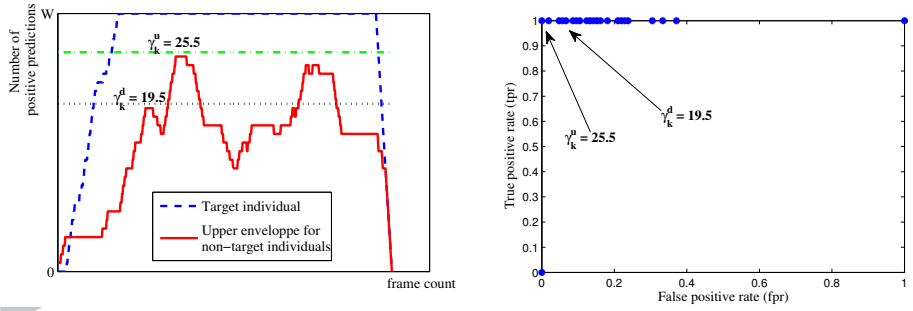
According to the "Doddington zoo" effect, the performance of biometric systems may vary drastically between individuals [54]. Instead of using the overall amount of transactions, individual-specific error rates can be assessed according to four categories (types of animals). The resemblance of individuals performance to that of these animals can reveal fundamental weaknesses, and allows the development of more robust systems. According to this characterization, the system tend to perform well in a *sheep*-like individual, irrespective of whether this individual belongs to the target or non-target class. *Goat*-like individuals belong to the positive class, but are difficult to identify (low matching scores against themselves). A *wolf*-like individual belongs to the non-target class, and consistently impersonate different targets (high scores when matched against other individuals), and tend to elevate the false positive rate (fpr) of the system. Finally, a *lamb*-like individual belongs to the target class, and is easily impersonated (high matching scores when matched against others).

Table 3. Doddington's zoo thresholds for generalization performance at the operating point with $fpr = 1\%$, selected on validation data.

Category	Positive class	Negative class
Sheep	$tpr \geq 50\%$ and not a lamb	$fpr \leq 1\%$
Lamb	At least 5% of non-target individuals are wolves	-
Goat	$tpr < 50\%$ and not a lamb	-
Wolf	-	$fpr > 1\%$

Typically, the likeliness of a user to one of the 4 aforementioned categories is defined at the score space. However, for binary classifiers, the confusion matrix can be used [1]. To establish a criterion, thresholds can be set at the fpr and fnr, and applied to each EoD_k. Table 3 shows a criterion based on a system constraint of $fpr \leq 1\%$, considering a good fnr when it is just below 50%.

Trajectory-based performance analysis allows to assess performance over time of the entire system for FRiVS (see Fig. 3). This analysis is specially relevant given that it provides a global performance assessment of the system for FRiVS, with combined impact of face segmentation, tracking, recognition and fusion. Thus, all system functions are employed to process a video stream, and decisions taken by an operator occur on a time scale longer than a frame rate. Within the decision fusion system, positive predictions of each EoD_k are accumulated over a moving window of time for input ROI samples that correspond to a high quality facial track. Assume for instance a system that produces predictions at a maximum of 30fps. Each detected ROI is presented to all user-specific EoDs of the system, which produces predictions (positive or negative) for each person enrolled to the system. Given a high quality face track, the number of positive predictions from an EoD should grow rapidly for the person of interest. Thus, the operator can more reliably detect a person of interest.



(a) Accumulation curves (b) ROC space

Figure 7. Trajectory-based analysis to evaluate the quality of a system for spatio-temporal FRiVS.

The adaptive MCS proposed in this paper accumulates the positive predictions (responses of each EoD_k) over a window of W predictions. As shown in Fig. 7a, the quality of this system can be evaluated graphically by observing the evolution of positive predictions according to the frame count (discrete time defined by the frame rate). In addition, once several individuals have appeared before of camera in a long video stream, and related trajectories have been processed, the quality of system decisions (i.e., the tpr, fpr, trr, frr) may be assessed over the range of decision

threshold values, and represented in the ROC space (see Fig. 7b).

6. Results

6.1. Transaction-Based Analysis:

Reference systems used in comparison reflect the current state-of-the-art approaches appearing in literature. TCM-kNN was proposed by Li and Wechsler in [1], and constitutes a main reference in FRiVS. Learn++ is a popular reference point in ensemble-based techniques capable of supervised incremental learning [7]. Modular architectures with a single classifier per individual have been used for FR in [5], and implemented in experiments using monolithic PFAM and PFAM$_{inc}$. These modular architectures were extended to use ensembles of classifiers per individual in [10, 19], and implemented in experiments as EoD (PFAM). In this research it is shown how the self update with the proposed approach presents higher level of performance with respect to those approaches that are not updated. And it may perform better than certain approaches that perform supervised incremental learning (e.g., Learn++), even though the proposed self update approach automatically assigns the labels to the trajectories in the update data.

Table 4 presents the average transaction-level performance for the 3 updating scenarios obtained after updating the proposed and reference systems on ROI samples from trajectories stored in data blocks D, D_1 and D_2 (while testing on D_1, D_2 and D_3, respectively). Systems are compared according to the partial AUC for a $0 \leq fpr \leq 0.05$: pAUC (5%), as well as fpr, tpr and F_1 measures at a specific operating point selected on the validation ROC curve for a desired $fpr = 1\%$. Performance for modular systems were measured for each individual (user EoD), and average values are presented. In order to have comparable results for the multi-class TCM-kNN, empirical ROC curves were estimated on validation for each individual. The selection of the operations point, as well as performance evaluation were computed after applying the specialized rejection threshold of the TCM-kNN. Note that this rejection threshold is estimated on the training data, taking advantage of the peak-side-ratio that characterizes the distributions of p-values for each class.

In the no-update scenario, the EoD (PFAM) approach is generally the most accurate approach in terms of pAUC (5%). Overall results for all approaches show a degradation in the system performance after testing on D_2, with a slight recovery after testing on D_3, indicating the presence of changes in the classification environment going from D to D_1 and to D_2. This decline in performance underscores the importance of adapting facial models as new reference videos become available.

At the selected operations point (fpr=1%), it is interesting to note that, compared to monolithic classifiers (PFAM and TCM-kNN), both ensemble-based classifiers provide lower fpr, along with a lower standard error. The only multi-class classifier used in the comparison, the TCM-kNN, yields a significantly higher fpr, even though it was designed to avoid false acceptances by using a specialized rejection threshold. This issue is related to the difficulty faced by multi-class classifiers in estimating multiple decision boundaries during the same design process: between cohort and unknown individuals, and between individuals in the cohort. Modular architectures simplify the task by

Table 4. Average transaction-level performance of the system over the 10 individuals of interest and for 10 independent experiments, Systems were designed-updated with D, D_1 and D_2, and performance is shown after testing on D_1, D_2 and D_3 respectively (shown $D_1 \rightarrow D_2 \rightarrow D_3$). In all cases, the operations point was selected using the ROC space on the validation dataset D^s at a $fpr = 1\%$, except for the partial AUC that comprises the area for $0 \leq fpr \leq 0.05$. Bold values indicate significant differences from other approaches.

System	fpr (%) ↓	tpr (%) ↑	F_1 ↑	pAUC (5%) ↑
No update (supervised learning on D only)				
TCM-kNN	20.13 ±0.42 → 24.74 ±0.50 → 18.88 ±0.53	**90.65 ±1.43** → **54.86 ±3.30** → 49.03 ±4.01	0.093 ±0.003 → 0.055 ±0.004 → 0.102 ±0.009	88.71 ±1.47 → 48.55 ±3.39 → 46.05 ±4.06
Monolithic PFAM	0.95 ±0.18 → 0.94 ±0.20 → 0.82 ±0.18	80.84 ±2.05 → 32.88 ±3.44 → 37.35 ±3.91	0.665 ±0.019 → 0.280 ±0.029 → 0.358 ±0.035	90.40 ±1.21 → 54.67 ±3.24 → 61.54 ±3.58
Learn++ (PFAM)	**0.60 ±0.07** → 0.62 ±0.08 → 0.56 ±0.06	16.90 ±2.37 → 11.36 ±2.05 → 12.13 ±2.22	0.161 ±0.017 → 0.111 ±0.013 → 0.139 ±0.018	47.87 ±2.71 → 32.62 ±2.22 → 32.67 ±2.61
EoD (PFAM)	0.62 ±0.09 → 0.64 ±0.10 → **0.53 ±0.09**	77.02 ±2.10 → 26.75 ±2.99 → 31.85 ±3.44	**0.679 ±0.018** → 0.255 ±0.025 → 0.337 ±0.032	**92.88 ±0.81** → 60.17 ±2.94 → 65.96 ±3.12
Supervised update (supervised incremental learning on $D \rightarrow D_1 \rightarrow D_2$)				
TCM-kNN	20.13 ±0.42 → 22.81 ±0.41 → 18.32 ±0.19	**90.65 ±1.43** → 54.26 ±3.22 → **87.91 ±1.67**	0.094 ±0.003 → 0.058 ±0.004 → 0.175 ±0.004	88.71 ±1.47 → 48.54 ±3.34 → 83.16 ±2.29
PFAM$_{inc}$	0.95 ±0.18 → 1.20 ±0.12 → 1.91 ±0.24	80.84 ±2.05 → 54.06 ±3.46 → 84.52 ±2.31	0.665 ±0.019 → **0.438 ±0.029** → 0.666 ±0.024	90.40 ±1.21 → 69.18 ±2.86 → 87.75 ±1.66
Learn++ (PFAM)	**0.60 ±0.07** → **0.57 ±0.04** → 1.19 ±0.11	16.90 ±2.37 → 11.87 ±1.80 → 20.57 ±2.78	0.161 ±0.017 → 0.128 ±0.014 → 0.192 ±0.020	47.87 ±2.71 → 36.81 ±2.45 → 34.19 ±2.64
EoD$_{sup}$ (PFAM) LTM$_{KL,\lambda=\infty}$	0.62 ±0.09 → 0.67 ±0.05 → 0.84 ±0.07	77.02 ±2.10 → 45.51 ±3.63 → 76.70 ±2.71	**0.679 ±0.018** → 0.404 ±0.031 → **0.691 ±0.023**	**92.88 ±0.81** → **72.03 ±2.76** → **93.64 ±0.84**
Self update (semi-supervised incremental learning on $D \rightarrow D_1 \rightarrow D_2$)				
EoD$_{ss}$ (PFAM) LTM$_{KL,\lambda=\infty}$	0.62 ±0.09 → 0.74 ±0.07 → 0.93 ±0.11	77.02 ±2.10 → 43.33 ±3.59 → 50.10 ±4.12	0.679 ±1.77 → 0.388 ±0.031 → 0.461 ±0.037	92.88 ±0.81 → 68.50 ±2.90 → 75.60 ±3.04

optimizing parameters for user-specific 2-class classifiers for determining individual-specific bounds, which provides greater discrimination when design data per target individual is limited [55]. Consequently, TCM-kNN achieves the highest tpr, but fails meeting constraints for the fpr on test data. Ensemble approaches (Learn++ and EoD) have the lower fpr, although the PFAM and EoD (PFAM) provide the highest tpr and F_1 measures. This translates to a greater discrimination for target ROI samples. Results suggest that the EoD (PFAM) can achieve the most robust overall performance to gradually changing environments.

The average results (Table 4) for the supervised update scenario show the impact on performance of updating the facial models. The degradation seen in the no-update case is reduced. The pAUC (5%) reveals that the EoD$_{sup}$ (PFAM) LTM$_{KL,\lambda_k=\infty}$ provides a significantly higher level of performance, which confirms the utility of adaptive ensembles. This approach establishes an upper bound for self-updating, given that it correctly updates facial models with every new target trajectory. As in the no-update case, it can be seen that adaptive ensembles present lower fpr but also lower tpr, and PFAM$_{inc}$ and EoD$_{sup}$ (PFAM) LTM$_{KL,\lambda=\infty}$ provide the greater discrimination on target ROI samples. TCM-kNN presents the most significant degradation in performance after testing on D_2, even though it was retrained with samples from $D \cup D_1$. However, it also presents an important recovery after testing on D_3. A Kruskall-Wallis statistical test on the pAUC (5%) between the EoD$_{sup}$ (PFAM) and PFAM$_{inc}$ gives a p-value of 0.0123, which confirms that the differences between the mean performances are significant with a 95% confidence interval.

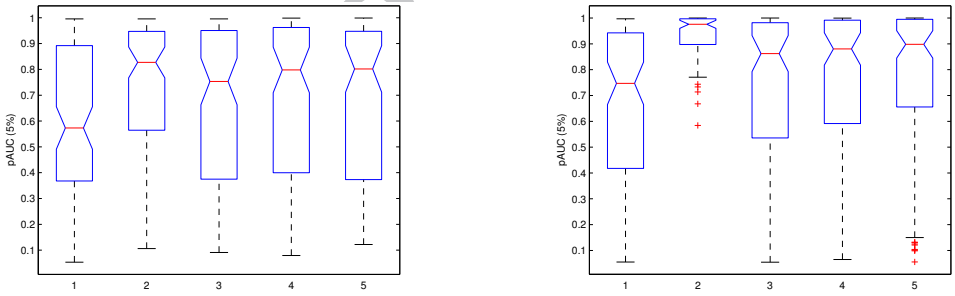
Average results achieved with the proposed semi-supervised adaptive MCS (EoD$_{ss}$) indicate that the performance is generally comparable to that of the supervised approaches in terms of pAUC (5%), although a higher fpr is eventually present. This degradation is the cumulative effect of false adaptations followed by trajectories that are incorrectly labeled (see analysis in Section 6.2). However the performance of the semi-supervised system evolves with a general improvement with respect to the no-update case as new reference data is integrated. And it remains

close to the upper bound established by the approaches that perform supervised update.

Table 5. Average transaction-level performance of the EoD$_{ss}$ (PFAM) system given different LTM sizes λ_k, after testing on $D_1 \rightarrow D_2 \rightarrow D_3$. In all cases, the operations point was selected using the ROC space on the validation dataset D^s for an $fpr = 1\%$, except for the pAUC (5%) that comprises the area for $0 \leq fpr \leq 0.05$.

System with EoD$_{ss}$ (PFAM)	fpr % ↓	tpr % ↑	F_1 ↑	pAUC (5%) ↑
LTM$_{KL,\lambda=0}$	0.62 ±0.09 → 0.96 ±0.09 → 1.55 ±0.22	77.02 ±2.10 → 44.25 ±3.60 → 51.39 ±3.95	0.679 ±0.018 → 0.373 ±0.030 → 0.428 ±0.032	92.88 ±0.81 → 65.88 ±2.92 → 72.37 ±3.09
LTM$_{KL,\lambda=25}$	0.62 ±0.09 → 1.42 ±0.22 → 1.74 ±0.24	77.02 ±2.10 → 36.17 ±3.43 → 48.83 ±3.85	0.679 ±0.018 → 0.306 ±0.029 → 0.402 ±0.031	92.88 ±0.81 → 62.80 ±3.04 → 70.95 ±3.05
LTM$_{KL,\lambda=50}$	0.62 ±0.09 → 1.25 ±0.16 → 1.44 ±0.15	77.02 ±2.10 → 35.28 ±3.35 → 48.84 ±3.90	0.679 ±0.018 → 0.304 ±0.029 → 0.407 ±0.032	92.88 ±0.81 → 62.35 ±3.08 → 71.48 ±3.13
LTM$_{KL,\lambda=75}$	0.62 ±0.09 → 1.27 ±0.16 → 1.90 ±0.29	77.02 ±2.10 → 36.76 ±3.53 → 50.13 ±3.90	0.679 ±0.018 → 0.307 ±0.029 → 0.404 ±0.032	92.88 ±0.81 → 61.50 ±3.12 → 71.84 ±3.11
LTM$_{KL,\lambda=100}$	0.62 ±0.09 → **0.92 ±0.09** → **1.45 ±0.18**	77.02 ±2.10 → **45.43 ±3.71** → **54.27 ±3.86**	0.679 ±0.018 → **0.385 ±0.031** → **0.468 ±0.033**	92.88 ±0.81 → **68.44 ±3.00** → **74.93 ±2.98**

A key parameter related to the accuracy and resources of EoD$_{ss}$ (PFAM) systems is the LTM size needed to store validation data. Table 5 shows the evolution of the average performance for LTM sizes $\lambda_k = \{0, 25, 50, 75, 100\}$ patterns. As the system self-updates, the overall performance improves when λ_k grows, at the expense of memory and computational complexity. However, this trend occurs differently for distinct individuals, as analyzed in the subject-based analysis. Finally, Fig. 8 shows the box plots for pAUC (5%) for the EoD$_{ss}$ (PFAM) system with different λ_k values. The first box in the graphs corresponds to the EoD (PFAM) that learns only on D, and establishes the lower bound in performance. The second box is the supervised EoD$_{sup}$ (PFAM) with a $\lambda_k = \infty$, and establishes the upper bound. It can be seen that pAUC (5%) grows with the LTM size. Using a $\lambda_k = 100$ provides a performance that is comparable to what is seen when $\lambda_k = \infty$.



(a) pAUC (5%) after updating on D_1 (b) pAUC (5%) after updating on D_2

Figure 8. Box plots comparing the pAUC (5%) of systems (a) after learning D_1 (testing on D_2), and (b) after learning D_2 (testing on D_3). The systems from left to right are (1) EoD (PFAM), (2) EoD$_{sup}$ (PFAM) LTM$_{KL,\lambda_k=\infty}$, (3) EoD$_{ss}$ (PFAM) LTM$_{KL,\lambda_k=0}$, (4) EoD$_{ss}$ (PFAM) LTM$_{KL,\lambda_k=100}$, (5) EoD$_{ss}$ (PFAM) LTM$_{KL,\lambda_k=\infty}$.

6.2. Subject-Based Analysis:

Table 6 presents the average performance of ensembles for the semi-supervised scenario obtained after self-update using ROI samples from trajectories stored in D, D_1 and D_2. The LTM size used corresponds to $\lambda_k = 25$ and 100 patterns. Modules 58 and 209 correspond to individuals of interest with good initial performance (pAUC (5%) ≥ 95). They are easy to detect with an EoD$_{ss}$ (PFAM) ($tpr \geq 50\%$), and to differentiate from non-target individuals

($fpr \leq 1\%$): These are typically *sheep*-like individuals in the Doddington zoo taxonomy. Results after learning D reveal the existence of 4 non-target individuals that are incorrectly detected more than 1% of the time (*wolves*) in both cases, corresponding to the 2.58% of the non-target individuals during tests. In contrast, EoDs$_{ss}$ 151 and 188 were selected because they initially provide poor performance (pAUC (5%) < 95%). EoD$_{ss}$ 151 corresponds to an individual that is difficult to detect by the system ($tpr < 50\%$), but is also difficult to impersonate ($fpr \leq 1\%$). The test at $t = 1$ reveals 5 wolves for this *goat*-like individual in the Doddington zoo taxonomy. The number of wolves corresponds to 3.23% of non-target individuals. EoD$_{188}$ corresponds to an individual which while being easy to detect by the system ($tpr \geq 50\%$), it is also easy to impersonate ($fpr > 1\%$). The test on D_1 reveals 32 wolves, corresponding to 20.65% of non-target individuals. Given the number of wolves, EoD$_{188}$ corresponds to a *lamb*-like individual.

Table 6. Average performance of the system for 4 individuals of interest over 10 independent experiments, after test on $D_1 \rightarrow D_2 \rightarrow D_3$. Two cases that initially provide a high level of performance correspond to EoDs with an initial pAUC (5%) ≥ 95% on D_1. Cases with initial performance that is poor are those with an initial pAUC (5%) < 95% on D_1.

Approach	**EoDs with good initial performance**		**EoDs with bad initial performance**	
	EoD$_{ss}$ 58 (*sheep*-like)	EoD$_{ss}$ 209 (*sheep*-like)	EoD$_{ss}$ 151 (*goat*-like)	EoD$_{ss}$ 188 (*lamb*-like)
EoD$_{ss}$ (PFAM), semi-supervised incremental learning, LTM_{KL}, $\lambda = 25$				
fpr (%) ↓	0.23 ±0.09 → 0.85 ±0.07 → 1.46 ±0.45	0.34 ±0.07 → 5.44 ±1.56 → 2.74 ±0.68	0.13 ±0.04 → 0.43 ±0.22 → 0.31 ±0.15	2.54 ±0.57 → 0.95 ±0.09 → 0.57 ±0.21
tpr (%) ↑	84.43 ±3.33 → 39.35 ±7.06 → 44.24 ±12.73	86.28 ±3.54 → 11.79 ±9.76 → 33.80 ±13.22	37.50 ±7.91 → 19.14 ±10.17 → 51.19 ±13.86	89.58 ±4.26 → 85.17 ±4.68 → 90.78 ±5.33
F$_1$ ↑	0.849 ±0.023 → 0.402 ±0.061 → 0.373 ±0.077	0.792 ±0.018 → 0.047 ±0.031 → 0.205 ±0.086	0.447 ±0.065 → 0.182 ±0.089 → 0.509 ±0.112	0.472 ±0.054 → 0.670 ±0.024 → 0.863 ±0.039
pAUC (5%) ↑	98.45 ±0.23 → 73.74 ±3.52 → 79.52 ±5.93	97.61 ±0.31 → 46.81 ±10.51 → 64.13 ±10.52	82.19 ±5.46 → 65.30 ±9.46 → 91.34 ±3.85	91.12 ±2.41 → 95.48 ±1.14 → 99.73 ±0.05
EoD$_{ss}$ (PFAM), semi-supervised incremental learning, LTM_{KL}, $\lambda = 100$				
fpr (%) ↓	0.23 ±0.09 → 0.86 ±0.09 → 1.62 ±0.39	0.34 ±0.07 → 0.46 ±0.07 → 1.10 ±0.26	0.13 ±0.04 → 0.25 ±0.14 → 0.26 ±0.15	2.54 ±0.57 → 1.18 ±0.20 → 0.31 ±0.10
tpr (%) ↑	84.43 ±3.33 → 35.44 ±8.10 → 51.16 ±14.32	86.28 ±3.54 → 88.33 ±3.33 → 98.10 ±0.71	37.50 ±7.91 → 27.17 ±12.63 → 48.15 ±13.56	89.58 ±4.26 → 89.88 ±3.09 → 93.70 ±1.74
F$_1$ ↑	0.849 ±0.023 → 0.353 ±0.066 → 0.384 ±0.093	0.792 ±0.018 → 0.793 ±0.023 → 0.802 ±0.037	0.447 ±0.065 → 0.274 ±0.119 → 0.498 ±0.112	0.472 ±0.054 → 0.667 ±0.032 → 0.920 ±0.013
pAUC (5%) ↑	98.45 ±0.23 → 74.58 ±3.54 → 80.44 ±6.34	97.61 ±0.31 → 97.16 ±0.27 → 99.59 ±0.11	82.19 ±5.46 → 68.64 ±9.32 → 91.39 ±3.86	91.12 ±2.41 → 96.39 ±0.48 → 99.72 ±0.05

Results for EoD$_{ss}$ 58 after updating on D_1 (testing on D_2) show a decline in pAUC (5%) performance for both λ_k values. However, the F_1 performance shows a greater decline for $\lambda_k = 100$, which reveals that D_1 contains some ROI samples that corrupt the facial model, and degrades the EoD$_{ss}$ (PFAM) accuracy. It can be seen however that some of these are filtered out by the KL selection strategy, given the higher performance with $\lambda_k = 25$. The overall results suggests that for this sheep-like individual, the performance can be maintained using small λ_k values.

The pAUC (5%) for EoD$_{ss}$ 209 after testing on D_2 also shows a decline in performance for $\lambda_k = 25$. Alto a small recovery is shown after testing on D_3, performance does not regain the same level due to the lack of representative validation data. On the other hand, an LTM with $\lambda_k = 100$ is shown to be able to maintain and improve the level of performance. This results suggest that sheep-like individuals benefit from higher λ_k values, and low λ_k values may lead to the corruption of the facial models. Given the results form EoDs 58 and 209, one can conclude that high values of λ_k ensure performance for sheep-like individuals, and individual-specific λ_k values should be estimated based on the evolution of specific EoDs.

With EoD$_{ss}$ 151, pAUC (5%) and F_1 performance declines after testing on D_2. This decline accentuated when $\lambda_k = 25$ patterns. Similarly to EoD$_{ss}$ 58, this trend reveals that D_2 contains some samples that corrupt this facial model. However, in this case, the system is benefits from higher λ_k values. Both EoDs show an increase in performance after testing on D_3, showing comparable performance in terms of F_1 and pAUC (5%) for both λ_k values. This reveals that, in the presence of corrupted data, goat-like individuals benefit from greater LTM sizes.

EoD$_{ss}$ 188 presents a constant increase in pAUC (5%) and F_1 performance. Despite the number of incorrect updates produced by multiple wolves, the fpr decreases after each self-update. This suggests that lamb-like individuals benefit from diverse samples from these updates as well. Similar performance is achieved by the EoD$_{ss}$ (PFAM) for small or large λ_k values.

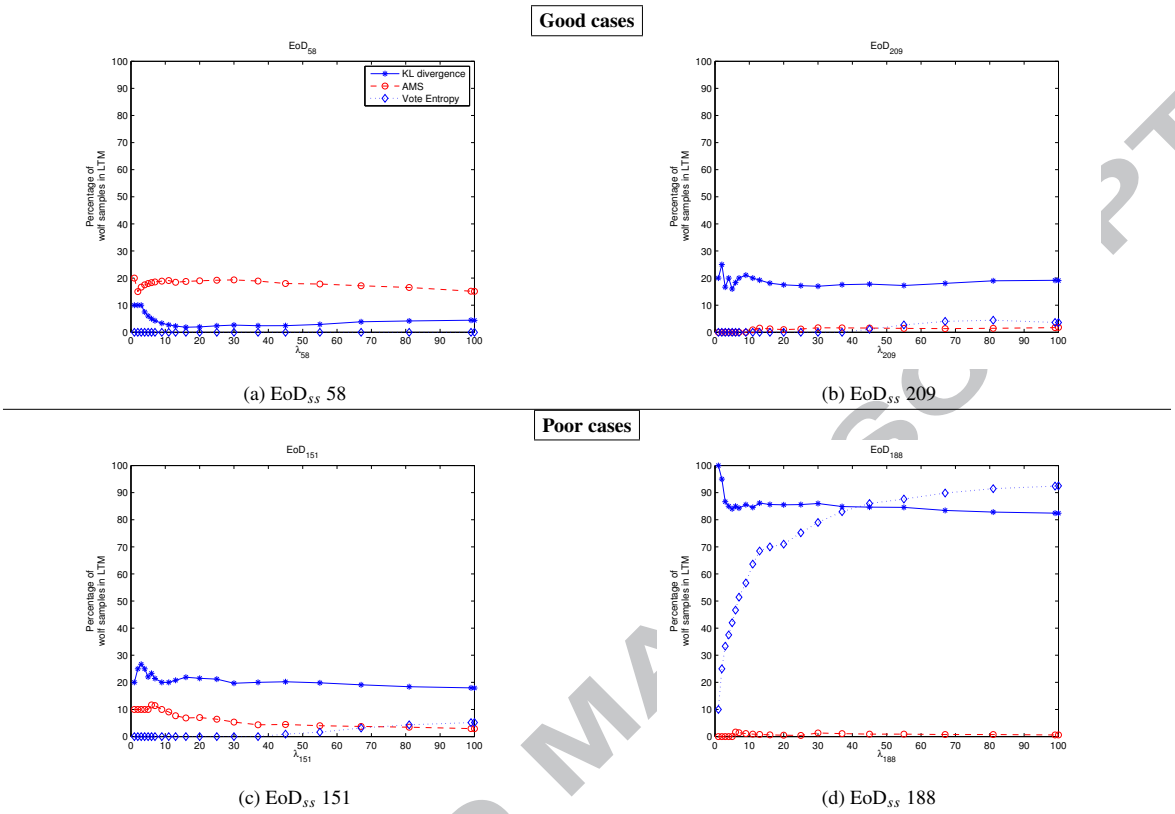
It is well known that samples from wolf-like individuals negatively affect the fpr of EoDs, and by definition, the effect is more pronounced if the EoD corresponds to a lamb-like individual. Figure 9 presents the percentage of samples from wolf-like individuals selected by KL divergence, Average Margin Sampling (AMS) and Vote Entropy (VE), corresponding to the analyzed individuals of interest. Different sizes of LTM were tested following the exponential scale $\lambda_k = \lceil e^x \rceil$, where $x = 0, 0.2, 0.4, ...4.6$ [6]. Results show no clear tendency for the good cases, as shown in the graphs in Figure 9**a** and 9**b**. For these two sheep-like individuals (EoD_{58} and EoD_{209}) the AMS and KL divergence select a similar amount of samples from wolf-like individuals in different cases. As shown in Figure 9**c**, the KL divergence retrieves more samples from wolf-like individuals when the EoD corresponds to a goat-like individual. Finally, Figure 9**d** shows that for lamb-like individuals, the KL divergence is specially effective in finding samples from wolf-like individuals given a small LTMs ($\lambda < 50$). In summary, the KL divergence is useful in cases with poor initial performance (lamb-like and goat-like individuals), and with only small LTM sizes.

6.3. Trajectory-Based Analysis:

Fig. 10 presents the accumulation curves showing the positive predictions produced by the EoDs in response to target and non-target trajectories in D_1 (replication 1). The detection and update thresholds estimated on the validation set are also depicted on the graphs. As can be observed in this case, the accumulative curves corresponding to the two sheep-like individuals surpass both detection and update thresholds. And the upper envelope for non-target individuals is always below the thresholds, which means that none of the negative trajectories was incorrectly assigned to the target individual. EoDs for IDs 58 and 209 both exhibit a correct detection through D_1, allowing for the correct rejection of all negative trajectories in D_1.

The accumulative curves for EoD$_{ss}$ 151 and 188 for the same replication are also presented in Fig. 10. While the goat-like individual (ID 151) remains hard to detect, the lamb-like individual (ID 188) is impersonated by wolves present in D_1. Results suggest that the level of Γ_k (in Eq. 3) should be different for each type of individual. For instance, sheep-like individuals require smaller Γ_k values, and lamb-like individuals require larger Γ_k values. On the other hand, goat-like individuals may require a reduction of the detection threshold.

[6] Note that $\lambda_k = \lceil e^{4.6} \rceil = 100$, the maximum λ_k considered in experiments.



(a) EoD$_{ss}$ 58 (b) EoD$_{ss}$ 209

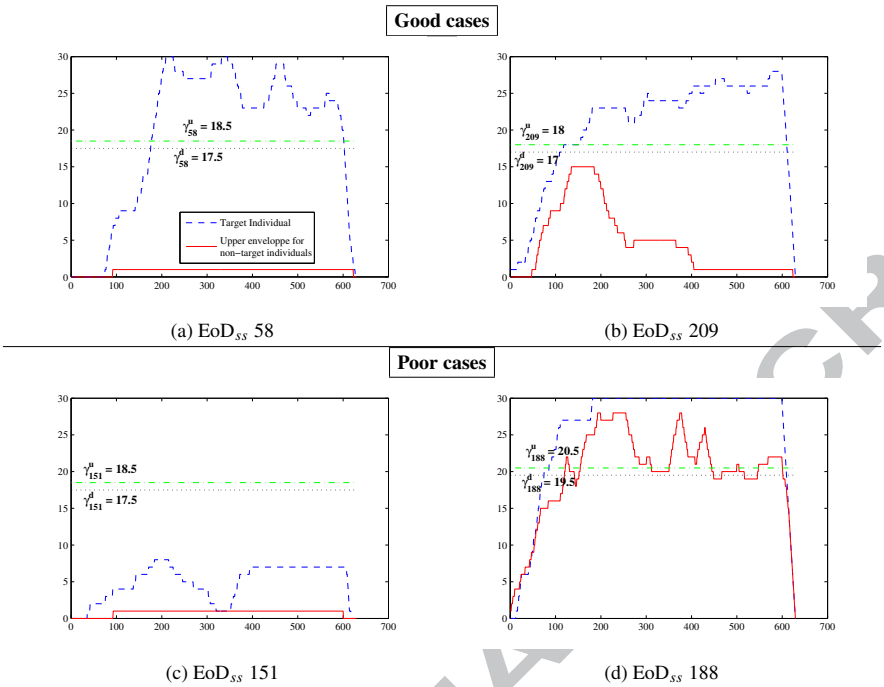
(c) EoD$_{ss}$ 151 (d) EoD$_{ss}$ 188

Figure 9. Percentage of wolf-like individuals in LTMs for the EoDs in the subject-based analysis.

Fig. 11 shows the ROC curves for the overall system at the decision level. These curves were obtained by varying the decision thresholds on the accumulation curves produced by target and non-target trajectories in D_3 (Fig. 10). It shows the high level of discrimination achieved with these EoD$_{ss}$ (PFAM) at the decision fusion system after two updates, by accumulating evidence. Even though the selected update threshold γ^{μ}_{188} permitted some false updates after testing on D_1, the EoD$_{ss}$ increased its level of discrimination, achieving only correct updates after testing on D_3.

Table 7 shows the average number of correct and incorrect trajectories detected by the selected EoD$_{ss}$ (PFAM) at the decision level. The benefit of accumulating predictions over a trajectory becomes evident for these EoDs by comparing the tpr and fpr before and after decision fusion. For instance, EoD$_{ss}$ 58 presents a $tpr = 84.43\%$ and $fpr = 0.23\%$ using transaction-based decisions (see Table 6), but using the whole trajectories in making the decision it produces a $tpr = 100\%$ and $fpr = 0\%$. This means that every time a target trajectory from D_1 was presented to the system, it was correctly detected by the corresponding EoD$_{ss}$, and all non-target trajectories were correctly rejected. A similar behavior is shown by EoD$_{ss}$ 209, which confirms that EoDs for sheep-like individuals may achieve a high level of discrimination with the proposed approach.

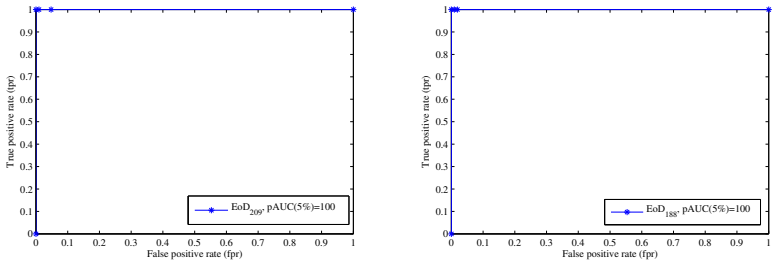
Performance is also seen to be increasing in EoDs for individuals 151 and 188, the tpr growing considerably and



(a) EoD$_{ss}$ 58 (b) EoD$_{ss}$ 209

(c) EoD$_{ss}$ 151 (d) EoD$_{ss}$ 188

Figure 10. Accumulated positive prediction curves produced by the EoD$_{ss}$ (PFAM) of target vs. the non-target individuals, after training on D (testing on D_1), along with detection and update thresholds.



(a) ROC curve for EoD$_{ss}$ 209 (b) ROC curve for EoD$_{ss}$ 188

Figure 11. ROC curves for EoDs 209 (a) and 188 (b) at the decision fusion level, test on D_3, experiment trial 1. In both cases the final curves are perfect after two updates, even though the EoD$_s$ s 188 was updated 5 times with non-target trajectories in D_1.

Table 7. The average performance of the overall system following a trajectory-based analysis. The number of target trajectories is 10, and the number of non-target trajectories is 1050 for the 10 replications after test on D_1. Results are produced by the system EoD$_{ss}$ (PFAM) LTM$_{KL,\lambda_k=100}$, for the 4 cases in analysis.

Measure	**EoDs with good initial performance**		**EoDs with bad initial performance**	
	EoD$_{ss}$ 58	EoD$_{ss}$ 209	EoD$_{ss}$ 151	EoD$_{ss}$ 188
tpr	100.00	100.00	50.00	100.00
fpr	0.00	0.00	0.00	0.86
F$_1$	1.00	1.00	0.667	0.6896
pAUC (5%)	100.00	100.00	51.25	91.40

simultaneously reducing the fpr to about 0%. Moreover, using the decisions based on trajectories, the number of wolves is reduced from 32 to only 5 for the $wolf$-like individual 188. This suggests that the EoDs for both goat- and lamb-like individuals may also benefit from the proposed trajectory-based decision scheme.

Table 8. IDs corresponding to the trajectories in FIA that surpassed the update threshold and were used for updating the selected EoDs on different replications (r) of the experiment (EoD_{ss}, $\text{LTM}_{KL,\lambda_k=100}$). Bold numbers correspond to trajectories used for correct updates, and conflicts are marked with a box around the ID of the trajectory.

Rep.	EoD_{ss} **58**	EoD_{ss} **151**	EoD_{ss} **188**	EoD_{ss} **209**	EoD_{ss} **58**	EoD_{ss} **151**	EoD_{ss} **188**	EoD_{ss} **209**
	Update trajectories in D_1				**Update trajectories in D_2**			
r=1	ID=**58**	ID=-	ID=6,60,186,**188**,193,224	ID=**209**	ID=-	ID=-	ID=**188**	ID=**209**
r=2	ID=**58**	ID=-	ID=**188**,224	ID=**209**	ID=-	ID=-	ID=104,**188**	ID=-
r=3	ID=**58**	ID=**151**	ID=**188**	ID=**209**	ID=**58**	ID=-	ID=-	ID=**209**
r=4	ID=**58**	ID=-	ID=**188**	ID=**209**	ID=-	ID=-	ID=-	ID=**209**
r=5	ID=**58**	ID=-	ID=**188**,224	ID=**209**	ID=**58**,134	ID=-	ID=**188**	ID=**209**
r=6	ID=**58**	ID=**151**	ID=**188**	ID=**209**	ID=**58**	ID=**151**	ID=104,**188**	ID=**209**
r=7	ID=**58**	ID=-	ID=**188**,224	ID=**209**	ID=-	ID=-	ID=**188**	ID=**209**
r=8	ID=**58**	ID=**151**	ID=**188**	ID=**209**	ID=**58**	ID=-	ID=104,122,**188**	ID=**209**
r=9	ID=**58**	ID=**151**	ID=**188**,224	ID=**209**	ID=-	ID=**151**	ID=104,**151**,153,**188**	ID=**209**
r=10	ID=**58**	ID=**151**	ID=**188**	ID=**209**	ID=**58**	ID=**151**,174	ID=104,**188**	ID=**209**

Table 8 provides further details on the updates over replications 1 to 10 for selected EoDs with $\text{LTM}_{KL,\lambda_k=100}$. After testing on D_1, EoD_{ss} 58 is always correctly and never incorrectly updated. However, after testing on D_2, only 50% of correct updates were performed, and an incorrect update was present at replication 5. This phenomenon is explained by the drop in performance due to the existence of ROI samples on D_1 that corrupted the facial model, as discussed earlier. A similar trend is presented by EoD_{ss} 151, dropping from 5 correct updates on D_1, to 3 correct and 1 incorrect updates. However, at replication 9, the correct update is discarded due to the conflict with EoD_{ss} 188. The facial model for individual 188 was correctly updated on all replications after testing on D_1, but 9 wrong updates were also performed on five of the replications. After test on D_2 the number of correct updates dropped to 8, and incorrect updates dropped to 8, in 5 of the replications. And one of the incorrect updates was discarded due to the conflict detected with EoD_{ss} 151. A different trend is shown by EoD_{ss} 209, for which a reduction in the number of correct updates was only seen at replication 2, and never presented a wrong update.

7. Conclusion

In this paper, an adaptive MCS is proposed for video-to-video FR, where the face of each target individual is modeled using an ensemble of 2-class classifiers. During operations, this new system integrates information from a face tracker and individual-specific ensembles for robust spatio-temporal recognition and for efficient self-update of facial models. The tracker defines a facial trajectory for each individual that appears in a video. Spatio-temporal FR occurs if the number of positive predictions accumulated along a trajectory surpass the detection threshold for an individual-specific ensemble. A higher update threshold allows the system to determine if the trajectory incorporates enough confidence for self-update of facial models. To update a facial model, all target samples extracted from the trajectory are combined with non-target samples selected from the cohort and universal models. Facial models are

updated using a learn-and-combine strategy to avoid knowledge corruption that can occur during self-update with an incremental learning classifier. In addition, a memory management strategy based on Kullback-Leibler divergence is used to rank and select the most relevant target and non-target reference ROI samples for validation.

Proof of concept validation has been performed on the CMU-FIA video dataset with a particular realisation of the proposed system.The individual-specific EoDs are formed with of ARTMAP neural network classifiers generated using a DPSO incremental learning strategy, where classifiers are combined using BC. Transaction-level results indicate that the proposed adaptive MCS improved pAUC (5%) by about 8% over the system that do not perform self-update. It provides an average performance comparable to the same system that performs supervised update of facial models with all relevant trajectories. Subject-level analysis reveals that facial models from sheep- and goat-like individuals benefit from using a large LTM, while *lamb*-like individuals present similar performance with large or small LTM sizes. This is a consequence of the capacity of the KL divergence to select samples from wolf-like individuals, which are more numerous for EoDs corresponding to *lamb*-like individuals. For trajectory-level analysis shown by the accumulated decisions, the system increases discrimination and robustness compared to transaction-level decisions. In all the cases that were analyzed, the individual-specific EoDs were able to simultaneously increase the overall pAUC (5%), tpr and F_1 measures, and reduce the fpr. Finally, an analysis of the updates achieved by the system shows that by virtue of the increased discrimination, it presented a low number of incorrect updates even with the large number of non-target trajectories presented to the system during simulations.

In this paper, trajectories define the design samples used for (re)enrollment (supervised learning) and update (supervised or unsupervised learning) of facial models encoded in a video-to-video FR system. The proposed MCS has been characterized using data that exhibits a gradual pattern of changes over different capture sessions. Future research should analyze performance under abrupt patterns of change, as seen in sharp variations of illumination and face pose. A dynamic adaptation of the fusion functions of the ensembles to these scenarios may allow a better exploitation of the availability of abundant operational data. Since the proportion of target to non-target ROIs captured in practice is imbalanced is imbalanced, and the level of imbalance changes over time, classifier ensembles should be selected dynamically according to the context to improve performance. Regarding resource management, the exploration of pruning strategies for ensembles is another open issue. In practice, the system should exploit internal knowledge (age, performance relevance, etc.) to remove some older or redundant classifiers over time. With respect to the KL based LTM management scheme, it might be characterized on different applications of adaptive ensembles, like iris or gait recognition, signature verification, or in general object recognition. Finally, the system may also benefit from knowledge of ROI samples from wolf- and goat-like individuals, and the amount of validation samples stored in LTM may be optimized per individual. This could allow to select target and non-target ROI samples that lead to more discriminant individual-specific EoDs.

Acknowledgment

This work was partially supported by the Natural Sciences and Engineering Research Council of Canada, and the Defence Research and Development Canada Centre for Security Science Public Security Technical Program. This work was also supported by the Program for the Improvement of the Professoriate of the Secretariat of Public Education, Mexico, and the Mexican National Council for Science and Technology.

References

[1] F. Li, H. Wechsler, Open set face recognition using transduction, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (11) (2005) 1686–97.
[2] M. De-la Torre, E. Granger, P. V. W. Radtke, R. Sabourin, D. O. Gorodnichy, Incremental update of biometric models in face-based video surveillance, in: Proceedings on International Joint Conference on Neural Networks, Brisbane, Australia, 2012, pp. 1–8.
[3] F. Matta, J.-L. Dugelay, Person recognition using facial video information: a state of the art, Journal of Visual Languages and Computing 20 (3) (2009) 180–7.
[4] V. Despiegel, S. Gentric, J. Fondeur, Border control: From technical to operational evaluation, in: Proceedings on International Biometric Performance Testing Conference, Gaithersburg, Maryland, US, 2012.
[5] H. K. Ekenel, J. Stallkamp, R. Stiefelhagen, A video-based door monitoring system using local appearance-based face models, Computer Vision Image Understanding 114 (5) (2010) 596–608.
[6] S. Zhou, V. Krueger, R. Chellappa, Probabilistic recognition of human faces from video, Computer Vision and Image Understanding 91 (1-2) (2003) 214 – 45.
[7] R. Polikar, L. Udpa, S. S. Udpa, V. Honavar, Learn++: An Incremental Learning Algorithm for MLP Networks, IEEE Transactions Systems, Man and Cybernetics 31 (4) (2001) 497–508.
[8] R. Singh, M. Vatsa, A. Ross, A. Noore, Biometric classifier update using online learning: A case study in near infrared face verification, Image and Vision Computing 28 (2010) 1098–1105.
[9] J.-F. Connolly, E. Granger, R. Sabourin, Evolution of heterogeneous ensembles through dynamic particle swarm optimization for video-based face recognition, Pattern Recognition 45 (7) (2012) 2460 – 2477.
[10] C. Pagano, E. Granger, R. Sabourin, D. O. Gorodnichy, Detector ensembles for face recognition in video surveillance, in: Proceedings on International Joint Conference on Neural Networks, Brisbane, Australia, 2012, pp. 1–8.
[11] A. Rattani, Adaptive biometric system based on template update procedures, Ph.D. thesis, University of Cagliari (2010).
[12] F. Roli, G. L. Marcialis, Semi-supervised pca-based face recognition using self-training, in: Proceedings on Joint International Association for Pattern Recognition - International Workshop on Structural and Syntactical Pattern Recognition and Statistical Techniques in Pattern Recognition, Vol. 4109, Springer, Hong Kong, China, 2006, pp. 560–568.
[13] A. Franco, D. Maio, D. Maltoni, Incremental template updating for face recognition in home environments, Pattern Recognition 43 (8) (2010) 2891 – 903.
[14] F. Roli, L. Didaci, G. Marcialis, Template co-update in multimodal biometric systems, in: Proceedings on International Conference on Biometrics, Vol. 4642, Seoul, Korea, 2007, pp. 1194 – 202.
[15] A. Merati, N. Poh, J. Kittler, Extracting discriminative information from cohort models, in: Proceedings on IEEE International Conference on Biometrics: Theory Applications and Systems, 2010, pp. 1 –6.
[16] P. Hart, The condensed nearest neighbor rule (correspondence), IEEE Transactions on Information Theory 14 (3) (1968) 515 – 516.
[17] A. Kachites McCallum, K. Nigam, Employing EM and pool-based active learning for text classification, in: Proceedings on International Conference on Machine Learning, San Francisco, USA, 1998, pp. 350–8.
[18] A. Yilmaz, O. Javed, M. Shah, Object tracking: A survey, ACM Computing Surveys 38 (4) (2006) 1–45.
[19] D. Tax, R. Duin, Growing a multi-class classifier with a reject option, Pattern Recognition 29 (10) (2008) 1565 – 70.
[20] A. K. Jain, A. Ross, Learning user-specific parameters in a multibiometric system, in: Proceedings on International Conference on Image Processing, 2002, pp. 57–60.
[21] B. Kamgar-Parsi, W. Lawson, B. Kamgar-Parsi, Toward development of a face recognition system for watchlist surveillance, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (10) (2011) 1925 – 37.
[22] Y. Zhang, A. Martinez, From stills to video: Face recognition using a probabilistic approach, in: Proceedings on Workshop Conference on Computer Vision and Pattern Recognition, 2004, p. 78. doi:10.1109/CVPR.2004.75.
[23] X. Liu, T. Cheng, Video-based face recognition using adaptive Hidden Markov Models, in: Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 1, Los Alamitos, CA, USA, 2003, pp. 340 – 5.
[24] Y.-C. Chen, V. M. Patel, S. Shekhar, R. Chellappa, P. J. Phillips, Video-based face recognition via joint sparse representation, in: Proceedings on International Conference on Automatic Face and Gesture Recognition, Shangai, China, 2013.
[25] B. Freni, G. L. Marcialis, F. Roli, Template selection by editing algorithms: A case study in face recognition, in: Proceedings on Joint International Association of Pattern Recognition, Vol. 5342, Orlando, USA, 2008, pp. 745–754.
[26] D. D. Lewis, J. Catlett, Heterogeneous uncertainty sampling for supervised learning, in: Proceedings on International Conference on Machine Learning, Morgan Kaufmann, 1994, pp. 148–156.
[27] W. Liu, J. C. Principe, HaykinSimon, Kernel Adaptive Filtering: A Comprehensive Introduction, Wiley, 2010.
[28] T. Scheffer, C. Decomain, S. Wrobel, Active Hidden Markov models for information extraction, in: Proceedings on International Conference in Advances in Intelligent Data Analysis, Vol. 2189, Berlin, Germany, 2001, pp. 309–18.

[29] C. E. Shannon, A mathematical theory of communication, The Bell Systems Technical Journal 27 (1948) 379–423, 623–656.
[30] I. Dagan, S. Engelson, Committee-based sampling for training probabilistic classifiers, in: Proceedings on International Conference on Machine Learning, San Francisco, USA, 1995, pp. 150–7.
[31] E. Tang, P. Suganthan, X. Yao, An analysis of diversity measures, Machine Learning 65 (1) (2006) 247 – 71.
[32] X. Guo, Y. Yin, C. Dong, G. Yang, G. Zhou, On the class imbalance problem, in: Proceedings on International Conference on Natural Computation, Vol. 4, Piscaatway, NJ, USA, 2008, pp. 192 – 201.
[33] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, F. Herrera, A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches, IEEE Transactions on Systems, Man and Cybernetics 42 (2011) 463–484.
[34] I. Tomek, Two modifications of cnn, IEEE Transactions on Systems, Man and Cybernetics 6 (11) (1976) 769 –772.
[35] P. Flach, E. Matsubara, On classification, ranking, and probability estimation, in: Proceedings on Probabilistic, Logical and Relational Learning - A Further Synthesis, no. 07161 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2008, pp. 1–10.
[36] F. Roli, L. Didaci, G. L. Marcialis, Adaptive biometric systems that can improve with use, in: N. R. V. Govindaraju (Ed.), Advances in Biometrics: Sensors, Systems and Algorithms, Springer, 2008, pp. 447–471.
[37] K. Okada, L. Kite, C. von der Malsburg, An adaptive person recognition system, in: Proceedings on IEEE International Workshop on Robot and Human Interactive Communication, Piscataway, NJ, USA, 2001, pp. 436–41.
[38] A. Rattani, G. Marcialis, F. Roli, Capturing large intra-class variations of biometric data by template co-updating, in: Proceedings on IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Piscataway, NJ, USA, 2008, pp. 1–6.
[39] A. Rattani, B. Freni, G. L. Marcialis, F. Roli, Template update methods in adaptive biometric systems: A critical review, in: Lecture Notes in Computer Science (included Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 5558, Alghero, Italy, 2009, pp. 847 – 856.
[40] I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, T. S. Huang, Semisupervised learning of classifiers: Theory, algorithms, and their application to human-computer interaction, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (12) (2004) 1553–1567.
[41] K. Lu, Z. Ding, J. Zhao, Y. Wu, A novel semi-supervised face recognition for video, in: Proceedings of the International Conference on Intelligent Control and Information Processing, 2010, pp. 313–316.
[42] L. Didaci, F. Roli, Using co-training and self-training in semi-supervised multiple classifier systems, in: Lecture Notes on Computer Sciences, Vol. 4109, Hong Kong, China, 2006, pp. 522 – 530.
[43] N. El Gayar, S. A. Shaban, S. Hamdy, Face recognition with semi-supervised learning and multiple classifiers, in: Proceedings on World Scientific Engineering Academy and Society International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics, USA, 2006, pp. 296–301.
[44] G. Yu, G. Zhang, C. Domeniconi, Z. Yu, J. YouZ, Semi-supervised classification based on random subspace dimensionality reduction, Pattern Recognition 45 (3) (2012) 1119 – 1135.
[45] R. Hewitt, S. Belongie, Active learning in face recognition: Using tracking to build a face model, in: Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, NY, United states, 2006, p. 157.
[46] T. Fawcett, An introduction to roc analysis, Pattern Recognition Letters 27 (8) (2006) 861–874.
[47] L. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms, Wiley, 2004.
[48] R. Goh, L. Liu, X. Liu, T. Chen, The CMU Face In Action Database, in: Analysis and Modelling of Faces and Gestures, Carnegie Mellon University, 2005, pp. 255–263.
[49] P. Viola, M. Jones, Robust real-time face detection, International Journal of Computer Vision 2 (57) (2004) 137–154.
[50] G. R. Bradski, Computer vision face tracking for use in a perceptual user interface, Intel Technology Journal Q2 (1998) 1–15.
[51] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (7) (2002) 971–87.
[52] C. P. Lim, R. F. Harrison, Probabilistic fuzzy artmap: An autonomous neural network architecture for bayesian probability estimation, in: Proceedings International Conference on Artificial Neural Networks, 1995, pp. 148–153.
[53] W. Khreich, E. Granger, A. Miri, R. Sabourin, Iterative Boolean Combination of classifiers in the ROC space: An application to anomaly detection with HMMs, Pattern Recognition 43 (8) (2010) 2732 – 52.
[54] G. Doddington, W. Liggett, A. Martin, M. Przybocki, D. Reynolds, Sheep, goats, lambs and wolves: A statistical analysis of speaker performance, in: Proceedings on International Conference on spoken language processing, 1998, pp. 1351–1354.
[55] I. Oh, C. I. Suen, A class-modular feedforward neural network for handwritting recognition, Pattern Recognition 35 (2002) 229–244.

Appendix A. Synthetic Experiment on Relevance Measures

Two synthetic 2-class problems were designed to characterize the relevance measures in the 1D space. Fig. A.12 shows the original probability distributions used to generate the data for experiments. The central Gaussian distribution in both problems generates the positive samples, with a center of mass $\mu_2 = 0.5$. The centers of mass of the negative Gaussian distributions in Fig. A.12 (a) are $\mu_1 = 0.2$ and $\mu_3 = 0.8$, and in Fig. A.12 (b) the negative samples are randomly drawn from the 1D space according to a uniform distribution. All Gaussian distributions are characterized by a fixed variance of $\sigma = 0.01$. An ensemble of 7 PFAM classifiers has been trained for both problems

on a balanced training set. A learning strategy based on DPSO is used for generation of base classifiers and co-jointly optimize all PFAM parameters, as proposed in [9]. Classifier fusion is performed using BC.

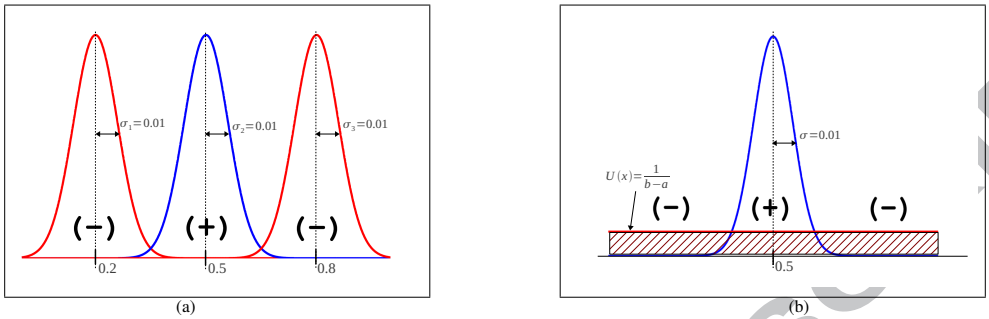


Figure A.12. Data distributions used to generate the training data for problems 1 (a) and 2 (b). In both figures the Gaussian distribution at the center generates the positive (+) samples, and the left and right distributions generate the negative (-) samples.

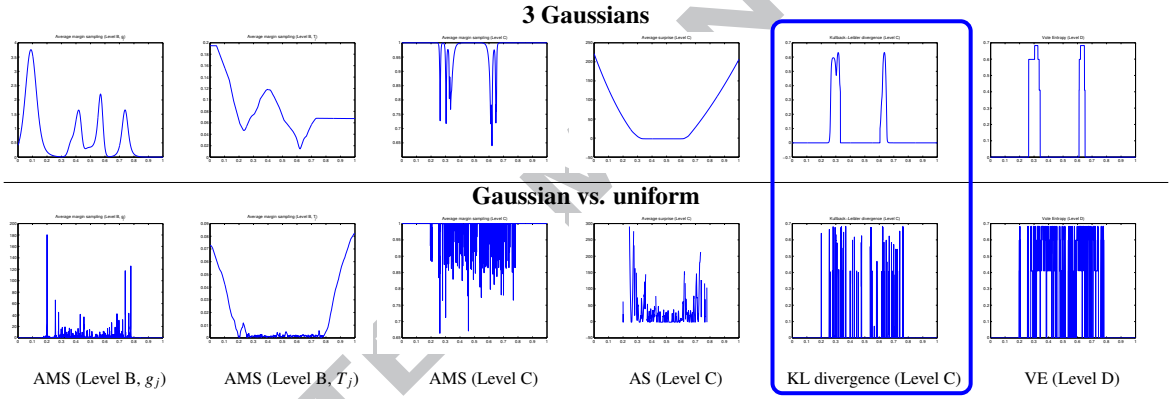


Figure A.13. Value of relevance measures obtained over the input space with an ensemble of 2-class PFAM classifiers for the *3 Gaussians* (top) and *Gaussian vs. uniform* (bottom) problems. From left to right, average margin sampling (AMS) at level B on g_j, AMS at level B on T_j, AMS at score level, average surprise (AS) at score level, Kullback-Leibler (KL) divergence at score level, and vote entropy (VE) at prediction level.

The value of relevance measures for the PFAM ensembles corresponding to both problems are presented in Fig. A.13. Whereas the extension of surprise (average surprise) follows a shape similar to that of the surprise estimated for a single model, other measures focus on the overlapping of data distribution zones. Vote entropy uses decision level information (level D from Fig. 2), and hence presents a lower resolution (e.g. fewer ranking values). While KL divergence and average margin sampling both present a good resolution, the smoothness of curves for KL divergence, provide a better representation of the overlapping area.

Appendix B. Full Update Table

Table B.9 presents the details of the updates for the 10 independent replications of the experiment for the individuals of interest enrolled to the system, with the EoD$_{ss}$ (PFAM) LTM$_{KL,\lambda_k=100}$.

Table B.9. IDs corresponding to the trajectories that surpassed the update threshold and were used for updating the selected EoDs on different replications (r) of the experiment (EoD_{ss}, $\text{LTM}_{KL,\lambda_k=100}$). Bold numbers correspond to trajectories selected for correct updates, and conflicts are marked with a box around the ID of the trajectory.

Replic.	EoD_{ss} 2	Mod. 58	Mod. 72	Mod. 92	Mod. 147	Mod. 151	Mod. 176	Mod. 188	Mod. 190	Mod. 209
Update trajectories in D_1										
r=1	ID=**2**	ID=**58**	ID=**72**	ID=-	ID=**147**	ID=-	ID=-	ID=6,60,186,**188**,193,224	ID=-	ID=**209**
r=2	ID=**2**	ID=**58**	ID=**72**,179	ID=**92**	ID=-	ID=-	ID=-	ID=**188**,224	ID=**190**	ID=**209**
r=3	ID=**2**	ID=**58**	ID=**72**	ID=**92**,235	ID=**147**	ID=**151**	ID=-	ID=**188**	ID=136,**190**	ID=**209**
r=4	ID=**2**	ID=**58**	ID=**72**	ID=**92**,235	ID=**147**	ID=-	ID=-	ID=**188**	ID=-	ID=**209**
r=5	ID=**2**	ID=**58**	ID=**72**,179	ID=**92**	ID=**147**	ID=-	ID=-	ID=**188**,224	ID=-	ID=**209**
r=6	ID=**2**	ID=**58**	ID=**72**	ID=**92**	ID=**147**	ID=**151**	ID=-	ID=**188**	ID=**190**	ID=**209**
r=7	ID=-	ID=**58**	ID=**72**,179	ID=-	ID=**147**	ID=-	ID=-	ID=**188**,224	ID=**190**	ID=**209**
r=8	ID=**2**	ID=**58**	ID=**72**,179	ID=**92**	ID=**147**	ID=**151**	ID=**176**	ID=**188**	ID=-	ID=**209**
r=9	ID=**2**	ID=**58**	ID=**72**	ID=-	ID=**147**,222	ID=**151**	ID=**176**	ID=**188**,224	ID=**190**	ID=**209**
r=10	ID=**2**	ID=**58**	ID=**72**,179	ID=-	ID=**147**	ID=**151**	ID=**176**	ID=**188**	ID=-	ID=**209**
Update trajectories in D_2										
r=1	ID=220	ID=-	ID=[136],175	ID=-	ID=**147**	ID=-	ID=-	ID=**188**	ID=[136]	ID=**209**
r=2	ID=220	ID=-	ID=179	ID=-	ID=-	ID=-	ID=-	ID=104,**188**	ID=99,127,136,**190**,201	ID=-
r=3	ID=-	ID=**58**	ID=-	ID=-	ID=**147**	ID=-	ID=-	ID=-	ID=127,136,**190**	ID=**209**
r=4	ID=-	ID=-	ID=148	ID=-	ID=**147**	ID=-	ID=-	ID=-	ID=136	ID=**209**
r=5	ID=-	ID=**58**,134	ID=23,148,175	ID=-	ID=**147**	ID=-	ID=-	ID=**188**	ID=136	ID=**209**
r=6	ID=220	ID=**58**	ID=-	ID=-	ID=**147**	ID=**151**	ID=**176**	ID=104,**188**	ID=136,**190**	ID=**209**
r=7	ID=-	ID=-	ID=-	ID=-	ID=**147**	ID=-	ID=**176**	ID=**188**	ID=136,**190**,197	ID=**209**
r=8	ID=-	ID=**58**	ID=-	ID=-	ID=**147**	ID=-	ID=**176**	ID=104,122,**188**	ID=134,136	ID=**209**
r=9	ID=-	ID=-	ID=134	ID=-	ID=**147**	ID=[**151**]	ID=**176**	ID=104,[151],153,**188**	ID=99,136,**190**	ID=**209**
r=10	ID=-	ID=**58**	ID=94	ID=-	ID=**147**	ID=**151**,174	ID=-	ID=104,**188**	ID=136	ID=**209**
Update trajectories in D_3										
r=1	ID=**2**	ID=-	ID=136	ID=37,**92**,134,148	ID=-	ID=140,**151**	ID=3	ID=**188**	ID=148,**190**	ID=**209**
r=2	ID=**2**,108	ID=-	ID=179	ID=**92**,134,148	ID=-	ID=[**151**]	ID=140,[151]	ID=**188**	ID=136,**190**	ID=**209**
r=3	ID=**2**	ID=**58**	ID=179	ID=134,[148]	ID=-	ID=92,107,**151**,202	ID=-	ID=**188**	ID=136,[148],**190**	ID=**209**
r=4	ID=**2**	ID=-	ID=179	ID=**92**,134,148	ID=-	ID=[**151**]	ID=108,[151],177	ID=-	ID=47,84,136,**190**	ID=**209**
r=5	ID=**2**	ID=**58**	ID=-	ID=134,[148]	ID=-	ID=[**151**]	ID=[151]	ID=**188**	ID=47,84,136,[148],**190**	ID=**209**
r=6	ID=-	ID=**58**	ID=[37],179	ID=[37],**92**,134,[148]	ID=-	ID=**151**	ID=**176**,177	ID=**188**	ID=12,47,58,84,136,[148],**190**	ID=**209**
r=7	ID=**2**	ID=-	ID=179	ID=37,**92**,134,[148]	ID=-	ID=**151**	ID=-	ID=**188**	ID=136,[148],**190**	ID=**209**
r=8	ID=**2**	ID=-	ID=37,179	ID=**92**	ID=-	ID=107,**151**	ID=**176**	ID=**188**	ID=84,136,**190**	ID=**209**
r=9	ID=**2**	ID=-	ID=84,134,148	ID=58,**92**	ID=-	ID=**151**	ID=-	ID=**188**	ID=136,**190**	ID=**209**
r=10	ID=**2**	ID=**58**	ID=37,179,197	ID=148	ID=-	ID=11,**151**	ID=140	ID=-	ID=84,136,**190**	ID=-