Maximum Margin Projection Subspace Learning for Visual Data Analysis

Symeon Nikitidis, Anastasios Tefas, *Member, IEEE*, and Ioannis Pitas, *Fellow, IEEE*

***Abstract*— Visual pattern recognition from images often involves dimensionality reduction as a key step to discover a lower dimensional image data representation and obtain a more manageable problem. Contrary to what is commonly practiced today in various recognition applications where dimensionality reduction and classification are independently treated, we propose a novel dimensionality reduction method appropriately combined with a classification algorithm. The proposed method called maximum margin projection pursuit, aims to identify a low dimensional projection subspace, where samples form classes that are better discriminated, i.e., are separated with maximum margin. The proposed method is an iterative alternate optimization algorithm that computes the maximum margin projections exploiting the separating hyperplanes obtained from training a support vector machine classifier in the identified low dimensional space. Experimental results on both artificial data, as well as, on popular databases for facial expression, face and object recognition verified the superiority of the proposed method against various state-of-the-art dimensionality reduction algorithms.**

***Index Terms*— Maximum margin projections, support vector machines, face recognition, facial expression recognition, object recognition.**

I. Introduction

ONE of the most crucial problems that every image analysis algorithm encounters is the high dimensionality of the image data, which can range from several hundreds to thousands of extracted image features. Directly dealing with such high dimensional data is not only computationally inefficient, but also yields several problems in subsequently performed statistical learning algorithms, due to the so-called *"curse of dimensionality"*. Thus, various techniques have been proposed in the literature for efficient data embedding (or dimensionality reduction) that obtain a more manageable problem and alleviate computational complexity. Such a popular category of methods is the subspace image representation algorithms which aim to discover a low dimensional representation of the image data by projecting linearly or non-linearly the high-dimensional input samples to a low-dimensional subspace, where an appropriately formed criterion is optimized.

The most popular dimensionality reduction algorithms can be roughly categorized, according to their underlying optimization criteria, into two main categories. Those that form their optimization criterion based on geometrical arguments and those that attempt to enhance data discrimination in the projection subspace. The goal of the first category methods is to embed data into a low-dimensional space, where the intrinsic data geometry is preserved. Principal Component Analysis (PCA) [1] is such a representative method that exploits the global data structure, in order to identify a subspace where the sample variance is maximized. While PCA exploits the global data characteristics in the Euclidean space, the local data manifold structure is ignored. To overcome this deficiency, manifold-based embedding algorithms assume that the data reside on a submanifold of the ambient space and attempt to discover and preserve its structure. Such representative methods include e.g. ISOMAP [2], Locally Linear Embedding (LLE) [3], Locality Preserving Projections [4], Orthogonal Locality Preserving Projections (OLPP) [5] and Neighborhood Preserving Embedding (NPE) [6].

Discrimination enhancing embedding algorithms aim to identify a discriminative subspace, in which the data samples from different classes are far apart from each other. Linear Discriminant Analysis (LDA) [7] and its variants, are such representative methods that extract discriminant information by finding projection directions that maximize the ratio of the between-class and the within-class scatter. Margin maximizing embedding algorithms [8]–[10] inspired by the great success of Support Vector Machines (SVMs) [11] also fall in this category, since their goal is to enhance class discrimination in the low dimensional space.

The Maximum Margin Projection (MMP) algorithm [9] is an unsupervised embedding method that attempts to find orthogonal projection directions that separate data in different clusters with maximum margin. To do so, MMP iteratively seeks for such a data partitioning, so that if a binary SVM classifier is trained, the resulting separating hyperplane separates the two data clusters with maximum margin. Thus, the projection direction of the corresponding SVM trained on such a data labelling, is considered as one of the directions of the sought subspace, while considering different possible data clusterings and enforcing the constraint that each subsequently found

Manuscript received January 24, 2014; revised May 31, 2014; accepted July 19, 2014. Date of publication August 18, 2014; date of current version September 5, 2014. This work was supported by the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement 248434 (MOBISERV). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Andrea Cavallaro.

S. Nikitidis was with the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece. He is now with the Department of Computing, Imperial College London, London, SW7 2AZ, U.K. (e-mail: s.nikitidis@imperial.ac.uk).

A. Tefas and I. Pitas are with the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece (e-mail: tefas@aiia.csd.auth.gr; pitas@aiia.csd.auth.gr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2014.2348868

SVM hyperplane is orthogonal to the previous ones, several projections are derived and added to the subspace. He et. al [8] proposed a semisupervised dimensionality reduction method for image retrieval that aims to discover both geometrical and discriminant structures of the data manifold. To do so, the algorithm constructs a within-class and a between-class graph by exploiting both class and neighborhood information and finds a linear transformation matrix that maps image data to a subspace, where, at each local neighborhood, the margin between relevant and irrelevant images is maximized.

Recently significant attention has been attracted by Compressed Sensing (CS) [12] that combines data acquisition with data dimensionality reduction performed by Random Projections (RP). RP are a desirable alternative of traditional embedding techniques, since they offer certain advantages. Firstly, they are data independent and do not require a training phase thus being computationally efficient. Secondly, as it has been shown in the literature [13]–[15], a Gaussian random projection matrix preserves the pairwise distances between data points in the projection subspace and, thus, can be effectively combined with distance-based classifiers, such as SVMs. Another important aspect for real life applications using sensitive biometric data is the provision of security and user privacy protection mechanisms, since the use of random features, instead of the actual biometric data for e.g. person identification, protects the original data [16] from malicious attacks.

In this paper we integrate optimal data embedding and SVM classification in a single framework to be called Maximum Margin Projection Pursuit (MMPP). MMPP algorithm first initializes the projection matrix as a semiorthogonal Gaussian RP matrix in order to exploit the aforementioned merits. Subsequently, it iteratively and till convergence trains an SVM classifier in order to identify the optimal decision hyperplanes in the low dimensional subspace and updates the projection matrix such that the separating margin between the projected samples of different classes is maximized. The MMPP approach brings certain advantages, both to data embedding and classification. In contrary to what is commonly practiced where dimensionality reduction and classification are treated independently, MMPP combines these into a single framework. Furthermore, in contrast to the conventional classification approaches, which consider that the training data points are fixed in the input space, the SVM classifier is trained over the projected data samples in the projection subspace determined by MMPP. Thus, working on low dimensional data reduces the required computational effort. Moreover, since the decision hyperplane identified by SVM training is explicitly determined by the support vectors, data outliers and the overall data samples distribution inside classes do not affect MMPP performance, in contrast to other discriminant subspace learning algorithms, such as LDA which assumes a Gaussian data distribution for optimal class discrimination. Furthermore, although the proposed method and the MMP algorithm share similar characteristics, since both capitalize on the maximum margin principle and exploit the SVM training problem, our method is radically different overcoming certain deficiencies of [9]. More precisely, MMPP is a supervised learning algorithm unlike MMP which is unsupervised build upon the assumption that samples of the same class are grouped together in the initial high dimensional input space thus being close to each other in the Euclidean sense forming compact data clusters. However, this assumption is rarely true, since usually data do not form compact data clusters but have a multimodal distribution [17], [18]. This fact significantly affects the correctness of the projection directions identified by MMP. In addition, the proposed algorithm exploits in the learning process the actual separating hyperplanes, contrary to MMP algorithm which at each step constraints the identified by SVM normal vector so that it is orthogonal to all previously found projection directions thus the resulting subspace bases are no longer directly determined by the support vectors.

In summary, the novel contributions of this paper are the following:

- The MMPP algorithm integrates data embedding and classification into a single framework, thus possessing certain desired advantages (good classification performance, computational speed and robustness to data outliers).
- MMPP is derived both for two class and multiclass data embedding problems.
- The MMPP non-linear extension that seeks to identify a projection matrix that separates different classes in the feature space with maximum margin is also demonstrated.
- The superiority of the proposed method against various state-of-the-art embedding algorithms for facial image characterization problems and object recognition is verified by several simulation experiments on popular datasets.

The rest of the paper is organized as follows. Section II presents the proposed MMPP dimensionality reduction algorithm for a two-class linear classification problem and discusses its initialization using a semiorthogonal Gaussian random projection matrix, in order to form the basis of the projection subspace. MMPP extension to a multiclass problem is presented in Section III, while its non-linear extension considering either a Gaussian Radial Basis or an arbitrary degree polynomial kernel function is derived in Section IV. Section V describes the conducted experiments and presents experimental evidence regarding the superiority of the proposed algorithm against various state-of-the-art data embedding methods. Finally, concluding remarks are drawn in Section VI.

II. Maximum Margin Projection Pursuit

The MMPP algorithm aims to identify a low-dimensional projection subspace, where samples form classes that are better discriminated, i.e., are separated with maximum margin. To do so, MMPP involves three main steps. The first step, performed during the initialization of the MMPP algorithm, extracts the random features from the initial data and forms the basis of the low-dimensional projection subspace using RP, while the second and the third steps involve two optimization problems that are combined in a single iterative alternate optimization framework. More precisely, the second step identifies the optimal decision hyperplane that separates different classes with maximum margin, in the respective subspace determined

by the projection matrix, while the third step updates the projection matrix, so that the identified separating margin between the projected samples of different classes is increased. Next, we first formulate the optimization problems considered by MMPP, discuss algorithm initialization and demonstrate the iterative optimization framework considering both a two class and a multiclass separation problem. Subsequently, we derive the non-linear MMPP algorithms extension and propose update rules considering polynomial and Gaussian kernel functions to project data into a Hilbert space, using the so-called kernel trick.

A. MMPP Algorithm for Binary Classification Problems

Given a set $\mathcal{X} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ of N training data pairs, where $\mathbf{x}_i \in \mathbb{R}^m, i = 1, \ldots, N$ are the m-dimensional input feature vectors and $y_i \in \{-1, 1\}$ is the class label associated with each sample $\mathbf{x}_i$, a binary SVM classifier attempts to find the separating hyperplane that separates training data points of the two classes with maximum margin, while minimizing the classification error defined according to which side of the decision hyperplane training samples of each class fall in. Considering that each training sample of $\mathcal{X}$ is firstly projected from the initial m-dimensional input space to a low-dimensional subspace using a projection matrix $\mathbf{R} \in \mathbb{R}^{r \times m}$, where $r \ll m$ and performing the linear transformation $\acute{\mathbf{x}}_i = \mathbf{R}\mathbf{x}_i$, the binary SVM optimization problem is formulated as follows:

$$\min_{\mathbf{w}, \xi_i, \mathbf{R}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i \tag{1}$$

subject to the constraints:

$$y_i\left(\mathbf{w}^T\mathbf{R}\mathbf{x}_i + b\right) \geq 1 - \xi_i \tag{2}$$

$$\xi_i \geq 0, \quad i = 1, \ldots, N, \tag{3}$$

where $\mathbf{w} \in \mathbb{R}^r$ is the normal vector of the separating hyperplane, which is r-dimensional, since training is performed in the projection subspace, $b \in \mathbb{R}$ is its bias term, $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_N]^T$ are the slack variables, each one associated with a training sample and C is the term that penalizes the training error.

The MMPP algorithm attempts to learn a projection matrix $\mathbf{R}$, such that the low-dimensional data sample projection is performed efficiently, thus enhancing the discrimination between the two classes. To quantify the discrimination power of the projection matrix $\mathbf{R}$, we formulate our MMPP algorithm based on geometrical arguments. To do so, we employ a combined iterative optimization framework, involving the simultaneous optimization of the separating hyperplane normal vector $\mathbf{w}$ and the projection matrix $\mathbf{R}$, performed by successively updating the one variable, while keeping the other fixed. Next we first discuss the derivation of the optimal separating hyperplane normal vector $\mathbf{w}_o$, in the projection subspace determined by $\mathbf{R}$ and subsequently, we demonstrate the projection matrix update with respect to the fixed $\mathbf{w}_o$.

1) Finding the Optimal $\mathbf{w}_o$ in the Projection Subspace Determined by $\mathbf{R}$: The optimization with respect to $\mathbf{w}$, is essentially the conventional binary SVM training problem performed in the projection subspace determined by $\mathbf{R}$, rather than in the input space. To solve the constrained optimization problem in (1) with respect to $\mathbf{w}$, we introduce positive Lagrange multipliers α_i and β_i each associated with one of the constraints in (2) and (3), respectively and formulate the Lagrangian function $\mathcal{L}(\mathbf{w}, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha}, \boldsymbol{\beta})$:

$$\begin{aligned}\mathcal{L}(\mathbf{w}, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i \\ - \sum_{i=1}^{N}\alpha_i\left[y_i\left(\mathbf{w}^T\mathbf{R}\mathbf{x}_i + b\right) - 1 + \xi_i\right] \\ - \sum_{i=1}^{N}\beta_i\xi_i.\end{aligned} \tag{4}$$

The solution can be found from the saddle point of the Lagrangian function, which has to be maximized with respect to the dual variables $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ and minimized with respect to the primal ones $\mathbf{w}, \boldsymbol{\xi}$ and b. According to the Karush-Kuhn-Tucker (KKT) conditions [19] the partial derivatives of $\mathcal{L}(\mathbf{w}, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ with respect to the primal variables $\mathbf{w}, \boldsymbol{\xi}$ and b vanish deriving the following equalities:

$$\frac{\partial\mathcal{L}(\mathbf{w}, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial\mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{N}\alpha_i y_i\mathbf{R}\mathbf{x}_i, \tag{5}$$

$$\frac{\partial\mathcal{L}(\mathbf{w}, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N}\alpha_i y_i = 0, \tag{6}$$

$$\frac{\partial\mathcal{L}(\mathbf{w}, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial\xi_i} = 0 \Rightarrow \beta_i = C - \alpha_i. \tag{7}$$

By substituting terms from the above equalities into (4), we switch to the dual formulation, where the optimization problem with respect to the primal variables in (1) is reformulated to the maximization of the following Wolfe dual problem:

$$\min_{\mathbf{R}}\max_{\boldsymbol{\alpha}} \quad \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j}^{N}\alpha_i\alpha_j y_i y_j\mathbf{x}_i^T\mathbf{R}^T\mathbf{R}\mathbf{x}_j \tag{8}$$

subject to the constraints:

$$\sum_{i=1}^{N}\alpha_i y_i = 0, \quad \alpha_i \geq 0, \quad \forall \;\; i = 1, \ldots, N. \tag{9}$$

Consequently, solving (8) for $\boldsymbol{\alpha}$ the optimal separating hyperplane normal vector $\mathbf{w}_o$ in the reduced dimensional space determined by $\mathbf{R}$, is subsequently derived from (5).

2) Maximum Margin Projection Matrix Update for Fixed $\mathbf{w}_o$: At each optimization round t we seek to update the projection matrix $\mathbf{R}^{(t-1)}$, so that its new estimate $\mathbf{R}^{(t)}$ improves the objective function in (8) by maximizing the margin between the two classes. To do so, we first project the high dimensional training samples $\mathbf{x}_i$ from the input space to a low dimensional subspace, using the projection matrix $\mathbf{R}^{(t-1)}$ derived during the previous step, and subsequently,

train the binary SVM classifier in order to obtain the optimal Lagrange multipliers $\boldsymbol{\alpha}_o$ specifying the normal vector of the separating hyperplane $\mathbf{w}_o^{(t)}$.

To formulate the optimization problem for the projection matrix $\mathbf{R}$, we exploit the dual form of the binary SVM cost function defined in (8). However, since term $\sum_{i=1}^{N} \alpha_i$ is constant with respect to $\mathbf{R}$, we can remove it from the cost function. Moreover, in order to retain the geometrical correlation between samples in the projection subspace, we constrain the derived updated projection matrix $\mathbf{R}^{(t)}$ to be semiorthogonal. Consequently, the constrained optimization problem for the projection matrix $\mathbf{R}$ update can be summarized by the objective function $\mathcal{O}(\mathbf{R})$ as follows:

$$\max_{\mathbf{R}} \mathcal{O}(\mathbf{R}) = \frac{1}{2} \sum_{i,j}^{N} \alpha_{i,o} \alpha_{j,o} y_i y_j \mathbf{x}_i^T \mathbf{R}^T \mathbf{R} \mathbf{x}_j, \tag{10}$$

subject to the constraints:

$$\mathbf{R}\mathbf{R}^T = \mathbf{I}, \tag{11}$$

where $\mathbf{I}$ is an $r \times r$ identity matrix. The orthogonality constraint introduces an optimization problem on the Stiefel manifold solved to find the dimensionality reduction matrix $\mathbf{R}$.

In the literature, optimization of problems with orthogonality constraints is performed using a gradient descent algorithm along the Stiefel manifold geodesics [20], [21]. However, the simplest approach to take the constraint $\mathbf{R}\mathbf{R}^T = \mathbf{I}$ into account, is to update $\mathbf{R}$ using any appropriate unconstrained optimization algorithm, and then to project $\mathbf{R}$ back to the constraint set [22]. This is the direction we have followed in this paper, where we first solve (10), without the orthonormality constraints on the rows of the projection matrix and obtain $\acute{\mathbf{R}}$. Consequently, the projection matrix update is accomplished orthonormalizing the rows of $\acute{\mathbf{R}}$ by performing a Gram-Schmidt procedure. Thus, we solve (10) for $\mathbf{R}$ keeping $\mathbf{w}_o^{(t)}$ fixed, by applying a steepest ascent optimization algorithm, which, at a given iteration t, invokes the following update rule:

$$\acute{\mathbf{R}}^{(t)} = \mathbf{R}^{(t-1)} + \lambda_t \nabla \mathcal{O}(\mathbf{R}^{(t-1)}), \tag{12}$$

where λ_t is the learning step parameter for the t-th iteration and $\nabla \mathcal{O}(\mathbf{R}^{(t-1)})$ is the partial derivative of the objective function $\mathcal{O}(\mathbf{R})$ in (10) with respect to $\mathbf{R}^{(t-1)}$, evaluated as:

$$\begin{aligned} \nabla \mathcal{O}(\mathbf{R}^{(t-1)}) &= \sum_{i,j}^{N} \alpha_{i,o} \alpha_{j,o} y_i y_j \mathbf{R}^{(t-1)} \mathbf{x}_i \mathbf{x}_j^T \\ &= \sum_{i=1}^{N} \alpha_{i,o} y_i \mathbf{w}_o^{(t)} \mathbf{x}_i^T. \end{aligned} \tag{13}$$

Thus, $\acute{\mathbf{R}}^{(t)}$ is derived as:

$$\acute{\mathbf{R}}^{(t)} = \mathbf{R}^{(t-1)} + \lambda_t \left(\sum_{i=1}^{N} \alpha_{i,o} y_i \mathbf{w}_o^{(t)} \mathbf{x}_i^T \right). \tag{14}$$

Obtaining the projection matrix $\acute{\mathbf{R}}^{(t)}$ that increases the separating margin between the two classes in the projection subspace, we subsequently orthonormalize its rows to derive $\mathbf{R}^{(t)}$.

An efficient approach for setting an appropriate value to the learning step parameter λ_t based on the Armijo rule [23] is presented in [24], which is also adopted in this work. According to this strategy, the learning step takes the form $\lambda_t = \beta^{g_t}$, where g_t is the first non-negative integer value found satisfying:

$$\mathcal{O}(\mathbf{R}^{(t)}) - \mathcal{O}(\mathbf{R}^{(t-1)}) \geq \sigma \langle \nabla \mathcal{O}(\mathbf{R}^{(t-1)}), \mathbf{R}^{(t)} - \mathbf{R}^{(t-1)} \rangle, \tag{15}$$

where operator $\langle ., . \rangle$ is the Frobenius inner product. Parameters β and σ in our experiments have been set to $\beta = 0.1$ and $\sigma = 0.01$, which is an efficient parameter selection, as has been verified in other studies [24], [25].

After deriving the new projection matrix $\mathbf{R}^{(t)}$, the previously identified separating hyperplane is no longer optimal, since it has been evaluated in the projection subspace determined by $\mathbf{R}^{(t-1)}$. Consequently, it is required to re-project the training samples using $\mathbf{R}^{(t)}$ and retrain the SVM classifier to obtain the current optimal separating hyperplane and its normal vector. Thus, MMPP algorithm iteratively updates the projection matrix and evaluates the normal vector of the optimal separating hyperplane $\mathbf{w}_o$ in the projection subspace determined by $\mathbf{R}$, until the algorithm converges.

In order to verify whether the learned projection matrix $\mathbf{R}^{(t)}$ at each iteration round t is optimal or not, we track the partial derivative value in (13) to identify stationarity. The following stationarity check step is performed, which examines whether the following termination condition is satisfied:

$$||\nabla \mathcal{O}(\mathbf{R}^{(t)})||_F \leq e_{\mathbf{R}} ||\nabla \mathcal{O}(\mathbf{R}^{(0)})||_F, \tag{16}$$

where $e_{\mathbf{R}}$ is a predefined stopping tolerance satisfying: $0 < e_{\mathbf{R}} < 1$. In our conducted experiments, we considered that $e_{\mathbf{R}} = 10^{-3}$. The combined iterative optimization process of the MMPP algorithm for the binary classification problem is summarized in Algorithm 1.

B. MMPP Algorithm Initialization

In order to initialize the iterative optimization framework, it is first required to train the binary SVM classifier and obtain the optimal $\mathbf{w}_o$ in a low dimensional subspace determined by an initial projection matrix $\mathbf{R}^{(0)}$, used in order to perform dimensionality reduction and form the basis of the projection subspace. To do so, we construct $\mathbf{R}^{(0)}$ as a semiorthogonal Gaussian random projection matrix. To derive $\mathbf{R}^{(0)}$ we create the $m \times r$ matrix $\mathbf{R}$ of i.i.d., zero-mean, unit variance Gaussian random variables, normalize its first row and orthogonalize the remaining rows with respect to the first, via a Gram-Schmidt procedure. This procedure results in the Gaussian random projection matrix $\mathbf{R}^{(0)}$ having orthonormal rows that can be used for the initialization of the iterative optimization framework.

III. MMPP Algorithm for Multiclass Classification Problems

The dominant approach for solving multiclass classification problems using SVMs has been based on reducing the multiclass task into multiple binary ones and building a set

Algorithm 1 Outline of the Maximum Margin Projection Pursuit Algorithm Considering a Binary Classification Problem

1: **Input:** The set $\mathcal{X} = \{(\mathbf{x}_i, y_i), \quad i = 1, \ldots, N\}$ of N m-dimensional two class train data samples.
2: **Output:** The optimal maximum margin projection matrix $\mathbf{R}_o$ and the optimal separating hyperplane normal vector $\mathbf{w}_o$.
3: **Initialize:** $t = 1$ and $\mathbf{R}^{(0)} \in \mathbb{R}^{r \times m}$ as a semiorthogonal Gaussian random projection matrix.
4: **repeat**
5: **Project** $\mathbf{x}_i$ to a low dimensional subspace performing the linear transformation: $\acute{\mathbf{x}}_i = \mathbf{R}^{(t-1)}\mathbf{x}_i \quad \forall i = 1, \ldots, N$.
6: **Train** the binary SVM classifier in the projection subspace by solving the optimization problem in (8) subject to the constraints in (9) to obtain the optimal Lagrange multipliers $\boldsymbol{\alpha}_o$.
7: **Obtain** the normal vector of the optimal separating hyperplane as: $\mathbf{w}_o^{(t)} = \sum_{i=1}^{N} \alpha_{i,o} y_i \mathbf{R}^{(t-1)}\mathbf{x}_i$.
8: **Evaluate** gradient $\nabla\mathcal{O}(\mathbf{R}^{(t-1)}) = \sum_{i=1}^{N} \alpha_{i,o} y_i \mathbf{w}_o^{(t)} \mathbf{x}_i^T$.
9: **Determine** learning rate λ_t.
10: **Update** projection matrix $\mathbf{R}^{(t-1)}$ given $\mathbf{w}_o^{(t)}$ as: $\mathbf{R}^{(t)} = \text{Orthogonalize}\big(\mathbf{R}^{(t-1)} + \lambda_t \sum_{i=1}^{N} \alpha_{i,o} y_i \mathbf{w}_o^{(t)} \mathbf{x}_i^T\big)$.
11: $t = t + 1$, $\mathbf{R}_o = \mathbf{R}^{(t)}$ and $\mathbf{w}_o = \mathbf{w}_o^{(t)}$.
12: **until** $||\nabla\mathcal{O}(\mathbf{R}^{(t)})||_F \leq 10^{-3}||\nabla\mathcal{O}(\mathbf{R}^{(0)})||_F$,

of binary classifiers, where each one distinguishes samples between one pair of classes [26]. However, adopting such an one-against-one multiclass SVM classification schema to our MMPP algorithm requires to learn one projection matrix for each of the $k(k-1)/2$ binary SVM classifiers that handle a k-class classification problem. Clearly, this approach becomes impractical for classification tasks involving a large number of classes, as for instance, in face recognition.

A different approach to generalize SVMs to multiclass problems is to handle all available training data together forming a single optimization problem by adding appropriate constraints for every class [27], [28]. However, the size of the generated quadratic optimization problem may become extremely large, since it is proportional to the product of the number of training samples multiplied by the number of classes in the classification task at hand. Crammer and Singer [29] proposed an elegant approach for multiclass classification, by solving a single optimization problem, where the number of added constraints is reduced and remains proportional to the number of the available training samples. More importantly, such a one-against-all multiclass SVM formulation enables us to learn a single maximum margin projection matrix common for all training samples, independently of the class they belong to. Therefore, we adopt this multiclass SVM formulation [29] in this work.

In the multiclass classification context, the training samples $\mathbf{x}_i$ are assigned a class label $y_i \in \{1, \ldots, k\}$, where k is the number of classes. We extend the multiclass SVM formulation proposed in [29], by considering that all training samples are first projected on a low-dimensional subspace determined by the projection matrix $\mathbf{R}$. Our goal is to solve the MMPP optimization problem and to learn a common projection matrix $\mathbf{R}$ for all classes, such that the training samples of different classes are projected in a subspace, where they are separated with maximum margin, and also, to derive k separating hyperplanes, where the p-th hyperplane $p = 1, \ldots, k$ determined by its normal vector $\mathbf{w}_p \in \mathbb{R}^r$, separates the training vectors of the p-th class from all the others with maximum margin.

The multiclass SVM optimization problem in the projection subspace is formulated as follows:

$$\min_{\mathbf{w}_p, \xi_i, \mathbf{R}} \quad \frac{1}{2}\sum_{p=1}^{k} \mathbf{w}_p^T \mathbf{w}_p + C\sum_{i=1}^{N} \xi_i, \tag{17}$$

subject to the constraints:

$$\mathbf{w}_{y_i}^T \mathbf{R}\mathbf{x}_i - \mathbf{w}_p^T \mathbf{R}\mathbf{x}_i \geq b_i^p - \xi_i, \quad i = 1, \ldots, N \quad p = 1, \ldots, k. \tag{18}$$

Here bias vector $\mathbf{b}$ is defined as:

$$b_i^p = 1 - \delta_{y_i}^p = \begin{cases} 1, & \text{if } y_i \neq p \\ 0, & \text{if } y_i = p, \end{cases} \tag{19}$$

where $\delta_{y_i}^p$ is the Kronecker delta function which is 1 for $y_i = p$ and 0, otherwise.

Similar to the binary classification case, we employ a combined iterative optimization framework that successively optimizes either variables $\mathbf{w}_p$, $p = 1, \ldots, k$ keeping matrix $\mathbf{R}$ fixed, (thus, training the multiclass SVM classifier in the projection subspace determined by $\mathbf{R}$) or updates the projection matrix $\mathbf{R}$, so that it improves the objective function i.e., it projects the training samples in a subspace where the margin that separates the training samples of each class from all the others, is maximized. Next, we first demonstrate the optimization process with respect to the normal vectors of the separating hyperplanes in the projection subspace of $\mathbf{R}$ and subsequently, we discuss the projection matrix $\mathbf{R}$ update, while keeping the optimal normal vectors $\mathbf{w}_{p,o}$ fixed.

1) Finding the Optimal $\mathbf{w}_{p,o}$ *in the Projection Subspace Determined by* $\mathbf{R}$*:* Since the derivation of the following dual optimization problem is rather technical, we will briefly demonstrate it and refer the interested reader to [30] and [31] for its complete exposition. To solve the constrained optimization problem in (17) we introduce positive Lagrange multipliers α_i^p, each associated with one of the constraints in (18). Note that it is not required to introduce additional Lagrange multipliers regarding the non-negativity constraint applied on the slack variables ξ_i. This is already included in (18), since, for $y_i = p$, $b_i^p = 0$, inequalities in (18) become $\xi_i \geq 0$. The Lagrangian function $\mathcal{L}(\mathbf{w}_p, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha})$ takes the

form:

$$\mathcal{L}(\mathbf{w}_p, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha}) = \frac{1}{2}\sum_{p=1}^{k}\mathbf{w}_p^T\mathbf{w}_p + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\sum_{p=1}^{k}\alpha_i^p\left[\left(\mathbf{w}_{y_i}^T - \mathbf{w}_p^T\right)\mathbf{R}\mathbf{x}_i + \xi_i - b_i^p\right]. \quad (20)$$

Switching to the dual formulation, the solution of the constrained optimization problem in (17) can be found from the saddle point of the Lagrangian function in (20), which has to be maximized with respect to the dual variables $\boldsymbol{\alpha}$ and minimized with respect to the primal ones $\mathbf{w}_p$ and $\boldsymbol{\xi}$. To find the minimum over the primal variables we require that the partial derivatives of $\mathcal{L}(\mathbf{w}_p, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha})$ with respect to $\boldsymbol{\xi}$ and $\mathbf{w}_p$ vanish, which gives the following equalities:

$$\frac{\partial\mathcal{L}(\mathbf{w}_p, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha})}{\partial\xi_i} = 0 \Rightarrow \sum_{p=1}^{k}\alpha_i^p = C, \quad (21)$$

$$\frac{\partial\mathcal{L}(\mathbf{w}_p, \boldsymbol{\xi}, \mathbf{R}, \boldsymbol{\alpha})}{\partial\mathbf{w}_p} = 0 \Rightarrow \mathbf{w}_p = \sum_{i=1}^{N}\left(\alpha_i^p - C\delta_{y_i}^p\right)\mathbf{R}\mathbf{x}_i \Leftrightarrow \mathbf{w}_p = \sum_{i=1}^{N}\left(\alpha_i^p - \sum_{p=1}^{k}\alpha_i^p\delta_{y_i}^p\right)\mathbf{R}\mathbf{x}_i. \quad (22)$$

By substituting terms from (21) and (22) into (20), and expressing the corresponding to the i-th training sample bias terms and Lagrange multipliers in a vector form as $\mathbf{b}_i = [b_i^1, \ldots, b_i^k]^T$ and $\boldsymbol{\alpha}_i = [\alpha_i^1, \ldots, \alpha_i^k]^T$, respectively and performing the substitution $\mathbf{n}_i = C\mathbf{1}_{y_i} - \boldsymbol{\alpha}_i$, (where $\mathbf{1}_{y_i}$ is a k-dimensional vector with all its components equal to zero except of the y_i-th, which is equal to one) the saddle point of the Lagrangian is reformulated to the minimization of the following Wolfe dual problem:

$$\min_{\mathbf{n}}\max_{\mathbf{R}} \quad \frac{1}{2}\sum_{i,j}^{N}\mathbf{x}_i^T\mathbf{R}^T\mathbf{R}\mathbf{x}_j\mathbf{n}_i^T\mathbf{n}_j + \sum_{i=1}^{N}\mathbf{n}_i^T\mathbf{b}_i, \quad (23)$$

subject to the constraints:

$$\sum_{p=1}^{k}n_i^p = 0, \quad n_i^p \leq \begin{cases} 0, & \text{if } y_i \neq p \\ C, & \text{if } y_i = p \end{cases} \quad \forall i = 1, \ldots, N \quad , \quad p = 1, \ldots, k. \quad (24)$$

By solving (23) for $\mathbf{n}$, and consequently, deriving the optimal value of the actual Lagrange multipliers $\boldsymbol{\alpha}_o$, the normal vector $\mathbf{w}_{p,o}$ is derived from (22), corresponding to the optimal decision hyperplane that separates the training samples of the p-th class from all the others with maximum margin in the projection subspace of $\mathbf{R}$.

2) Maximum Margin Projection Matrix Update for Fixed $\mathbf{w}_{p,o}$*:* Similar to the binary classification case, we formulate the optimization problem by exploiting the dual form of the multiclass SVM cost function in (23). To do so, we remove the term $\sum_{i=1}^{N}\mathbf{n}_i^T\mathbf{b}_i$ from (23), since it is independent of the optimized variable $\mathbf{R}$ and derive the objective function $\mathcal{O}(\mathbf{R})$. In addition we impose orthogonality constraints on the derived projection matrix $\mathbf{R}^{(t)}$ rows, thus leading to the following optimization problem:

$$\max_{\mathbf{R}}\mathcal{O}(\mathbf{R}) = \frac{1}{2}\sum_{i,j}^{N}\mathbf{x}_i^T\mathbf{R}^T\mathbf{R}\mathbf{x}_j\mathbf{n}_i^T\mathbf{n}_j, \quad (25)$$

subject to the constraints:

$$\mathbf{R}\mathbf{R}^T = \mathbf{I}. \quad (26)$$

To derive a new estimate of $\mathbf{R}_o$ at a given iteration t the steepest ascent update rule in (12) is invoked, where $\nabla\mathcal{O}(\mathbf{R}^{(t-1)})$ is the partial derivative of (25) with respect to $\mathbf{R}^{(t-1)}$:

$$\nabla\mathcal{O}(\mathbf{R}^{(t-1)}) = \sum_{i,j}^{N}\mathbf{R}^{(t-1)}\mathbf{x}_i\mathbf{x}_j^T\mathbf{n}_{i,o}^T\mathbf{n}_{j,o} = -\sum_{i=1}^{N}\sum_{p=1}^{k}\alpha_{i,o}^p\left(\mathbf{w}_{y_i,o}^{(t)} - \mathbf{w}_{p,o}^{(t)}\right)\mathbf{x}_i^T. \quad (27)$$

Thus, $\acute{\mathbf{R}}^{(t)}$ is updated as:

$$\acute{\mathbf{R}}^{(t)} = \mathbf{R}^{(t-1)} - \lambda_t\left(\sum_{i=1}^{N}\sum_{p=1}^{k}\alpha_{i,o}^p\left(\mathbf{w}_{y_i,o}^{(t)} - \mathbf{w}_{p,o}^{(t)}\right)\mathbf{x}_i^T\right). \quad (28)$$

The projection matrix update is followed by the orthonormalization of the rows of $\acute{\mathbf{R}}^{(t)}$, in order to to satisfy the imposed constraints. Similar to the binary classification task, MMPP algorithm for multiclass classification problems successively updates the maximum margin projection matrix $\mathbf{R}$ and evaluates the normal vectors $\mathbf{w}_{p,o}$ $p = 1, \ldots, k$ of the k optimal separating hyperplanes in the projection subspace determined by $\mathbf{R}$. The involved learning rate parameter λ_t is set using the previously presented methodology for the binary classification case, while the iterative optimization process is terminated by tracking the partial derivative value in (27) and examining the termination condition in (16).

IV. Non-Linear Maximum Margin Projections

When data can not be linearly separated in the initial input space, a common approach is to perform the so-called kernel trick, using a mapping function $\phi(.)$ that maps (usually non-linearly) the input feature vectors $\mathbf{x}_i$ to a possibly high dimensional space $\mathcal{F}$, called feature space, which usually has the structure of a Hilbert space [32], [33], where the data are supposed to be linearly or near linearly separable. The exact form of the mapping function is not required to be known, since all required subsequent operations of the learning algorithm are expressed in terms of dot products between the input vectors in the Hilbert space performed by the kernel trick [34].

To derive the non-linear extension of the MMPP algorithm, we assume that the low dimensional training sample representations are non-linearly mapped in a Hilbert space using a kernel function and seek to identify such a projection matrix that separates different classes in the feature space with

maximum margin. Next we will only demonstrate the derivation of the update rules for the maximum margin projection matrix $\mathbf{R}$, both for the two class and the multiclass classification problems, considering two popular kernel functions: an arbitrary degree polynomial kernel function and the Radial Basis Function (RBF). However, it is straightforward to extend the non-linear MMPP algorithm, such as to exploit other popular kernel functions using the presented methodology.

A d-degree polynomial kernel function is defined as: $K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + 1)^d$. Considering that training samples are first projected into the low dimensional subspace determined by $\mathbf{R}$, the d-degree polynomial kernel function over the projected samples takes the form:

$$K(\mathbf{R}\mathbf{x}_i, \mathbf{R}\mathbf{x}_j) = \left((\mathbf{R}\mathbf{x}_i)^T \mathbf{R}\mathbf{x}_j + 1\right)^d. \tag{29}$$

Consequently, the partial derivative $\nabla\mathcal{O}(\mathbf{R}^{(t-1)})$ of the objective function for the binary classification case in (10) is evaluated as below:

$$\begin{aligned}\nabla\mathcal{O}(\mathbf{R}^{(t-1)}) &= \frac{\frac{1}{2}\sum_{i,j}^{N} \alpha_{i,o}\alpha_{j,o} y_i y_j K(\mathbf{R}^{(t-1)}\mathbf{x}_i, \mathbf{R}^{(t-1)}\mathbf{x}_j)}{\partial \mathbf{R}^{(t-1)}} \\ &= d\sum_{i,j}^{N} \alpha_{i,o}\alpha_{j,o} y_i y_j \\ &\quad \times \left(\left(\mathbf{R}^{(t-1)}\mathbf{x}_i\right)^T \mathbf{R}^{(t-1)}\mathbf{x}_j + 1\right)^{d-1} \mathbf{R}^{(t-1)}\mathbf{x}_i\mathbf{x}_j^T,\end{aligned} \tag{30}$$

while for the multiclass formulation is evaluated using the cost function in (25) as:

$$\begin{aligned}\nabla\mathcal{O}(\mathbf{R}^{(t-1)}) &= \frac{\frac{1}{2}\sum_{i,j}^{N} K(\mathbf{R}^{(t-1)}\mathbf{x}_i, \mathbf{R}^{(t-1)}\mathbf{x}_j)\mathbf{n}_i^T\mathbf{n}_j}{\partial \mathbf{R}^{(t-1)}} \\ &= d\sum_{i,j}^{N} \left(\left(\mathbf{R}^{(t-1)}\mathbf{x}_i\right)^T \mathbf{R}^{(t-1)}\mathbf{x}_j + 1\right)^{d-1} \\ &\quad \times \mathbf{R}^{(t-1)}\mathbf{x}_i\mathbf{x}_j^T\mathbf{n}_{i,o}^T\mathbf{n}_{j,o}.\end{aligned} \tag{31}$$

On the other hand, the RBF kernel function is defined using the projected samples as: $K(\mathbf{R}\mathbf{x}_i, \mathbf{R}\mathbf{x}_j) = e^{-\gamma||\mathbf{R}\mathbf{x}_i - \mathbf{R}\mathbf{x}_j||^2}$, where γ is the Gaussian spread. Similarly, the partial derivative of (10), with respect to $\mathbf{R}^{(t-1)}$ is evaluated as follows:

$$\begin{aligned}\nabla\mathcal{O}(\mathbf{R}^{(t-1)}) &= 2\gamma\,\mathbf{R}^{(t-1)}\sum_{i,j}^{N} \alpha_{i,o}\alpha_{j,o} y_i y_j \left(\mathbf{x}_i\mathbf{x}_i^T + \mathbf{x}_j\mathbf{x}_j^T\right) \\ &\quad \times K(\mathbf{R}^{(t-1)}\mathbf{x}_i, \mathbf{R}^{(t-1)}\mathbf{x}_j),\end{aligned} \tag{32}$$

while, for the multiclass separation problem, it is evaluated from (25):

$$\begin{aligned}\nabla\mathcal{O}(\mathbf{R}^{(t-1)}) &= 2\gamma\,\mathbf{R}^{(t-1)}\sum_{i,j}^{N} \left(\mathbf{x}_i\mathbf{x}_i^T + \mathbf{x}_j\mathbf{x}_j^T\right) \\ &\quad \times K(\mathbf{R}^{(t-1)}\mathbf{x}_i, \mathbf{R}^{(t-1)}\mathbf{x}_j)\mathbf{n}_{i,o}^T\mathbf{n}_{j,o}.\end{aligned} \tag{33}$$

The update rules for the maximum margin projection matrix are subsequently derived by substituting the respective partial derivatives in (12). Moreover, similar extensions can be derived for other popular non-linear kernel functions, by simply evaluating their partial derivatives with respect to the projection matrix $\mathbf{R}$ and by modifying accordingly the respective update rules.

V. Experimental Results

We compare the performance of the proposed method with that of several state-of-the-art dimensionality reduction techniques, such as PCA, LDA, Subclass Discrimnant Analysis (SDA) [17], LPP, Orthogonal LPP (OLPP), MMP and the linear approximation of the LLE algorithm called Neighborhood Preserving Embedding (NPE). Moreover, in our comparison we include RP [35] and also we directly feed the initial high dimensional samples without performing dimensionality reduction to a multiclass SVM classifier, to serve as our baseline testing methods. Experiments have been performed for facial expression recognition on the Cohn-Kanade database [36], for face recognition on "Labeled Faces in the Wild" (LFW) [37], Extended Yale B [38] and AR [39] datasets and for object recognition on ETH-80 image dataset [40].

On the experiments for facial expression and face recognition as our classification features, we considered the facial image intensity information and its augmented Gabor wavelet representation, which provides robustness to illumination and facial expression variations [41]. To create the augmented Gabor feature vectors we convolved each facial image with Gabor kernels considering 5 different scales and 8 directions. Thus, for each facial image, and for each Gabor kernel a complex vector containing a real and an imaginary part was generated. Based on these parts we computed the Gabor magnitude information creating in total 40 feature vectors for each facial image. Each such feature vector was subsequently downsampled, in order to reduce its dimension using interpolation and normalized to zero mean and unit variance. Finally, for each facial image we derived its augmented Gabor wavelet representation by concatenating the 40 feature vectors into a single vector. On the experiments for object recognition we used the cropped and scaled to a fixed size of 128×128 pixels binary images of ETH-80 containing the contour of each object.

To determine the optimal projection subspace dimensionality for MMPP, MMP, PCA, RP and SDA a validation step was performed exploiting the training set. Moreover, for SDA the optimal number of subclasses that each class is partitioned to has been also specified using the same validation step and exploiting the stability criterion introduced in [42]. For validation we randomly divided the training set into training (80% of the samples) and validation (20% of the samples) sets and the parameters that resulted to the best recognition rate on the validation set were subsequently adopted for the test set.

In order to train the proposed MMPP algorithm and derive the maximum margin projection matrix, we have combined our optimization algorithm with LIBLINEAR [43], which provides an efficient implementation of the considered multiclass linear

TABLE I

BEST AVERAGE EXPRESSION RECOGNITION ACCURACY RATES (%) IN COHN-KANADE DATABASE. IN PARENTHESES IT IS SHOWN THE DIMENSION THAT RESULTS IN THE BEST PERFORMANCE FOR EACH METHOD

	SVM	PCA	LDA (6)	SDA	LPP (6)	OLPP (6)	NPE (6)	MMP	RP	MMPP
Intensity	73.4(30,000)	74.5(260)	74.2	76.4(55)	76.6	75.2	76.4	70.3 (250)	75.2(500)	**80.1(120)**
Gabor	77.8(48,000)	84.6(150)	86.5	86.1(69)	85.5	83.3	84.8	77.1 (150)	79.8(500)	**89.2(80)**

Fig. 1. Sample images depicting facial expressions in the Cohn-Kanade database.

kernel SVM classifier. Moreover, for the fairness of the experimental comparison, the discriminant low-dimensional facial representations derived from each examined algorithm were also fed to the same multiclass SVM implemented in LIBLINEAR for classification. We should note that, by adopting LIBLINEAR we explicitly exploit a linear kernel. However, as it has been shown in the literature [35], linear SVMs are already appropriate for separating different classes and this also makes it possible to directly compare between different algorithms and draw trustworthy conclusions regarding their efficacy. Nevertheless, better performance could be achieved by MMPP algorithm by projecting the input high dimensional samples non-linearly and using non-linear kernel SVMs for their classification.

A. Facial Expression Recognition in the Cohn-Kanade Database

The Cohn-Kanade AU-Coded facial expression database is among the most popular databases for benchmarking methods that perform facial expression recognition. Each subject in each video sequence of the database poses a facial expression, starting from the neutral emotional state and finishing at the expression apex. To form our data collection we discarded the intermediate video frames depicting subjects performing each facial expression in increasing intensity level and considered only the last video frame depicting each formed facial expression at its highest intensity. Face detection was performed on these images and the resulting facial Regions Of Interest (ROIs) were manually aligned with respect to the eyes position. Subsequently, they were anisotropically scaled to a fixed size of 150 $\times$ 200 pixels and converted to grayscale. Thus, in our experiments, we used in total 407 images depicting 100 subjects, posing 7 different expressions (anger, disgust, fear, happiness, sadness, surprise and the neutral emotional state). Fig. 1 shows example images from the Cohn-Kanade dataset, depicting the recognized facial expressions arranged in the following order: anger, disgust, fear, happiness, sadness, surprise and the neutral emotional state.

To measure the facial expression recognition accuracy, we randomly partitioned the available samples into 5 approximately equal sized subsets (folds) and a 5-fold cross-validation has been performed by feeding the projected discriminant facial expression representations to the linear SVM classifier. This resulted into such a test set formation, where some expressive samples of an individual were left for testing, while his rest expressive images (depicting other facial expressions) were included in the training set. This fact significantly increased the difficulty of the expression recognition problem, since person identity related issues arose.

Table I summarizes the best average facial expression recognition rates achieved by each examined embedding method, both for the considered facial image intensity and the augmented Gabor features. The mean facial expression recognition rates attained by directly feeding the initial high dimensional data to the linear SVM classifier are also provided in Table I. Considering the facial image intensity as the input features, MMPP outperforms, in terms of recognition accuracy percentage points, all other competing embedding algorithms. The best average expression recognition rate attained by MMPP is 80.1% using 120D discriminant representations of the initial 30,000D input samples. Exploiting the augmented Gabor features significantly improved the recognition performance of all examined methods, verifying the appropriateness of these descriptors in the task compared against the image intensity features. MMPP algorithm performance increased by more than 9% reaching an average recognition rate of 89.2%. Again MMPP attained the highest average expression recognition rate outperforming the second best method (LDA) by 2.7%. MMP algorithm failed to provide competitive expression recognition performance on both experiments due to the applied experimental protocol. More precisely, since we do not perform person independent expression recognition where all the expressive images of an individual are left for testing, MMP failed to derive discriminative projection directions, since all facial images of the same subject belonging to different expression classes tend to cluster together in the high dimensional input space.

It is significant to highlight the difference in expression recognition performance between PCA, RP and the proposed algorithm when a varying number of features are extracted by each method. Fig. 2 plots the average facial expression recognition rate achieved by each method with respect to the number of extracted features. As it can be observed, MMPP attained the highest recognition rate while was able to extract features with higher discriminant information. Fig. 3 demonstrates the average facial expression recognition rate attained by MMPP, when the initial 48,000D augmented Gabor wavelet representations are projected on a 3D space, versus the number of MMPP learning process iterations. It should be noted that since for the initialization of MMPP

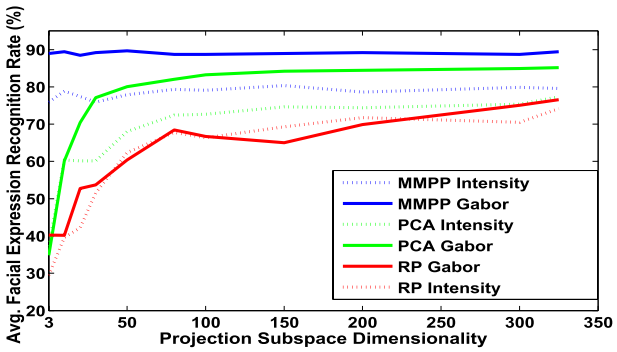


Fig. 2. Average facial expression recognition rate in Cohn-Kanade database with respect to the extracted features.

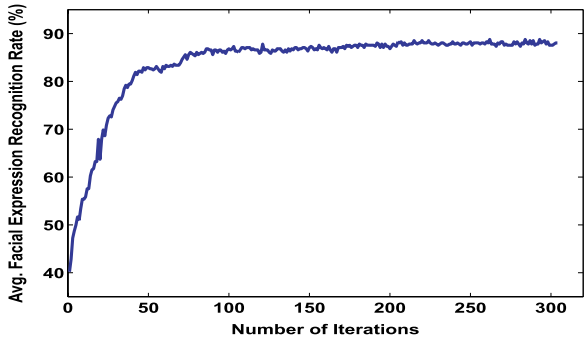


Fig. 3. MMPP average facial expression recognition rate versus the number of optimization iterations. The initial 48,000D Gabor wavelet representations derived from Cohn-Kanade database are projected on a 3D space.

it is exploited a semiorthogonal Gaussian RP matrix, thus practically dimensionality reduction is performed using RP, the reported recognition accuracy at the first iteration, which is 40.2%, is identical to that achieved by RP. As can be observed in Fig. 3, during the iterative optimization process class discrimination in the projection subspace is enhanced and the attained mean expression recognition rate is gradually increased reaching 88.3%.

B. MMPP Algorithms Convergence and Computational Complexity

To investigate MMPP optimization performance, we examined its ability to minimize the cost function in (17) in every optimization round, thus maximizing the separating margin between the projected samples of different classes. We also investigated whether MMPP is able to reach a stationary point by monitoring the gradient magnitude in (27).

In the conducted experiment, we used approximately half (200) of the available expressive images from Cohn-Kanade, in order to train MMPP and learn a 100D discriminant projection subspace. From the expressive facial images we extracted the augmented Gabor features by convolving each with Gabor kernels of 5 different scales and 8 orientations and downsampled, normalized and concatenated the derived filter responses following the same procedure as previously described. The resulting from each facial image $48,000$D feature vectors were used in order to train

TABLE II
TRAINING TIME IN SECONDS REQUIRED BY PCA, LDA, SDA, LPP AND MMPP ON COHN-KANADE DATASET

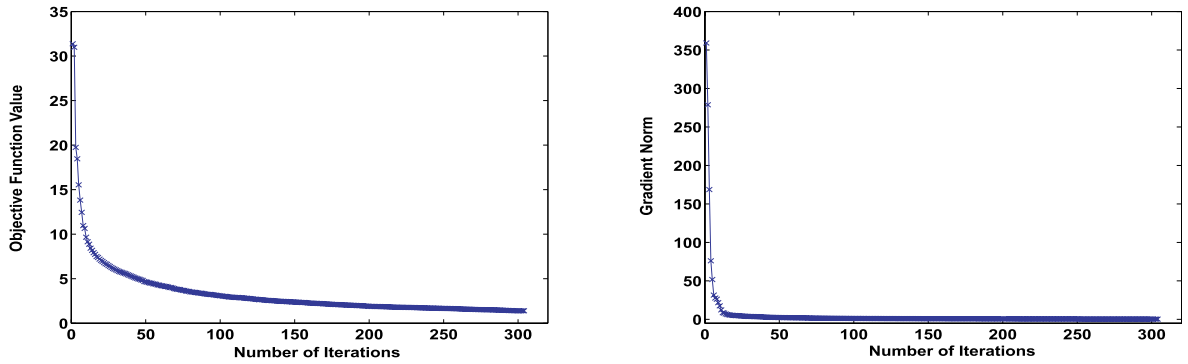
Dimensionality						
Input	Projection	PCA	LDA	SDA	LPP	MMPP
$48,000$	100	0.55	0.81	46.3	0.7	48.7

MMPP and obtain the projection matrix $\mathbf{R}$ of dimensions $100 \times 48,000$. In Fig. 4a, the objective function in (17) value versus the number of iterations is plotted, which is monotonically decreasing, verifying that the separating margin increases per iteration. Moreover, the gradient Frobenius norm value after each update is demonstrated in Fig. 4b. In this experiment MMPP required 304 iterations in order to sufficiently decrease the gradient norm and reach the convergence point (i.e. to satisfy the termination condition in (16)). In Table II we show the recorded CPU training time, measured in seconds, required by PCA, LDA, SDA, LPP and MMPP algorithms in this dataset. All algorithms have been implemented on Matlab R2012b [44] and the required by each method CPU time during training has been recorded on a 2.66 GHz and 8 GB RAM computer. As it can be observed since PCA, LDA and LPP all solve a Generalized Eigenvalue Problem (GEP) have the shortest training times. SDA although it also solves a GEP problem its training time is significantly higher since it requires to determine the optimal subclasses partition using the stability criterion [42] which is costly. MMPP required the highest training time in the comparison which is attributed to the considered iterative optimization framework.

To visualize the ability of MMPP algorithm to estimate useful subspaces that enhance data discrimination, we run the proposed algorithm, aiming to learn a 2D projection space in a two class toy classification problem, using artificial data. To generate our toy dataset we collected 600 300D samples for each class, with the first class features drawn randomly from a standard normal distribution $\mathcal{N}(0, 1)$ and the second class samples drawn from a $\mathcal{N}(0.2, 1)$ normal distribution and used 100 samples of each class for training, while the rest were used to compose the toy test set. Fig. 5 shows the 2D projection of the two classes training data samples after different iterations of the MMPP algorithm, where circled samples denote the identified support vectors. As can be observed, the proposed algorithm was able, after a few iterations, to perfectly separate linearly the two classes, by continuously maximizing the separating margin. Moreover, as a side effect of MMPP algorithm, we observed that the SVM training process converges faster and into a more sparse solution after each iteration of MMPP algorithm, since the number of identified support vector decreases as class discrimination increases.

C. Face Recognition in LFW Image Set

LFW image set realistically simulates the variability evident to unconstrained, in the wild, face recognition problems. More precisely, LFW data set is the standard benchmark on face



(a) (b)

Fig. 4. MMPP convergence results using augmented Gabor features derived from half of the Cohn-Kanade images; a) Objective function value versus the number of iterations, b) Gradient Frobenius norm versus the number of algorithm iterations.

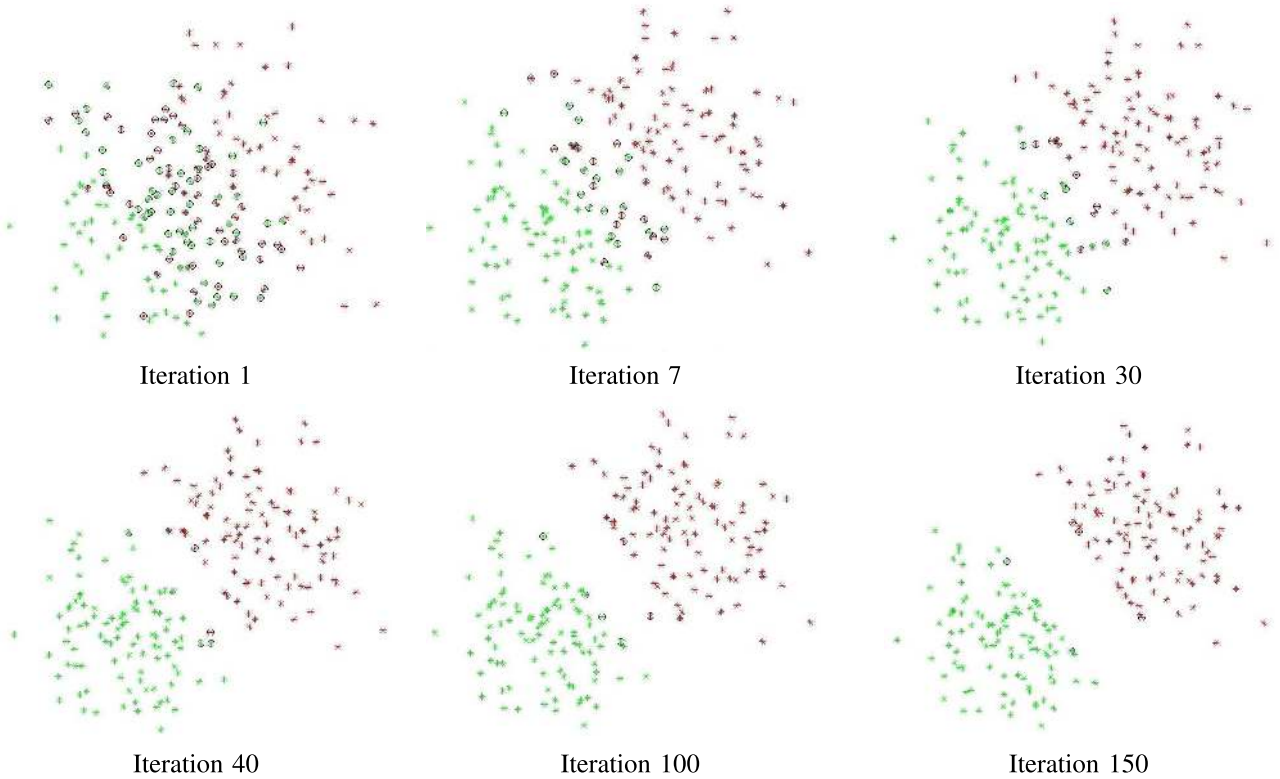


Fig. 5. Training data 2D projection at different iterations of the MMPP algorithm. Circled data samples denote the identified support vectors which reduce during MMPP algorithms convergence.

verification which focuses on pair matching where given two face images, the goal is to decide whether the two samples depict the same individual or not. In total LFW data consist of $13,233$ facial images depicting $5,749$ different individuals of which $4,069$ only have a single image in the database. In order to perform face recognition using the LFW image set we employed a subset of the available data, considering only the face images of those individuals that have more than 30 samples sufficient to contribute both to the training and test set. Thus the dataset we considered from LFW image set consists in total of $2,310$ face images from 32 individuals where for each subject a different number of samples were available varying from 30 to 530 images. The considered facial images were aligned, cropped and scaled to a fixed size of 64×64 pixels using their facial landmarks location and gray scaled. To form our testing set we randomly selected 5 images of each individual, while the remaining were used for training.

Table III summarizes the highest face recognition rates achieved by each method in the comparison. The proposed MMPP algorithm performs the best reaching 93.1% recognition rate followed in order by PCA when we used the image intensity information as the input features, while for the Gabor wavelet representations MMPP reached 95% followed by SDA that attained 93.1%.

TABLE III
BEST FACE RECOGNITION ACCURACY RATES (%) IN LFW IMAGE SET. IN PARENTHESES IT IS SHOWN THE DIMENSION THAT RESULTS IN THE BEST PERFORMANCE FOR EACH METHOD

	SVM	PCA	LDA (31)	SDA	LPP (31)	OLPP (31)	NPE (31)	MMP	RP	MMPP
Intensity	80.6(4,096)	90.3(400)	85.6	89.5(270)	86.3	87.5	86.3	86.3 (300)	75.2(500)	**93.1(200)**
Gabor	82.5(33,640)	90.1(400)	91.3	93.1(350)	91.3	91.9	91.3	88.8 (200)	87.5(500)	**95.0(250)**

TABLE IV
FACE RECOGNITION ACCURACY RATES (%) IN THE EXTENDED YALE B DATABASE. IN PARENTHESES IT IS SHOWN THE DIMENSION THAT RESULTS IN THE BEST PERFORMANCE FOR EACH METHOD

	SVM (32,256)	PCA	LDA (37)	SDA	LPP (37)	OLPP (37)	NPE (37)	MMP	RP	MMPP
Train 10%	90.6	90.8(255)	97.0	96.8(150)	97.0	**97.2**	97.0	94.9(300)	90.9(400)	**97.2(150)**
Train 30%	94.5	95.5(300)	99.7	**99.8(271)**	99.7	99.7	99.7	99.5(300)	96.0(350)	**99.8(500)**
Train 50%	94.7	96.2(500)	**100.0**	**100.0(300)**	**100.0**	99.8	99.9	99.5(300)	96.8(300)	**100.0(150)**

TABLE V
FACE RECOGNITION ACCURACY RATES (%) IN THE AR DATABASE. IN PARENTHESES IT IS SHOWN THE DIMENSION THAT RESULTS IN THE BEST PERFORMANCE FOR EACH METHOD

		SVM	PCA	LDA (99)	SDA	LPP (99)	OLPP (99)	NPE (99)	MMP	RP	MMPP
Exp 1	Intensity	82.0(19,800)	91.0(300)	93.3	91.5(300)	**93.5**	**93.5**	**93.5**	91.0(250)	88.3(500)	93.3(100)
	Gabor	85.6(88,000)	93.3(399)	96.5	95.8(321)	96.7	96.7	96.5	93.5(350)	89.4(500)	**97.2(350)**
Exp 2	Intensity	79.7(19,800)	85.4(500)	88.7	89.4(400)	90.1	**93.0**	90.4	83.7(350)	85.6(500)	91.0(100)
	Gabor	81.9(88,000)	91.7(500)	92.4	93.7(500)	92.7	93.9	93.1	89.7(300)	86.9(500)	**96.2(250)**
Exp 3	Intensity	76.4(19,800)	82.4(500)	85.2	86.0(250)	85.1	**87.9**	84.7	79(300)	81.3(500)	87.0(200)
	Gabor	80.2(88,000)	89.2(500)	89.9	91.7(500)	90.1	91.0	89.3	85.6(350)	81.5(500)	**93.4(300)**

D. Face Recognition in the Extended Yale B Database

The Extended Yale B database consists of $2,414$ frontal facial images of 38 individuals, captured under 64 different laboratory controlled lighting conditions. The database version used in this experimental evaluation has been manually aligned, cropped and then resized to 168×192 pixels by the database creators. For our experimental comparison, we have considered three different experimental settings, by randomly selecting 10%, 30% and 50% of the available images of each subject for training, while the rest of the images were used for testing. In this experiment we did not exploit the augmented Gabor features, since the recognition accuracy rates attained using the facial image intensity values as our classification features were already sufficiently high. Table IV presents the highest face recognition rate achieved by each method. As can be observed, the proposed MMPP method achieves the best performance across all considered experiments. Moreover, LDA, LPP and OLPP, since they are all based in the Fisher discriminant ratio, were able to handle the lighting variations in the Extended Yale B dataset [7].

E. Face Recognition in the AR Database

The AR database is much more challenging than the Extended Yale B dataset and exhibits significant variations among its facial image samples. It contains color images corresponding to 126 different subjects, depicting their frontal facial view under different facial expressions, illumination conditions and occlusions (sunglasses and scarf). For this experiment we used the pre-aligned and cropped version of the AR database containing in total $2,600$ facial images of size 120×165 pixels, corresponding to 100 different subjects captured during two sessions, separated by two weeks time. Thus, 13 images are available for each subject per each session.

In order to investigate MMPP algorithms robustness, we have conducted three different experiments with increasing degree of difficulty. For the first experiment (Exp 1), we formed our training set by considering only those facial images with illumination variations captured during the first session, while for testing we considered the respective images captured during the second recording session. For the second experiment (Exp 2), we used facial images with both varying illumination conditions and facial expressions from the first session for training and the respective images from the second session for testing. Finally, for the third experiment (Exp 3), we used all the first session images for training and the rest for testing.

Table V summarizes the highest attained recognition rate and the respective subspace dimensionality, by each method in each performed experiment. The proposed method achieved

TABLE VI
OBJECT RECOGNITION ACCURACY RATES (%) IN THE ETH-80 DATABASE. IN PARENTHESES IT IS SHOWN THE DIMENSION THAT RESULTS IN THE BEST PERFORMANCE FOR EACH METHOD

	SVM	PCA	LDA	SDA	LPP	OLPP	NPE	MMP	RP	MMPP
ETH-80	$80.3(16,384)$	$81.9(20)$	$74.4(7)$	$79.8(300)$	$74.2(7)$	$74.4(7)$	$74.8(7)$	$80.8(100)$	$79.4(200)$	$\mathbf{84.6(80)}$

recognition rates equal to 93.3% 91% and 87% in each experiment respectively using the facial image intensities, while for the augmented Gabor features attained recognition rates equal to 97.2% 96.2% and 93.4% which are the best or the second best among all examined methods.

F. Object Recognition in the ETH-80 Image Dataset

ETH-80 image dataset [40] depicts 80 objects divided into 8 different classes, where for each object 41 images have been captured from different view points, spaced equally over the upper viewing hemisphere. Thus, the database contains 3,280 images in total. For this experiment we used the cropped and scaled to a fixed size of 128×128 pixels binary images containing the contour of each object. In order to form our training set we randomly picked 25 binary images of each object, while the rest were used for testing. Table VI shows the highest attained object recognition accuracy rate by each method and the respective subspace dimensionality. Again MMPP outperformed in this experiment attaining the highest object recognition rate of 84.6%. It is significant to note that all discriminant dimensionality reduction algorithms in our comparison, based on Fisher discriminant ratio (i.e. LDA, LPP, OLPP and NPE) attained a reduced performance compared against the baseline approach which is feeding directly the initial high dimensional feature vectors to the linear SVM for classification. This can be attributed to the fact that since each category in the ETH-80 dataset includes images depicting 10 different objects captured from various view angles, data samples inside classes span large in-class variations. As a result all the aforementioned methods which have the Gaussian data distribution optimality assumption [17], [45] fail to identify appropriate discriminant projection directions. In contrast to the proposed MMPP method which depends only on the support vectors and the overall data samples distribution inside classes does not affect its performance.

VI. CONCLUSION

We proposed a discrimination enhancing subspace learning method called Maximum Margin Projection Pursuit algorithm that aims to identify a low dimensional projection subspace where samples form classes that are separated with maximum margin. The proposed method is an iterative alternate optimization algorithm that computes the maximum margin projections exploiting the separating hyperplanes obtained from training an SVM classifier in the identified low dimensional space. We also demonstrated the non-linear extension of our algorithm that identifies a projection matrix that separates different classes in the feature space with maximum margin. Finally we showed that it outperforms current state-of-the-art linear data embedding methods on challenging computer vision recognition tasks such as face, expression and object recognition on several popular datasets.

REFERENCES

[1] I. T. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer-Verlag, 1986.

[2] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[3] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[4] X. He and P. Niyogi, "Locality preserving projections," in *Advances in Neural Information Processing Systems*, vol. 16. Cambridge, MA, USA: MIT Press, 2003.

[5] D. Cai, X. He, J. Han, and H.-J. Zhang, "Orthogonal laplacianfaces for face recognition," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3608–3614, Nov. 2006.

[6] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood preserving embedding," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Oct. 2005, pp. 1208–1213.

[7] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[8] X. He, D. Cai, and J. Han, "Learning a maximum margin subspace for image retrieval," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 2, pp. 189–201, Feb. 2008.

[9] T. J. F. Wang, B. Zhao, and C. Zhang, "Unsupervised large margin discriminative projection," *IEEE Trans. Neural Netw.*, vol. 22, no. 9, pp. 1446–1456, Sep. 2011.

[10] A. Zien and J. Q. Candela, "Large margin non-linear embedding," in *Proc. 22nd Int. Conf. Mach. Learn. (ICML)*, Aug. 2005, pp. 1060–1067.

[11] V. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer-Verlag, 1995.

[12] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[13] A. Majumdar and R. K. Ward, "Robust classifiers for data reduced via random projections," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 5, pp. 1359–1371, Oct. 2010.

[14] Q. Shi, C. Shen, R. Hill, and A. van den Hengel, "Is margin preserved after random projection?" in *Proc. 29th Int. Conf. Mach. Learn. (ICML)*, 2012, pp. 1–8.

[15] S. Paul, C. Boutsidis, M. Magdon-Ismail, and P. Drineas, "Random projections for support vector machines," in *Proc. Int. Conf. Artif. Intell. (AISTATS)*, 2013, pp. 498–509.

[16] J. K. Pillai, V. M. Patel, R. Chellappa, and N. K. Ratha, "Secure and robust iris recognition using random projections and sparse representations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1877–1893, Sep. 2011.

[17] M. Zhu and A. M. Martinez, "Subclass discriminant analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1274–1286, Aug. 2006.

[18] X.-W. Chen and T. Huang, "Facial expression recognition: A clustering-based approach," *Pattern Recognit. Lett.*, vol. 24, nos. 9–10, pp. 1295–1302, 2003.

[19] R. Fletcher, *Practical Methods of Optimization*, 2nd ed. New York, NY, USA: Wiley, 1987.

[20] K. R. Varshney and A. S. Willsky, "Learning dimensionality-reduced classifiers for information fusion," in *Proc. 12th Int. Conf. Inf. Fusion*, Jul. 2009, pp. 1881–1888.

[21] D.-S. Pham and S. Venkatesh, "Robust learning of discriminative projection for multicategory classification on the Stiefel manifold," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–7.

[22] K. Torkkola, "Feature extraction by non parametric mutual information maximization," *J. Mach. Learn. Res.*, vol. 3, pp. 1415–1438, Mar. 2003.

[23] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA, USA: Athena Scientific, 1999.
[24] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Comput.*, vol. 19, no. 10, pp. 2756–2779, 2007.
[25] C. Lin and J. J. Moré, "Newton's method for large bound-constrained optimization problems," *SIAM J. Optim.*, vol. 9, no. 4, pp. 1100–1127, 1999.
[26] J. C. Platt, N. Cristianini, and J. Shawe-Taylor, "Large margin DAGs for multiclass classification," in *Advances in Neural Information Processing Systems*, vol. 12. Cambridge, MA, USA: MIT Press, 2000, pp. 547–553.
[27] J. Weston and C. Watkins, "Support vector machines for multi-class pattern recognition," in *Proc. Eur. Symp. Artif. Neural Netw.*, Apr. 1999, pp. 1–6.
[28] V. N. Vapnik, *Statistical Learning Theory*. New York, NY, USA: Wiley, 1998.
[29] K. Crammer and Y. Singer, "On the learnability and design of output codes for multiclass problems," *Mach. Learn.*, vol. 47, nos. 2–3, pp. 201–233, May 2002.
[30] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *J. Mach. Learn. Res.*, vol. 2, pp. 265–292, Mar. 2001.
[31] S. Nikitidis, N. Nikolaidis, and I. Pitas, "Multiplicative update rules for incremental training of multiclass support vector machines," *Pattern Recognit.*, vol. 45, no. 5, pp. 1838–1852, May 2012.
[32] B. Schölkopf *et al.*, "Input space versus feature space in kernel-based methods," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1000–1017, Sep. 1999.
[33] K. R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–201, Mar. 2001.
[34] B. Schölkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2002.
[35] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
[36] T. Kanade, J. F. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, Mar. 2000, pp. 46–53.
[37] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Univ. Massachusetts, Amherst, MA, USA, Tech. Rep. 7-49, 2007.
[38] A. S. Georghiades, P. N. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.
[39] A. M. Martinez and R. Benavente, "The AR face database," CVC, Tech. Rep. 24, 1998.
[40] B. Leibe and B. Schiele, "Analyzing appearance and contour based methods for object categorization," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2003, pp. 409–415.
[41] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 467–476, Apr. 2002.
[42] A. M. Martinez and M. Zhu, "Where are linear feature extraction methods applicable?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 12, pp. 1934–1944, Dec. 2005.
[43] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Jun. 2008.
[44] *MATLAB, Version R2012b*. MathWorks Inc., Natick, MA, USA, 2012.
[45] F. De la Torre and T. Kanade, "Multimodal oriented discriminant analysis," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Aug. 2005, pp. 177–184.
[46] T. Joachims, "Training linear SVMS in linear time," in *Proc. 12th ACM Int. Conf. Knowl. Discovery Data Mining*, Aug. 2006, pp. 217–226.
[47] T. Joachims, T. Finley, and C.-N. J. Yu, "Cutting-plane training of structural SVMS," *Mach. Learn.*, vol. 77, no. 1, pp. 27–59, 2009.
[48] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Process. Lett.*, vol. 9, no. 3, pp. 293–300, 1999.
[49] E. Meyers and L. Wolf, "Using biologically inspired features for face processing," *Int. J. Comput. Vis.*, vol. 76, no. 1, pp. 93–104, 2008.
[50] L. Wolf, T. Hassner, and Y. Taigman, "Effective unconstrained face recognition by combining multiple descriptors and learned background statistics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 1978–1990, Oct. 2011.

Symeon Nikitidis received the B.Sc. and Ph.D. degrees in informatics from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2004 and 2013, respectively, and the M.Sc. degree in advanced computing from the University of Glasgow, Glasgow, U.K., in 2005. From 2006 to 2012, he was a Research and Teaching Assistant with the Department of Informatics, Aristotle University of Thessaloniki, and since 2012, he has been a Research Associate with the Department of Computing, Imperial College London, London, U.K. His current research interests include statistical machine learning, digital signal and image processing, pattern recognition, and computer vision.

Anastasios Tefas (M'04) received the B.Sc. and Ph.D. degrees in informatics from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1997 and 2002, respectively. Since 2013, he has been an Assistant Professor with the Department of Informatics, Aristotle University of Thessaloniki, where he was a Lecturer from 2008 to 2012. From 2006 to 2008, he was an Assistant Professor with the Department of Information Management, Technological Institute of Kavala, Kavala, Greece. From 2003 to 2004, he was a temporary Lecturer with the Department of Informatics, University of Thessaloniki. From 1997 to 2002, he was a Researcher and Teaching Assistant with the Department of Informatics, University of Thessaloniki. He was involved in 12 research projects financed by the national and European funds. He has co-authored 40 journal papers, 120 papers in international conferences, and contributed seven chapters to edited books in his area of expertise. Over 2 150 citations have been recorded to his publications and his H-index is 23 according to Google Scholar. His current research interests include computational intelligence, pattern recognition, statistical machine learning, digital signal and image processing, and computer vision.

Ioannis Pitas (SM'94–F'07) received the Diploma and Ph.D. degrees in electrical engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, where he has been a Professor with the Department of Informatics since 1994. He served as a Visiting Professor at several universities. His current interests are in the areas of image/video processing, intelligent digital media, machine learning, human centered interfaces, affective computing, computer vision, 3D imaging, and biomedical imaging. He has authored over 750 papers; contributed in 39 books in his areas of interest; and edited, authored, or co-authored another nine books. He has also been an invited speaker and/or a Program Committee Member of many scientific conferences and workshops. In the past, he served as an Associate Editor or a Co-Editor of eight international journals, and the General or Technical Chair of four international conferences (including the 2001 International Conference on Image Processing). He was involved in 68 research and development projects, primarily funded by the European Union, and is/was a Principal Investigator/Researcher of 40 such projects. He has over 18 400 citations (Google scholar), over 6 250 (Scopus) to his work, and has an H-index of over 64 (Google scholar) and over 38 (Scopus).