Boosted Online Learning for Face Recognition

David Masip, *Member, IEEE,* Àgata Lapedriza and Jordi Vitrià

***Abstract*—Face recognition applications commonly suffer from three main drawbacks: a reduced training set, information lying in high dimensional subspaces, and the need to incorporate new people to recognize. In the recent literature, the extension of a face classifier in order to include new people in the model has been solved using online feature extraction techniques. The most successful approaches of those are the extensions of the Principal Component Analysis (PCA) or the Linear Discriminant Analysis (LDA). In the current paper a new Online Boosting algorithm is introduced: a face recognition method that extends a boosting-based classifier by adding new classes while avoiding the need of retraining the classifier each time a new person joins the system. The classifier is learnt using the Multi-Task Learning principle where multiple verification tasks are trained together sharing the same feature space. The new classes are added taking advantage of the structure learnt previously, being the addition of new classes not computationally demanding. The present proposal has been (experimentally) validated with two different facial data sets by comparing our approach with the current state-of-the-art techniques. The results show that the proposed Online Boosting algorithm fares better in terms of final accuracy. In addition, the global performance does not decrease drastically even when the number of classes of the base problem is multiplied by 8.**

***Index Terms*—Online Learning, Incremental Learning, Face Recognition, Multi Task Learning, Small Sample Size Problem.**

I. Introduction

Face recognition problem can be stated as a machine learning process where we receive as input a high-dimensional data vector $\mathbf{x} \in \mathbb{R}^D$ (corresponding to the $n_1 \times n_2 = D$ face pixel image), and we must provide the identity or class membership $c \in \{C_1, \ldots, C_K\}$ of the subject. In real-world applications the number of training samples available from each class is usually limited, and the data dimensionality is large, making the estimation of the classifier parameters more inaccurate. This problem is known as *the curse of dimensionality* [1], which exponentially relates the number of samples needed to model an object with the dimensionality of its representative feature vector.

On the other hand, the number of classes in face recognition is large, and we often need to extend previously trained classifiers to recognize new people that joins the group. In this context most of the classic machine learning methods are not suitable for the face recognition task.

• David Masip is with the Universitat Oberta de Catalunya, Rambla del Poblenou 156, 08018 Barcelona, Spain. E-mail: dmasipr@uoc.edu

• Àgata Lapedriza is with Computer Vision Center, Computer Science Department, Universitat Autònoma de Barcelona, Edifici O Bellaterra, Barcelona 08193, Spain. E-mail: agata@cvc.uab.es

• Jordi Vitrià is with Computer Vision Center, Universitat Autònoma de Barcelona, Edifici O Bellaterra, Barcelona 08193, Spain and Departament de Matemàtica Aplicada i Anàlisi, Universitat de Barcelona, Gran Via de les Corts Catalanes 585, Barcelona 08007. E-mail: jordi@cvc.uab.es

Several face recognition algorithms found on the literature focus on the problem of classification in high dimensional subspaces. Usually a feature extraction step is performed in order to reduce the problem complexity, and then the Nearest Neighbor classifier (NN) is applied on the reduced space. Following this framework many unsupervised feature extraction methods have been applied to face recognition. In this context the seminal proposal is the *eigenfaces* approach, by Turk and Pentland [2], that uses Principal Component Analysis (PCA) to find the optimal subspace under the reconstruction error criterion. On the other hand, supervised feature extraction methods have been also applied for dimensionality reduction. In that case, Fisher Linear Discriminant Analysis (LDA) [3] is the most known technique, and different extensions of the algorithm have been developed to relax some of the original assumptions. Some examples are the Nonparametric Discriminant Analysis [4] or the Boosted Feature Extraction [5], [6]. However, the main drawback of these methods is that we need a large number of training samples to obtain competitive accuracies under the NN approach.

More recently, new machine learning techniques have been developed and applied to high dimensional data classification, improving considerably the accuracies of face recognition. Among these methodologies, Support Vector Machines (SVM) [7] and Boosting techniques [8] are the most successful. The efficiency of the Boosting family of classifiers has been shown theoretically an empirically [9], being related to the margin theory [10] and their generalization capabilities. In this context, the Adaboost algorithm was declared *the best of-the-shelf* [11] classification ensemble method. However, some of these algorithms are still difficult to scale when new classes are added to the system. For this reason, although there have been substantial improvements in the high dimensional data classification problem, the online learning topic is still an open issue in the current state-of-the-art face recognition classifiers.

In this paper we introduce a face recognition scheme to deal with some of the above mentioned difficulties: the robustness against the small-size training set problem, and the scalability to add new classes avoiding a new costly additional training step (online learning). For this purpose we consider the Multi Task Learning (MTL) paradigm for the face recognition problem. The term MTL was firstly introduced by Caruana in [12]. He showed that simultaneously learning related tasks in an environment can achieve important improvements at two different levels: (*i*) the number N of training samples needed to learn each classification task decreases as more related tasks are learned (*parallel knowledge transfer*), and (*ii*) a classifier trained on sufficiently related tasks is likely to find good solutions to solve novel related tasks, under some theoretical assumptions (*sequential knowledge transfer*). More recently,

Baxter [13] has proved that the number of training samples required to train each task decreases linearly as the number of tasks increases $O(\frac{1}{N}log(O(K)))$. On the other hand, different classifiers have been extended to the MTL paradigm, being one of the most successful approaches the JointBoost algorithm of Torralba et al. [14], that we describe in section III.

In this work we use the JointBoost algorithm to build a robust classifier for face recognition using a fixed number of classes K. Then we extend the algorithm in order to incorporate new unseen classes avoiding the expensive computational cost of retraining the whole system. A brief review of online learning techniques applied to face recognition is included in the next section, and after that we describe the proposed algorithm. Moreover, we perform subject recognition experiments with two standard face databases, focusing on evaluating the scalability against the addition of new subjects to the system. The experiments are detailed in section IV and, finally, section V concludes this work.

II. Online Learning

The online learning paradigm studies the capacity to evolve and update previous knowledge, given a set of new data inputs. In the machine learning community, the terms online learning, incremental learning and life long learning are usually used as synonymous, referring always to the need of rapidly adapting in time to past mastered tasks. Notice that the field covers the addition of new classification tasks, the extension of the previous classification tasks to include new classes, or simply the addition of updated samples to improve the model.

In the face classification field, psychological studies have shown that humans born with a pre-wired capacity to recognize general face patterns [18]. This capacity evolves as the babies grow and are able to adapt their behavior to the environment, recognizing specific "classes" such as familiar faces, gender, age groups or race [19]. In this psychological studies a two step modelling is remarked: (i) An initial set up where the global model is established, and (ii) a continuous adaptation process of the learned model from the environment and the changing object representations. These two steps are also typical in the online machine learning methodologies.

Despite of being a relatively recent discipline, the interest in the online learning techniques has grown considerably during the last years. In a first taxonomy, two families of online learning algorithms can be distinguished:

- Methods that perform online feature extraction to model the incremental addition of new data samples. The classification task is performed using a standard batch classifier, usually the Nearest Neighbor rule.
- Online learning classifiers designed to adjust their parameters to model new samples.

In this section we review some state-of-the-art online learning techniques belonging to both groups. In the feature extraction case, we focus on the online PCA and LDA, which have been used in the comparison of the experimental validation. Also, we summarize the online ensemble learning strategies suggested in previous works, stressing their limitations in incremental multiclass problems.

A. Online Feature Extraction

Classic online learning techniques applied to classification are focused on building a model of the known visual data, by performing a feature extraction process. Then a standard pattern recognition classifier is applied on the reduced space, for instance Mahalanobis mean-distance [20] or the Nearest Neighbor approach. After that, the parameters of the model are updated when new learning instances are given to the system.

One of the most successful approaches to online learning in visual problems is the extension of the Principal Component Analysis Algorithm (PCA) [21] to update the coefficients on the projected space [22][20] (Incremental PCA, IPCA). Given a set of images $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, a model $\mathbf{\Psi} = \{\mu, \mathbf{U}\}$ is constructed, where $\mathbf{U} = [\mathbf{u}_k]$ for $k = 1, \ldots, R$ is the projection matrix computed as the first R eigenvectors of the covariance matrix $\mathbf{C}$

$$\mathbf{C} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T \tag{1}$$

and μ is the sample mean $\mu = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$.

Once the model $\mathbf{\Psi}$ is learned, the goal is to update the parameters when a new image $\mathbf{x}_{N+1}$ is added to the initial set X. The first step is to update the mean, that is

$$\mu' = \frac{1}{N+1}(N\mu + \mathbf{x}_{N+1}) \tag{2}$$

And the projection matrix $\mathbf{U}$ is updated by:

$$\mathbf{U}' = [\mathbf{U} \quad \mathbf{h}_{N+1}]\mathbf{R} \tag{3}$$

where $\mathbf{h}_{N+1} = (\mathbf{U}\mathbf{p}_{N+1} + \mu) - \mathbf{x}_{N+1}$ is the normalized residual vector, being $\mathbf{p}_{N+1}$ the projection of the sample $\mathbf{x}_{N+1}$ using $\mathbf{U}$; $\mathbf{R}$ is a rotation matrix obtained from solving the eigenproblem $\mathbf{D}R = R\Lambda$, where

$$D = \frac{N}{N+1}\begin{pmatrix}\Lambda & 0\\ 0 & 0\end{pmatrix} + \frac{N}{(N+1)^2}\begin{pmatrix}\mathbf{p_{N+1}}\mathbf{p_{N+1}}^T & \gamma\mathbf{p_{N+1}}\\ \gamma\mathbf{p_{N+1}}^T & \gamma^2\end{pmatrix} \tag{4}$$

and $\gamma = \mathbf{h}_{N+1}^T(\mathbf{x}_{N+1} - \mu)$. Reader can find full details about the experimental robustness of the algorithm in [22].

The main drawback of this approach is that the PCA algorithm only optimizes the reconstruction error under the mean squared criterion. Nevertheless, if the labels of the data are taken into account, a supervised feature extraction can be performed to optimize the class separability. For instance, Fisher Linear Discriminant Analysis (LDA) uses the scatter between elements of the same and the scatter between elements of different classes to find a discriminative eigenspace. The online LDA algorithm (ILDA) [23] computes the initial model as $\mathbf{\Psi} = \{\mu, \mathbf{S_w}, \mathbf{S_b}, N\}$, such that

$$\mathbf{S_w} = \sum_{c=1}^{K}\Sigma_c = \sum_{c=1}^{K}\sum_{x\in C_c}(\mathbf{x} - \mu_c)(\mathbf{x} - \mu_c)^T \tag{5}$$

$$\mathbf{S_b} = \sum_{c=1}^{K}N_c(\mu_c - \mu)(\mu_c - \mu)^T \tag{6}$$

where N_c is the number of samples of class Cc, $\mu = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$ is the sample mean, and μ_c is the mean of the

samples of class Cc. The objective in FLD is to maximize the ratio between $\mathbf{S_b}$ and $\mathbf{S_w}$, obtaining the projection matrix $\mathbf{U}$ as the first eigenvectors with larger eigenvalue of $\mathbf{C} = {\mathbf{S_w}}^{-1}\mathbf{S_b}$.

According to [23] the update steps on the online learning algorithm using the model $\mathbf{\Psi}$ take a new sample $\mathbf{x}_{N+1}$ with label C_k, and adjust the model parameters to find $\mathbf{\Psi}' = \{\mu', \mathbf{S'_w}, \mathbf{S'_b}, N'\}$ using only $\mathbf{\Psi}$ and the new sample $\mathbf{x}_{N+1}$.

To update the between and within scatter matrices, two different situations must be distinguished: (*i*) the new sample $\mathbf{x}_{N+1}$ belongs to an known class C_k, $1 \leq k \leq K$ (*ii*) the new sample belongs to a new class C_{K+1}. In the first case the scatter matrixes are updated as

$$\mathbf{S'_b} = \sum_{c=1}^{K} N'_c(\mu_c - \mu')(\mu_c - \mu')^T \tag{7}$$

where, $\mu'_k = (1/(N_k + 1))(N_k\mu_k + \mathbf{x}_{N+1})$, $N'_k = N_k + 1$, $\mu'_c = \mu_c$ and $N'_c = N_c$ for all $c \neq k$, and μ' is the updated mean

$$\mu' = \frac{N\mu + \mathbf{x}_{N+1}}{N+1} \tag{8}$$

The within-class scatter matrix is updated as

$$\mathbf{S'_w} = \sum_{c=1, c\neq k}^{K} \mathbf{\Sigma_c} + \mathbf{\Sigma'_k} \tag{9}$$

where

$$\mathbf{\Sigma'_k} = \mathbf{\Sigma_k} + \frac{N_k}{N_k+1}(\mathbf{x}_{N+1} - \mu_c)(\mathbf{x}_{N+1} - \mu_c)^T \tag{10}$$

On the other hand, when $\mathbf{x}_{N+1}$ belongs to a new class $\in C_{K+1}$, the between-class scatter matrix is updated as

$$\mathbf{S'_b} = \sum_{c=1}^{K+1} N'_c(\mu_c - \mu')(\mu_c - \mu')^T \tag{11}$$

where the $N'_c = N_c$ for all $1 \leq c \leq K$, $N'_k = 1$ and $\mu'_{K+1} = \mathbf{x}_{N+1}$. Notice that in that case the within-class scatter does not change $\mathbf{S}'_w = \mathbf{S}_w$.

Sequential and chunk versions of the algorithm presented above have been proposed in the literature in order to efficiently compute the projected coefficients [24], [25]. In addition, incremental eigendecomposition has also been used in other visual applications such as active shape models [26] and clustering [27]. Notice that in all of these cases the whole online learning method has two steps: first a feature extraction is performed to learn a model, and then a classifier is trained in the new feature space. In the next section, we propose a new online learning method that brings the updating parameters step to the classifier itself, avoiding the feature extraction process. On the other hand our proposal can be complemented by IPCA or ILDA algorithms as a previous step to classification.

B. Online Ensemble Learning

In this paper we propose a new online boosting strategy to incrementally add new classes to a previously learned face recognition model. In the recent literature, different implementations of the Adaboost algorithm have been extended to the online framework. Most of them are based on the seminal works of Oza [28], [29], [30]. In his approach, he simulates the Adaboost sampling with replacement by using a Poisson distribution. He achieved similar (asymptotically equivalent) accuracies with respect to the batch Adaboost version, while running considerably faster. Grabner and Bischof [31] adapted this online boosting procedure to feature selection and applied it to the problems of background subtraction, tracking and object detection. Similar approaches have been followed by Javed et al. [32] in their online cotraining algorithm, and Pham and Cham [33] in their online asymmetric boosting proposal.

All these previous works have focused on two class problems, and the online learning methodology is used to enhance classifiers accuracy by adding new labelled samples to the previously learned model. Moreover, these methods only allow the addition of new samples to the model, being the number of classes fixed a priori. Up to our knowledge, no multiclass extension of the online boosting algorithm has been yet formally proposed.

In order to deal with face recognition problems, with typically 50-100 classes (subjects to recognize), we define a multiclass boosting algorithm that makes use of the multitask learning paradigm. The proposed methodology allows the dynamic addition of new classes, resulting a competitive alternative to the traditional online feature extraction algorithms.

III. The Online Boosting Algorithm

In the recent machine learning literature the interest for classifier ensembles has grown considerably [8]. Experimental results show that the combination of multiple classifiers can lead to a more powerful decision rule than single isolated learning. Depending on the classifier generation and combination rule, three different families of ensembles can be described: *Random Subspace Methods* (RSM), where weak learners are trained using random subsampling of the feature set; *Bagging* where the training set is split in several groups and a weak classifier is independently trained on each sub set; and *Boosting* where the weak classifiers are serially learned on reweighted versions of the training set. Although in the general case none of these ensemble methods outperforms the rest [34], the boosting methodology has been favored in the machine learning field. Notable methods are the Adaboost algorithm (from **Ada**ptive **Boost**ing) and the Gentleboost variant [35], which is specially robust for face classification tasks [36], [37]. In this context, some theoretical bounds have been proven [10], such as the fast convergence to a 0 training error and the strong resistance to overfitting.

There are two general approaches to extend the binary Adaboost classifier to the multiclass case: to adapt the optimized loss function to the multiclass case [38] or to combine different binary classifiers using error correction output codes [39]. In this last framework, Torralba et al. [14] recently introduced JointBoost, a new algorithm based on the knowledge transfer concept to extend the Gentleboost method to the multiclass case. The main idea of this approach is to see the multiclass classification problems as a set of binary classification tasks which are related by sharing features. Torralba et al. [14]

experimentally show that the obtained multiclass classifier can be trained using less examples and also less different weak learners.

In this section we detail the proposed online extension of the JointBoost approach, building a new global scheme where new classes can be added to the system once the model has been learned with an initial set of classes. The whole training algorithm can be divided in two steps: first the JointBoost algorithm is run, obtaining a model $\mathbf{\Psi}$. The second step takes as input the Q samples of the new class $\{\mathbf{x_{N+1}}, \ldots, \mathbf{x_{N+Q}}\}$ and the corresponding labels, $c_{N+1} = \ldots = c_{N+Q} = C_{K+1}$, and runs M rounds of the proposed Online Boosting algorithm.

A. Model Setting: JointBoost Algorithm

The algorithm takes as input the N training samples $\mathbf{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_N}\}$, the corresponding labels $\{c_1, ..., c_N\}$, $c_i \in \{C_1, \ldots, C_K\}$, and a predefined number M of boosting rounds are performed. At each boosting step, the multiclass classification problem is transformed to a binary problem by grouping the classes in two clusters, a positive one and a negative one, and a decision stumps classifier is trained on this binary problem.

Regarding to the grouping in positive and negative clusters, all the possible groupings should be considered at each round. Nevertheless, when the number of classes is large this approach is not possible, given that the number of possibilities is $O(2^K)$. In this case, Torralba et al.[14] followed a best first search approximation ($O(K^2)$), where the grouping is performed as follows:

1) Train K different weak learners by considering each class as the only candidate member of the positive cluster (the remaining classes are included in the negative cluster). Each weak learner is built by considering a feature and the optimal decision stump classifier which can be defined for the binary problem on that feature.
2) Select from the K problems the one which shows minimum weighted classification error and add the associated class to the initial Positive cluster.
3) For the remaining classes, and until no improvement on the classification error can be found, we iterate the following steps:
   - Train a set of different classifiers by considering the previous Positive cluster but adding one class candidate from the Negative cluster.
   - Add the class candidate to the Positive cluster only if the joint selection improves the previous classification error.

This process heuristically selects a class grouping with low classification error and defines a binary problem for each boosting step.

Once the binary problem has been defined, the set of weights $\mathbf{W_i^c}$ are adjusted according to the partial classification results. Note that the optimal grouping is different at each step, given that the error criterion is computed taking into account the weights that focus the cluster selection on the most difficult samples. This grouping step allows the transfer of knowledge among several recognition tasks. Moreover, the feature set is shared across classes on each weak classifier, allowing a more general representation which can be useful when new classes are added to the system (online learning from samples belonging to new unknown classes).

The parameters of the weak learner at the t-th iteration are computed as

$$\rho_t = \frac{\sum_{c \in \mathbf{Positive}_t} \sum_i \mathbf{W}_i^c b_i^c \delta(\mathbf{x}_i^j \leq \theta)}{\sum_{c \in \mathbf{Positive}_t} \sum_i \mathbf{W}_i^c \delta(\mathbf{x}_i^j \leq \theta)}, \tag{19}$$

$$\alpha_t + \rho_t = \frac{\sum_{c \in \mathbf{Positive}_t} \sum_i \mathbf{W}_i^c b_i^c \delta(\mathbf{x}_i^j > \theta)}{\sum_{c \in \mathbf{Positive}_t} \sum_i \mathbf{W}_i^c \delta(\mathbf{x}_i^j > \theta)}, \tag{20}$$

$$s_t^c = \frac{\sum_i \mathbf{W}_i^c b_i^c}{\sum_i \mathbf{W}_i^c}, \text{ if c}\notin \mathbf{Positive}_t \tag{21}$$

where s_t^c acts as a constant to prevent the effects of unbalanced training sets on the class selection; $\{W_i^c\}$ is the weights set, having one a weight for each i-th sample and c-th class; and $\mathbf{Positive_t}$ indicates whether each class was in the positive cluster at t-th iteration or not. Figure 1 summarizes the JointBoost algorithm.

B. Online Boosting: Adding new classes to the Model

As a result of the JointBoost algorithm, we obtain a model of the classifier defined by the parameters $\mathbf{\Psi} = \{\mathbf{h}_t, \mathbf{Positive_t}; \mathbf{t} = \mathbf{1}, ..., \mathbf{M}\}$. Thus, once the JointBoost algorithm is trained, it can be used only for the learned K-class problem, when a new class $K+1$ is added to the system, the whole model must be retrained. Notice that in a K-class problem, at each iteration of the JointBoost algorithm we have to try all possible ways of grouping the classes ($2^K - 1$). Moreover, for each grouping we have to find the best feature, trying each time all the D possibilities. In fact, for $K = 100$ and $D = 500$ it is unfeasible to train the JointBoost in a reasonable amount of time using a Pentium IV computer.

The idea of the proposed Online Boosting algorithm is to take benefit of the class grouping performed in the sharing features step in order to incorporate online new classes to the system. Thus we can avoid the mentioned computationally expensive part of the learning step. More concretely, when we want to add a new class $K+1$ to the model, the Online Boosting is iterated M times. At each iteration t we have to perform 2 steps:

1) Update $\mathbf{Positive}_t$. That is, for each iteration there is a class grouping of classes $1, ..., K$ in a positive and a negative cluster obtained by the JointBoost. This first step consists on assigning the new class $K+1$ to the most suitable cluster under an error minimization criterion, using the feature and the parameters previously learned. Moreover, the weak learner is adjusted using the new training samples.
2) According to the cluster assignation done in the previous step, the weights of the examples are updated.

In first step, and given the binary problem defined over the K initial classes, we consider the alternative assignation of the new to class to the positive or to the negative cluster. Then we compute the weighted error for both class groupings and select

Given the inputs

- training set $X = \{x_1, ..., x_N\}$ containing the data samples
- vector C with the corresponding labels $c_i \in \{C_1, \ldots, C_K\}$

1) Initialize a set of weights: $\mathbf{W}_i^c(1) = 1$ and $H(x_i, c) = 0$ for all $i = 1, ..., N$ and $c = 1, ..., K$
2) For $t = 1 \ldots M$
   a) For all the possible ways of grouping the classes in binary problems $n = 1, \ldots, 2^K - 1$
      i) Learn the shared regression stumps classifier on the projected data, obtaining the hypothesis:

$$h_t^n(\mathbf{x}_i, c) = \begin{array}{ll} \alpha_t \delta(\mathbf{x}_i^j > \theta) + \rho_t, & \text{when } c_i \in \textbf{Positive(n)} \\ s_t^c, & \text{when } c_i \notin \textbf{Positive(n)} \end{array} \quad (12)$$

      Denoting by $\mathbf{x}_i^j$ the j-th feature of the projected sample $\mathbf{x}_i$, where the decision stumps classifier obtains the maximum accuracy on the training data.
      ii) Compute the weighted error for the class grouping as:

$$Err_p(n) = \sum_{c=1}^{K} \sum_{i=1}^{N} \mathbf{W}_i^c (b_i^c - h_t^n(\mathbf{x}_i, c))^2. \quad (13)$$

   where $b_i^c \in \{-1, +1\}$ is the binary class label assigned to the class C_i in the n-th binary grouping.
   b) Find the binary grouping of classes m with minimum Err_p:

$$m = \arg\min_n Err_p(n) \quad (14)$$

   and set $h_t := h_t^m$
   c) Update the data weights:

$$\mathbf{W}_i^c(t+1) = \mathbf{W}_i^c(t) exp^{-b_i^c h_t(\mathbf{x}_i, c)}, \quad i = 1, \ldots, N. \quad (15)$$

   d) Update the estimation for each class:

$$\mathbf{H}(\mathbf{x}_i, c) = \mathbf{H}(\mathbf{x}_i, c) + h_t(\mathbf{x}_i, c) \quad (16)$$

3) Output: Classifier $H(x_i, c) = \sum_t h_t(x, c)$ and corresponding class clusterings $\mathbf{Positive}_t$, $t = 1, ..., M$

Fig. 1. The JointBoost algorithm used to set the initial model.

the assignation with minimum error rate. Thus the class has been definitively assigned to one of the clusters of the binary problem. In the second step the new weak learner is adjusted and we update the weights and the classification function H.

The computational complexity of the algorithm to add a new class is $O(M)$, while to retrain the whole system for $K + 1$ classes is $O(M \times (K + 1)^2)$. On the other hand, the method allows the inclusion of many new classes, given that the same process can be iteratively repeated adding a new class each time. Furthermore, the cost of adding a new class $K + 1$ do not depend on the number of classes K.

More details and the pseudo code of the Online Boosting algorithm can be found in figure 2. Particularly, the iterative process is detailed in point 2. Notice that the first step, consisting on the cluster updating, is composed by stages $a - e$, while the second stage of the iterative process, where the weights are adjusted and the classifier us updated, is done in stages $f - g$. Moreover, an example applied to face recognition is shown in figure 3. First, we show the class grouping evolution of an initial 10 class problem along 50 boosting steps, using the JointBoost. At each round, elements in the positive cluster are denoted with a white square and elements in the negative cluster are denoted with black squares. Furthermore, we show down the cluster assignation of a new class, obtained with the Online Boosting at each iteration.

IV. EXPERIMENTS

The experiments have been performed using two different face databases: the Face Recognition Grand Challenge (FRGC) [40], and the AR Face database [41]. The original FRGC data set consists of more than 3700 high resolution still images from 275 different subjects, and there are between 4 and 32 images per subject. In the experiments performed, only the 160 subjects with more than 20 images have been used, to estimate properly the scatter matrices in the classic online learning methods (ILDA). The AR Face data set consist of 26 images from 126 subjects with uniform background and different acquisition conditions, there are: 2 neutral images, 6 images with gesture effects, 6 images with strong changes in the illumination (left,right and both illumination types), 6 images with occlusion due to sunglasses (combined with the 3 illumination effects), and 6 images with occlusions due to the use of scarf (combined with the 3 illumination effects) from each person. Images where acquired in two different sessions, separated by two weeks. We only used samples from the 86 people that attended both sessions.

Each data set has been previously normalized before the learning phase. Images have been converted from the original RGB space to gray scale. Then, the faces have been rotated and scaled according to the inter-eye distance, in such a way that the center pixel of each eye coincides in all of them. The samples were then cropped obtaining a 37×33 thumbnail,

Given the inputs

- training set $X' = \{X, x_{N+1}, ..., x_{N+Q}\}$ containing the data samples from the new class C_{K+1}
- vector C with the corresponding labels $c_i \in \{C_1, \ldots, C_K, C_{K+1}\}$
- the model $\mathbf{\Psi} = \{\mathbf{h}_t, \mathbf{Positive_t}; t = 1, \ldots, M\}$ previously obtained with the training set $X = \{x_1, ..., x_N\}$

1) Initialize a set of weights: $\mathbf{W}_i^c(1) = 1$, $i = 1, \ldots, N+Q$, and $H'(x_i, c) = 0$ for all $i = 1, ..., N$ and $c = 1, ..., K+1$
2) For $t = 1 \ldots M$
   a) Assign the new samples to the Positive cluster, according to the optimal class grouping selected on the step t in the previous model $\mathbf{\Psi}$, obtaining $\mathbf{Positive}_t(p)$
   b) Classify the training data $\mathbf{X}'$ using the decision stumps generated at the step t of the previous JointBoost algorithm but adjusting the parameter s_t^c using the new samples.

$$h_t^p(\mathbf{x}_i, c) = \begin{array}{ll} \alpha_t \delta(\mathbf{x}_i^j > \theta) + \rho_t, & \text{when } c_i \in \mathbf{Positive}_t(p) \\ \widetilde{s}_t^c, & \text{when } c_i \notin \mathbf{Positive}_t(p) \end{array} \tag{17}$$

   c) Compute the weighted error for the class grouping as:

$$Err_p = \sum_{c=1}^{K+1} \sum_{i=1}^{N+Q} \mathbf{W}_i^c (b_i^c - h_t^p(\mathbf{x}_i, c))^2. \tag{18}$$

   where $b_i^c \in \{-1, +1\}$ is the label assigned to C_i in the current clustering.
   d) Assign the new samples to the Negative cluster, according to the optimal class grouping selected on the step t in the previous model $\mathbf{\Psi}$, obtaining $\mathbf{Positive}_t(n)$, and compute the error Err_n as in 17 and 18.
   e) Assign the new class to the clustering with minimum error $m = \arg\min_{p,r}(Err_p, Err_n)$ and set $h'_t := h_t^m$, $\mathbf{Positive}'_t = \mathbf{Positive}_t(m)$
   f) Update the data weights: $\mathbf{W}_i^c(t+1) = \mathbf{W}_i^c(t) exp^{-b_i^c h'_t(\mathbf{x}_i, c)}, \quad i = 1, \ldots, N.$
   g) Update the estimation for each class: $\mathbf{H}'(\mathbf{x}_i, c) = \mathbf{H}'(\mathbf{x}_i, c) + h'_t(\mathbf{x}_i, c)$
3) Output: Classifier $H'(x_i, c) = \sum_t h'_t(x, c)$ and corresponding class clusterings $\mathbf{Positive}'_t$, $t = 1, ..., M$

Fig. 2. Updating algorithm to adjust the parameters by adding new samples to the model.

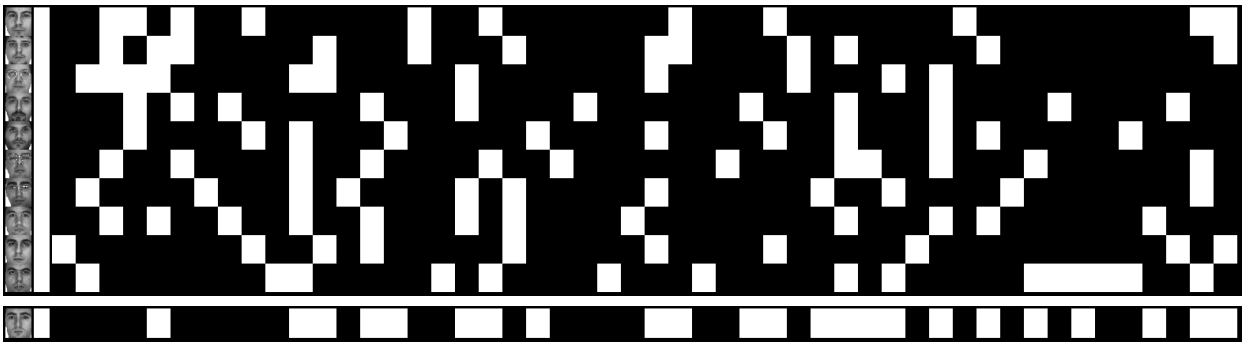


Fig. 3. Matrix representation of the first 50 boosting steps, for a 10 class problem (a representative face of the training set is shown for each class). We plot a white square for the samples belonging to the positive cluster, and a black square in the negative case. Features are shared across classes along the boosting steps. A new class is incrementally added by finding its optimal grouping.

preserving only the internal region of the faces. Thus, the final sample from each image becomes a 1221 feature vector. In Figure 4 some examples from both databases are shown.

We repeated all the experiments 10 times, according to the following protocol: (*i*) We randomly take 25 classes (subjects) from each database to set up the online learning algorithms, and (*ii*) we progressively add one class at each step up to the maximum number of classes, updating the online model parameters in each case. For each learning algorithm, we evaluate the mean accuracy across the 10 iterations, and we show the 95% confidence intervals near the mean value. The final results are shown in table I. The 50% of the samples are used for training and 50% for testing.

The experiments have been performed using 4 methods:

- *Batch* method: In this case, no online learning has been applied. We have chosen the non parametric discriminant analysis technique (NDA), which has been shown to improve the performance of other classic discriminant analysis techniques [42] under the NN rule. The model is trained using the first 25 classes, and the same projection matrix is used when new classes are added. The nearest neighbor rule is applied on the extracted features.
- *Incremental PCA*: The incremental PCA algorithm has been used, setting the original projection matrix with the first 25 classes, an applying the update rules seen in section 2.
- *Incremental LDA*: The incremental LDA algorithm, following the same protocol as the IPCA.

Fig. 4. Examples of faces from the FRGC (top) and AR Face (down) databases.

- *Online Boosting*: We train the online boosting algorithm described in section 3 with the initial 25 classes an update with the remaining as in IPCA and ILDA.

In the first 3 cases, the Nearest Neighbor rule (using Euclidean distance) has been used for classification on the extracted subspace. The reduced subspace retains 200 features, which has been shown to be optimal cross validating the training set.

A. Experimental Results

The results with the FRGC database show an accuracy close to 98% using our boosted approach for the initial problem with 25 classes, while the application of feature extraction methods with the NN classifier obtains an initial 92%. This experiment suggests that for a perfectly acquired and normalized set, the use of JointBoost is the best option for multiclass face classification problems. Figure 5 shows the accuracies as a function of the number of classes. The accuracy of the first 25 steps is intentionally plotted as a constant line to show the initial training of the original subset of classes, where any online learning is performed. We can see that the accuracy decreases, as expected, when new classes are added to the system. This fact is due to 2 reasons: first, larger class problems are more difficult to classify, and second, when new samples are added to the system, there is an implicit error given that the whole classifier has not been retrained (only an estimation of the new parameters is performed). Nevertheless, the accuracy does not decrease drastically, even when we increase the number of classes an 800%. The second best performing method in both data sets is the Online LDA, which abruptly decreases the accuracy when the models are updated the first time. The reason is that in high dimensional data (typical from visual problems) the proper estimation of the class distributions (scatters) becomes unprecise. The performance of the IPCA algorithm is clearly inferior, given that class memberships are not taken into account in the feature extraction process.

In addition, we show the absolute and relative loss of accuracy when new classes are added (see Table I). For each data set we add up to the maximum number of classes (160 and 86 for the FRGC and AR Face respectively) and take the resulting accuracy. The absolute decrease is computed subtracting the accuracy obtained when we use the maximum number of classes from the initial accuracy (25 subjects). The relative decrease is computed as the absolute decrease divided by the initial accuracy (considering the 25 classes). Note that with the proposed boosted approach the accuracy decreases less, specially in the case of the AR Face data set, obtaining a more robust classification rule in presence of occlusions and strong changes in the illumination. On the other hand, in both data sets the less decrease score is obtained by the IPCA technique, as expected, given the low performance of the technique and its unsupervised nature. As explained in section II, PCA finds the "face subspace" from the data, independently from the class membership. Therefore, given enough samples, the projection matrix obtained does not vary considerably.

The main advantage using our online learning approach is the reduction of the computational needs. It has been shown experimentally that the use of JointBoost achieves high accuracies in face classification. Nevertheless, the computational cost makes the method unfeasible when the problem has too many classes, due to the BFS clustering step (O(K^2)). More concretely, training the JointBoost algorithm using an initial set of 25 classes takes 8 hours on a Pentium IV computer (using the Matlab software), while learning the same algorithm using 80 classes can take weeks. However, to extend the previous 25 class problem to the new 80 class problem using

TABLE I
MEAN ACCURACY OF THE ANALYZED METHODS AND OUR PROPOSAL ON TWO FACE DATABASES. ONLY 25 CLASSES ARE USED FOR TRAINING, A TOTAL OF 135 EXTRA CLASSES HAVE BEEN ADDED IN THE FRGC CASE, AND 65 IN THE AR FACE. WE SHOW THE DECREASE (ABSOLUTE AND RELATIVE PERCENTAGE) IN THE MEAN ACCURACY FROM THE FIRST EXPERIMENT WITH ONLY 25 CLASSES AND THE LARGEST EXTENDED PROBLEM.

FRGC	Accuracy±Int.	Decrease	Relative
NDA	0.855±0.029	0.055	6.0%
IPCA	0.833±0.181	0.028	3.3%
ILDA	0.859±0.013	0.115	11.8%
Online Boosting	0.921±0.010	0.057	5.8%

ARFACE	Accuracy±Int.	Decrease	Relative
NDA	0.601±0.006	0.211	25.9%
IPCA	0.605±0.022	0.071	10.5%
ILDA	0.679±0.015	0.207	23.4%
Online Boosting	0.752±0.011	0.106	12.4%

our approach takes just a few minutes. Moreover, the 160 final class problem is nowadays non computable in a reasonable amount of time.

B. Initial class set selection

One of the free parameters of the proposed algorithm is the number of initial classes to train the model. In order to analyze the influence of this initial class set on the global accuracy we performed the same experiment as above, but using initial sets of 5, 10, 15, 20 and 25 classes. Figure 6 shows the mean accuracies as a function of the number of classes in the FRGC and ARFace data sets.

As expected, the algorithm fares better with large initial class sets, given that the sharing process involved in the JointBoost algorithm can use richer information for the face recognition task. In the small initial class sets, the drop in the accuracy is noticeable. Nevertheless, as the amount of initial classes increases, the improvement on the accuracy is less important, being not statistically significant in the 20-25 cases. The experiment suggests that an initial 25 class set could be enough to initialize the JointBoost model.

V. CONCLUSIONS

We propose an online extension of the JointBoost algorithm in order to solve real world face recognition problems. Our proposal deals with multiclass classification problems. Several extensions of the successful Adaboost algorithm have been suggested in the related works. Nevertheless, they are usually limited to two class problems, and the number of classes to classify can not be online extended. The main contribution of this work with respect to the state-of-the-art is the possibility of incrementally adding new classes to a previously trained problem, which is specially useful in face recognition applications.

The multiclass problem is seen as a set of multiple binary classification tasks (one versus all) that are trained sharing the feature space. The method uses an initial set to build a model using the JointBoost algorithm and then the system is readjusted when new classes are added to the initial set. In that way the proposed Online Boosting for face recognition is able to consider a final amount of subjects that will be computationally unfeasible with the JointBoost.

We have experimentally validated our proposal using two different face databases: the FRGC database acquired in a controlled environment, and the AR Face database which contains important artifacts due to strong changes in the illumination and partial occlusions. When the original sets are extended to large class problems, the results show that the classification accuracy decreases less drastically than using the classic NN rule used in state-of-the-art online learning methods.

The initial number of classes necessary to build the JointBoost model depends exclusively on the final application, being limited by the availability of samples in training time and the computational resources. In the experiments performed, we show that an initial 25 class set yields us a good trade off between training time and final accuracies.

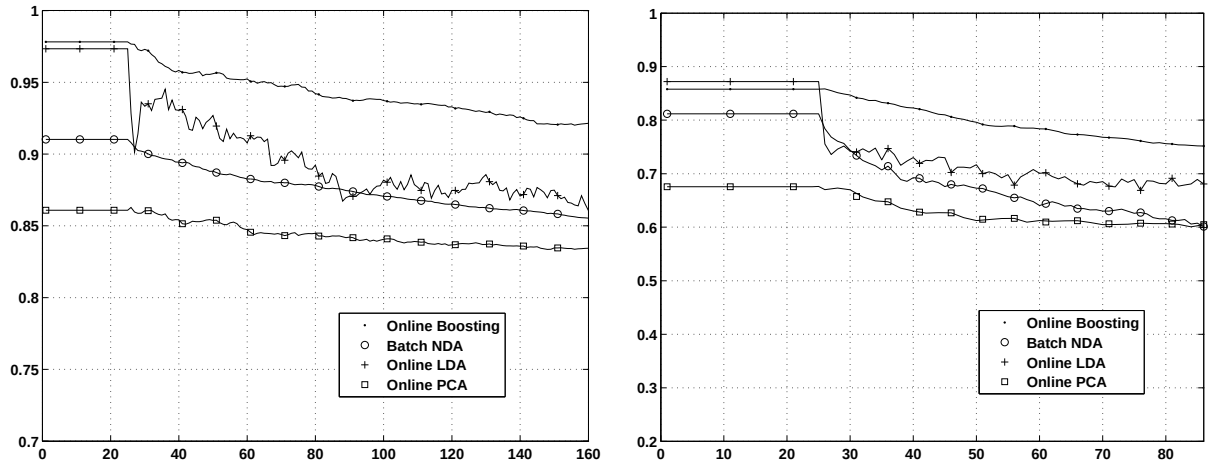
We plan as future work to analyze the importance of the classes chosen in the original trained algorithm. A diverse choice of the initial classes should allow a more general base for extending the classifier. Moreover, the use of an extra validation set could improve slightly the accuracies.

ACKNOWLEDGMENT

This work was partially supported by MEC grants TIC2006-15308-C02-01 and CONSOLIDER-INGENIO 2010 (CSD2007-00018).

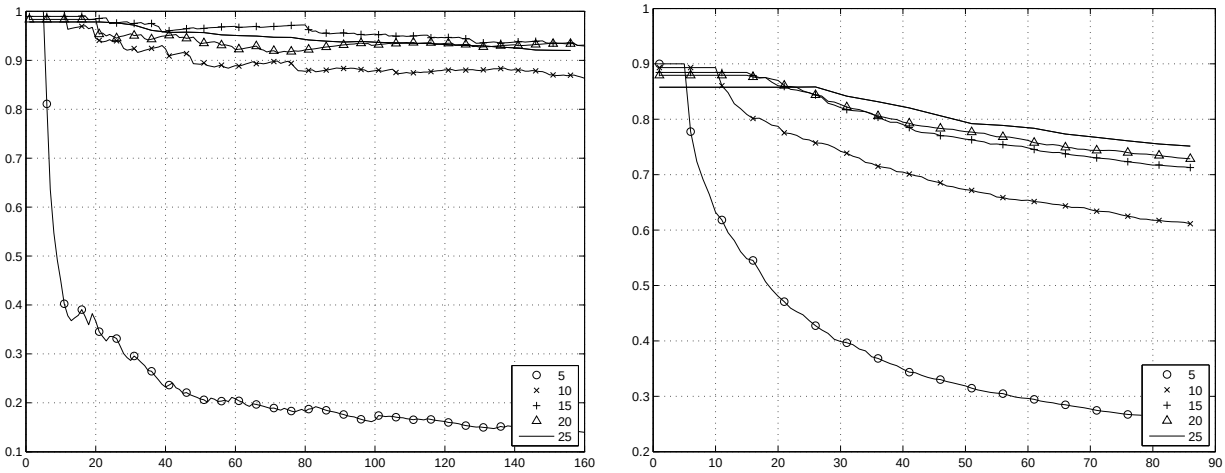
REFERENCES

[1] R. Bellman, *Adaptive Control Process: A Guided Tour*. New Jersey: Princeton University Press, 1961.
[2] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, Mar 1991.
[3] R. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, pp. 179–188, 1936.
[4] K. Fukunaga and J. Mantock, "Nonparametric discriminant analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, no. 6, pp. 671–678, nov 1983.
[5] D. Masip, L. I. Kuncheva, and J. Vitria, "An ensemble-based method for linear feature extraction for two-class problems," *Pattern Analysis and Applications*, vol. 8, pp. 227–237, 2005.
[6] D. Masip and J. Vitria, "Boosted discriminant projections for nearest neighbor classification," *Pattern Recognition*, vol. 39, no. 2, pp. 164–170, 2006.
[7] V. N. Vapnik, *The nature of statistical learning theory*. Springer-Verlag New York, Inc., 1995.
[8] Ludmila I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. New Jersey: Wiley, July 2004.
[9] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *International Conference on Machine Learning*, 1996, pp. 148–156.
[10] R. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated prediction," *Machine Learning*, vol. 3, p. 297336, 1999.
[11] L. Breiman, "Arcing classifiers," *The Annals of Statistics*, vol. 26, no. 3, pp. 801–849, 1998.
[12] R. Caruana, "Multitask learning." *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997.
[13] J. Baxter, "A model of inductive bias learning," *Journal of Machine Learning Research*, vol. 12, pp. 149–198, 2000.
[14] A. Torralba, K. Murphy, and W. Freeman, "Sharing features: efficient boosting procedures for multiclass object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2004.



(a) Accuracy using the FRGC data set. (b) Accuracy using the ARFace data set.

Fig. 5. Accuracy as a function of the number of classes



(a) Accuracy using the FRGC data set. (b) Accuracy using the ARFace data set.

Fig. 6. Mean accuracy of different initial training sets (with 5, 10, 15, 20 and 25 classes) as a function of the initial number of classes

[15] C. R. Linder, "Self-organization in a simple task of motor control based on spatial encoding," *Adaptive Behavior*, vol. 13, no. 3, pp. 189–209, 2005.

[16] J. Walker, S. Garrett, and M. Wilson, "Evolving controllers for real robots: A survey of the literature," *Adaptive Behavior*, vol. 11, no. 3, pp. 179–203, Sep. 2003.

[17] R. W. Paine and J. Tani, "How hierarchical control self-organizes in artificial adaptive systems," *Adaptive Behavior*, vol. 13, no. 3, pp. 211–225, 2005.

[18] M. Fischler and R. Elschlager, "The representation and matching of pictorial scenes," *IEEE Trans. on Computers*, vol. 22(1), p. 67, Jan. 1973.

[19] D. Gelder and B. Rouw, "Beyond localisation: A dynamical dual route account of face recognition," *Acta Psychologica*, vol. 107, pp. 183–207, 2001.

[20] P. M. Hall, A. D. Marshall, and R. R. Martin, "Incremental eigenanalysis for classification." in *BMVC*, 1998.

[21] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp. 103–108, Jan 1990.

[22] M. Artac, M. Jogan, and A. Leonardis, "Incremental pca or on-line visual learning and recognition." in *ICPR (3)*, 2002, pp. 781–784.

[23] S. Pang, S. Ozawa, and N. Kasabov, "Incremental linear discriminant analysis for classification of data streams," *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, vol. 35, no. 5, pp. 905–914, October 2005.

[24] B. S. Manjunath, S. Chandrasekaran, and Y. F. Wang, "An eigenspace update algorithm for image analysis," in *Symposium on Computer Vision*, 1995, pp. 551–556.

[25] P. M. Hall, A. D. Marshall, and R. R. Martin, "Merging and splitting eigenspace models," *IEEE Trans. Pattern Anal. Mach. Intell*, vol. 22, no. 9, 2000.

[26] M. Fussenegger, P. M. Roth, H. Bischof, and A. Pinz, "On-line, incremental learning of a robust active shape model," in *German Pattern Recognition Symposium*, 2006, pp. 122–131.

[27] S. Ozawa, S. Pang, and N. Kasabov, "Incremental learning of feature space and classifier for on-line pattern recognition," *Int. J. Know.-Based Intell. Eng. Syst.*, vol. 10, no. 1, pp. 57–65, 2006.

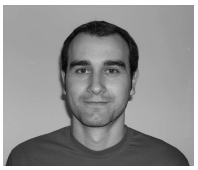
[28] N. C. Oza and S. Russell, "Online bagging and boosting," in *Eighth International Workshop on Artificial Intelligence and Statistics*, T. Jaakkola and T. Richardson, Eds. Key West, Florida. USA: Morgan Kaufmann, January 2001, pp. 105–112.

[29] N. C. Oza, "Online ensemble learning," Ph.D. dissertation, The University of California, Berkeley, CA, Sep 2001.

[30] Nikunj C. Oza, "Online bagging and boosting," in *International Conference on Systems, Man, and Cybernetics, Special Session on Ensemble Methods for Extreme Environments*. New Jersey: Institute for Electrical and Electronics Engineers, October 2005, pp. 2340–2345.

[31] H. Grabner and H. Bischof, "On-line boosting and vision," in *CVPR (1)*, 2006, pp. 260–267.

[32] O. Javed, S. Ali, and M. Shah, "Online detection and classification of moving objects using progressively improving detectors," in *CVPR (1)*, 2005, pp. 696–701.
[33] M.-T. Pham and T.-J. Cham, "Online learning asymmetric boosted classifiers for object detection," in *CVPR*, 2007.
[34] M. Skurichina and R. P.W.Duin, "Bagging, boosting and the random subspace method for linear classifiers," *Pattern Analysis and Applications*, vol. 5, pp. 121–135, 2002.
[35] J. Friedman, T.Hastie, and R.Tibshirani, "Additive logistic regression: a statistical view of boosting," *Annals of statistics*, vol. 28, pp. 337–374, 2000.
[36] V. P. Rainer Lienhart, Alexander Kuranov, "Empirical analysis of detection of boosted classifiers for rapid object detection," Microsoft Research Lab, Intel Labs, Tech. Rep., 2002.
[37] J. J. Yokono and T. Poggio, "A multiview face identification model with no geometric constraints," Sony Intelligence Dynamics Laboratories, Tech. Rep., March 2006.
[38] H. Z. Ji Zhu, Saharon Rosset and T. Hastie, "Multi-class adaboost," Standford University, Tech. Rep., January 2006.
[39] R. E. Schapire, "Using output codes to boost multiclass learning problems," in *Proc. 14th International Conference on Machine Learning*. Morgan Kaufmann, 1997, pp. 313–321.
[40] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "The 2005 ieee workshop on face recognition grand challenge experiments," in *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*. Washington, DC, USA: IEEE Computer Society, 2005, p. .45.
[41] A. Martinez and R. Benavente, "The AR Face database," Computer Vision Center, Tech. Rep. 24, june 1998.
[42] M. Bressan and J. Vitria, "Nonparametric discriminant analysis and nearest neighbor classification," *Pattern Recognition Letters*, vol. 24, no. 15, pp. 2743–2749, nov 2003.



David Masip received the Ph.D. degree in computer science at the Computer Vision Center (CVC), in the Autonomous University of Barcelona (UAB), Spain (2005). In 2007 he became Associate Professor in the Computer Science Department from the Universitat Oberta de Catalunya (UOC). His research interests concern the development of algorithms of feature extraction, pattern recognition, face classification and machine learning. He has authored more than 30 scientific papers and one book.

Agata Lapedriza works as assistant professor at the Universitat Oberta de Catalunya (UOC). She received her MS degree in Mathematics at the Universitat de Barcelona (UB) in 2003. The same year she joined to the Computer Vision Center and the Universitat Autònoma de Barcelona (UAB). In 2005 she obtained her BA degree in Computer Science at UAB and currently she is a Ph.D. candidate advised by Dr.Vitrià. Her research interests are related to statistical pattern recognition, face classification and machine learning.

Jordi Vitrià received the Ph.D. degree from the Autonomous University of Barcelona (UAB) for his work on mathematical morphology in 1990. He joined the Applied Mathematics and Analysis Department, University of Barcelona (UB), as an Associate Professor in 2007. His research interests include machine learning, pattern recognition, and visual object recognition. He is the author of more than 50 scientific publications and several books.