Novel Multiclass Classifiers Based on the Minimization of the Within Class Variance

Irene Kotsia[†], Stefanos Zafeiriou[*] and Ioannis Pitas[†], Fellow IEEE

[†]Aristotle University of Thessaloniki
Department of Informatics
Box 451
54124 Thessaloniki, Greece
[*]Imperial College London
Department of Electrical and Electronic Engineering
London, UK
Address for correspondence :
Professor Ioannis Pitas
Aristotle University of Thessaloniki
54006 Thessaloniki
GREECE
Tel. ++ 30 231 099 63 04
Fax ++ 30 231 099 63 04
email: pitas@aiia.csd.auth.gr

Abstract

In this paper, a novel class of multiclass classifiers inspired by the optimization of Fisher's discriminant ratio and the Support Vector Machine (SVM) formulation, is introduced. The optimization problem of the so-called Minimum Within-Class Variance Multiclass Classifiers (MWCVMC) is formulated and solved in arbitrary Hilbert spaces, defined by Mercer's kernels, in order to find multiclass decision hyperplanes/surfaces. Afterwards, MWCVMCs are solved using indefinite kernels and dissimilarity measures via pseudo-Euclidean embedding. The power of the proposed approach is firstly demonstrated in the facial expression recognition of the seven basic facial expressions (i.e., anger, disgust, fear, happiness, sadness and surprise plus the neutral state) problem in the presence of partial facial occlusion by using a pseudo-Euclidean embedding of Hausdorff distances and the MWCVMC. The experiments indicated a recognition accuracy rate achieved up to 99%. The MWCVMC classifiers are also applied to face recognition and other classification problems using Mercer's kernels.

Index Terms

Fisher's Linear Discriminant Analysis, Multiclass Classifiers, Support Vector Machines, Mercer's kernels, pseudo-Euclidean embedding, Facial Expression Recognition, Face Recognition.

I. INTRODUCTION

The best studied techniques for binary pattern classification include Fisher's Linear Discriminant analysis (FLDA) [1], its nonlinear counterpart, the so-called Kernel-Fisher's Discriminant Analysis (KFDA) [2], [3] and Support Vector Machines (SVMs) [4]. A combination of SVMs and FLDA has been performed in [5], where a two class classifier has been constructed, inspired by the optimization of the Fisher's discriminant ratio and the SVMs separability constraints. More precisely, motivated by the fact that the Fisher's discriminant optimization problem for two classes is a constraint least-squares optimization problem [2], [5], [6], the problem of minimizing the within-class variance has been reformulated, so that it can be solved by constructing the optimal separating hyperplane for both separable and nonseparable cases. The classifier, proposed in [5], has been applied successfully in order to weight the local similarity value of the elastic graphs nodes according to their corresponding discriminant power for frontal face verification. It has been also shown there that it outperforms the typical maximum margin SVMs in the specific problem.

In [5], the proposed classifier has been developed only for two class problems. Moreover, only the linear case has been considered and only when the number of training vectors is larger than the feature dimensionality (i.e., when the within-class scatter matrix of the samples is not singular). An effort to extend the two class classifiers of [5] in order to solve multiclass classification problems has been performed in [7]. The limitation of the multiclass classifier constructed in [7] is that its optimization problem has not been formally defined in Hilbert spaces, but has been considered only for cases in which the within-class scatter matrix of the data is invertible. The classifiers proposed in [7] have been shown to outperform the typical maximum margin SVMs in the recognition of the

six basic facial expressions by large margins.

A lot of research has been conducted regarding facial expression recognition in the past fifteen years [8]. The facial expressions under examination were defined by psychologists as a set of six basic facial expressions (anger, disgust, fear, happiness, sadness, and surprise) [9]. The interested reader may refer to [7], [10], [11] and in the references therein, regarding the various technologies developed for facial expression recognition. In the system proposed in [7], the Candide grid [12] is manually placed on the neutral image and afterwards tracked until the fully expressive video frame is reached. The vectors of the Candide node deformations are the features that have been used for facial expression recognition. The system requires the detection of the neutral facial expression prior to tracking and recognition. Highly related methods with the one proposed in [7] have been also proposed in [13] and [14].

In this paper, a general multiclass solution of the optimization problem proposed in [5], [7] is presented. The problem is solved in arbitrary Hilbert spaces built using Mercer's kernels, without having to assume the invertibility of the within-class scatter matrix neither in the input nor in the Hilbert space. In this way, a new class of multiclass decision hyperplanes/surfaces is defined. In order to build our classifiers in arbitrary dimensional Hilbert spaces we use a method similar to the one proposed in [3]. In [3] a framework for solving the Fisher's discriminant optimization problem (the KFDA optimization problem) using kernels has been proposed. That is, in [3] it has been shown that by using KPCA it is feasible to solve KFDA using kernels and that under KPCA the nonlinear Fisher discriminant analysis optimization problem with kernels is transformed into an equivalent linear (without kernels) optimization problem that produces the so-called Complete Kernel Fisher Discriminant Analysis (CKFDA). Since the approach proposed in this paper requires the solution of a quite different optimization problem than the one in [3] (i.e., the optimization problem in [3] is solved via eigenanalysis and our problem is a quadratic optimization problem), we explicitly prove that the framework in [3] can be safely applied in our case for providing solutions to proposed classifiers. Moreover, we provide some insights of the relationship between the proposed multiclass classifiers and the classifiers proposed in [3].

Afterwards, the problem is solved using indefinite kernels and/or dissimilarity measures with the help of pseudo-Euclidean embedding. The extension of the proposed classifiers using dissimilarity measures for facial expression recognition problems is motivated by the following. In [7] facial expression recognition has been performed by classifying the displacements of the grid nodes between the neutral and the expressive grid. In that case the knowledge of the neutral state is required a-priori. In order to be able to recognize the neutral state, as well as, the other expressions we had to deal with directly comparing grids (and not grid displacements). The grids consist of a set of points and some of the most widely used measures for comparing point sets that are also robust to a series of manipulations (i.e., partial occlusion etc) is the family of Hausdorff distances (which are dissimilarity measures). Thus, we had to successfully combine the multiclass classifiers (which are naturally defined in Euclidean spaces) with pseudo-Euclidean spaces defined by dissimilarity measures. By using the proposed classifier in pseudo-Euclidean spaces, combined with Haussdorf distances, the recognition

of the six basic facial expressions plus the neutral state is achieved.

The use of dissimilarity measures and indefinite kernels has gained significant attention in the research community due to their good performance in various pattern recognition applications [15], [16], [17], [18]. In [15], various classifiers, like two-class FLDA and maximum margin SVMs, have been designed in various pseudo-Euclidean spaces. For more details on the geometry of Euclidean and pseudo-Euclidean spaces the interested reader may refer to [19], [20], [21], [22], [23]. In [16], [18] indefinite kernels have been used for feature extraction to boost the performance of face recognition. The geometric interpretation of maximum margin SVMs with indefinite kernels has been given in [17].

In summary, the contributions of this paper are:

- the presentation of the Minimum Within-Class Variance Multiclass Classifiers (MWCVMC) in their general form for multiclass classification problems using the multiclass SVM formulation in [4], [24], the exploration of the relationship with SVMs and with Fisher Linear Discriminant analysis;
- the generalization of MWCVMC in arbitrary Hilbert spaces, using Mercer's kernels in order to define a novel class of non-linear decision surfaces;
- the solution of MWCVMC using indefinite kernels and pseudo-Euclidean embedding..

Finally, the power of the proposed classifiers is demonstrated in various classification problems. In order to show the potentials of the proposed MWCVMCs we apply:

- Mercer's kernels, like polynomial kernels, for face recognition and for various other classification problems using multiclass datasets from UCI repository [25]
- dissimilarity measures with pseudo-Euclidean embedding for the recognition of seven basic facial expressions.

The rest of this paper is organized as follows. The problem is stated in Section II. The novel class of multiclass classifiers in Hilbert spaces is developed in Section III. The proposed classifier in pseudo-Euclidean spaces are described in Section IV. The application of the novel classifiers in facial expression, face recognition and other classification problems is demonstrated in Section V. Conclusions are drawn in Section VI.

II. Problem Statement

Let $\mathcal{U}$ be a training data set with finite number of elements $\mathcal{U} = \{\mathbf{x}_i, i \in \{1, \ldots, N\}\}$, whose elements belong to two different classes $\mathcal{U}_1$ and $\mathcal{U}_2$, containing training data samples (feature vectors) $\mathbf{x}_i \in \Re^M$ and class labels $y_i \in \{1, -1\}$. The simplest way to separate these classes is by finding a separating hyperplane:

$$\mathbf{w}^T\mathbf{x} + b = 0 \tag{1}$$

where $\mathbf{w} \in \Re^M$ is the normal vector of the hyperplane and $b \in \Re$ is the corresponding scalar term of the hyperplane, also known as bias term [5]. The decision whether a test sample $\mathbf{x}$ belongs to one of the different classes $\mathcal{U}_1$ or $\mathcal{U}_2$ is taken by using the linear decision function $g_{\mathbf{w},b}(\mathbf{x}) = \mathrm{sign}(\mathbf{w}^T\mathbf{x}+b)$, also known as canonical decision hyperplane [4].

A. Fisher's Linear Discriminant Analysis

The best known pattern classification algorithm for separating these classes is the one that finds a decision hyperplane that maximizes the Fisher's discriminant ratio, also known as Fisher's Linear Discriminant Analysis (FLDA):

$$\max_{\mathbf{w},b} \frac{\mathbf{w}^T\mathbf{S}_b\mathbf{w}}{\mathbf{w}^T\mathbf{S}_w\mathbf{w}}, \tag{2}$$

where the matrix $\mathbf{S}_w$ is the within-class scatter matrix defined as:

$$\mathbf{S}_w = \sum_{\mathbf{x}\in\mathcal{U}_1}(\mathbf{x}-\mathbf{m}_1)(\mathbf{x}-\mathbf{m}_1)^T + \sum_{\mathbf{x}\in\mathcal{U}_2}(\mathbf{x}-\mathbf{m}_2)(\mathbf{x}-\mathbf{m}_2)^T. \tag{3}$$

$\mathbf{m}_1$ and $\mathbf{m}_2$ are the mean sample vectors for the classes $\mathcal{U}_1$ and $\mathcal{U}_2$, respectively. The matrix $\mathbf{S}_b$ is the between-class scatter matrix defined in the two class case as:

$$\mathbf{S}_b = N_1(\mathbf{m}-\mathbf{m}_1)(\mathbf{m}-\mathbf{m}_1)^T + N_2(\mathbf{m}-\mathbf{m}_2)(\mathbf{m}-\mathbf{m}_2)^T \tag{4}$$

$$= N_1N_2(\mathbf{m}_1-\mathbf{m}_2)(\mathbf{m}_1-\mathbf{m}_2)^T \tag{5}$$

where N_1 and N_2 are the cardinalities of the classes $\mathcal{U}_1$ and $\mathcal{U}_2$, respectively and $\mathbf{m}$ is the overall mean vector of the set $\mathcal{U}$. The solution of the optimization problem (2) can be found in [1]. It can be proven that the corresponding separating hyperplane is the optimal Bayesian solution, when the samples of each class follow Gaussian distributions with same covariance matrices [1].

B. Support Vector Machines

In the SVM case, the optimal separating hyperplane is the one which separates the training data with maximum margin [4]. The SVM optimization problem is defined as:

$$\min_{\mathbf{w},b} \frac{1}{2}\mathbf{w}^T\mathbf{w} \tag{6}$$

subject to the separability constraints:

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, \;\; i = 1,\ldots,N. \tag{7}$$

C. Minimum Within-Class Variance Two-Class Classifier

In [5], inspired by the maximization of the Fisher's discriminant ratio (2) and the SVM separability constraints, the Minimum Within-Class Variance Two-Class Classifier (MWCVTCC) has been introduced. The MWCTCC optimization problem is defined as:

$$\min_{\mathbf{w},b} \mathbf{w}^T\mathbf{S}_w\mathbf{w}, \;\; \mathbf{w}^T\mathbf{S}_w\mathbf{w} > 0, \tag{8}$$

subject to the separability constraints (7). Thus, the within-class variance of the training samples is minimized when projected to the direction $\mathbf{w}$ subject to the constraint that the samples are separable along this projection. More details about the motivations of the optimization problem (8) can be found in [5].

If training errors are allowed, the optimum decision hyperplane is found by using the *soft* formulation [4], [5] and solving the following optimization problem:

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \mathbf{w}^T\mathbf{S}_w\mathbf{w} + C\sum_{i=1}^{N}\xi_i,\ \ \mathbf{w}^T\mathbf{S}_w\mathbf{w} > 0 \tag{9}$$

subject to the separability constraints:

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i\ , \xi_i \geq 0,\ \ i = 1, \ldots, N \tag{10}$$

where $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_N]$ is the vector of the non-negative slack variables and C is a given constant that defines the cost of the errors after the classification. Larger values of C correspond to higher penalty assigned to errors. The linearly separable case (8) can be found when choosing $C \to \infty$.

The solution of the minimization of (9), subject to the constraints (10), is given by the saddle point of the Lagrangian:

$$L(\mathbf{w}, b, \mathbf{a}, \mathbf{r}, \boldsymbol{\xi}) = \mathbf{w}^T\mathbf{S}_w\mathbf{w} + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}a_i[y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i] - \sum_{i=1}^{N}r_i\xi_i, \tag{11}$$

where $\mathbf{a} = [a_1, \ldots, a_N]^T$ and $\mathbf{r} = [r_1, \ldots, r_N]^T$ are the vectors of the Lagrangian multipliers for the constraints (10). The Karush-Kuhn-Tucker (KKT) conditions [26] imply that for the optimal choice of $\mathbf{w}, \mathbf{a}, \mathbf{r}, b, \boldsymbol{\xi}$, the following hold:

$$\begin{aligned}
\nabla_{\mathbf{w}}L|_{\mathbf{w}=\mathbf{w}_o} &= \mathbf{0} \Leftrightarrow \mathbf{S}_w\mathbf{w}_o = \tfrac{1}{2}\textstyle\sum_{i=1}^{N} a_{i,o}y_i\mathbf{x}_i \\
\tfrac{\partial L}{\partial b}|_{b=b_o} &= 0 \Leftrightarrow \mathbf{a}_o^T\mathbf{y} = 0 \\
\tfrac{\partial L}{\partial \xi_i}|_{\xi_i=\xi_{i,o}} &= 0 \Leftrightarrow r_{i,o} = C - a_{i,o} \\
r_{i,o} \geq 0, 0 \leq a_{i,o} &\leq C, \xi_{i,o} \geq 0, r_{i,o}\xi_{i,o} = 0 \\
y_i(\mathbf{w}_o^T\mathbf{x}_i + b_o) - 1 + \xi_{i,o} &\geq 0, a_{i,o}\{y_i(\mathbf{w}_o^T\mathbf{x}_i + b_o) - 1 + \xi_{i,o}\} = 0
\end{aligned} \tag{12}$$

where the subscript o denotes the optimal case and $\mathbf{y} = \{y_1, \ldots, y_N\}$ is the vector denoting the class labels. If the matrix $\mathbf{S}_w$ is invertible, i.e. the feature vector dimensionality is less or equal to the number of samples minus two ($M \leq N-2$), the optimal normal vector $\mathbf{w}$ of the hyperplane is given by (12):

$$\mathbf{S}_w\mathbf{w}_o = \frac{1}{2}\sum_{i=1}^{N}a_{i,o}y_i\mathbf{x}_i \Leftrightarrow \mathbf{w}_o = \frac{1}{2}\mathbf{S}_w^{-1}\sum_{i=1}^{N}a_{i,o}y_i\mathbf{x}_i. \tag{13}$$

By replacing (13) to (11) and using the KKT conditions (12), the constraint optimization problem (9) is reformulated to the Wolf dual problem:

$$\begin{aligned}
\max_{\mathbf{a}} f(\mathbf{a}) = \mathbf{1}_N^T\mathbf{a} - \frac{1}{2}\mathbf{a}^T\mathbf{Q}\mathbf{a} \quad \text{subject to} \\
0 \leq a_i \leq C,\ i = 1, \ldots, N,\ \ \mathbf{a}^T\mathbf{y} = 0
\end{aligned} \tag{14}$$

where $\mathbf{1}_N$ is a N dimensional vector of ones and $[\mathbf{Q}]_{i,j} = \frac{1}{2}y_iy_j\mathbf{x}_i^T\mathbf{S}_w^{-1}\mathbf{x}_j$. It is worth noting here that, for the typical maximum margin SVM problem [4], the matrix $\mathbf{Q}$ has elements $[\mathbf{Q}]_{i,j} = y_iy_j\mathbf{x}_i^T\mathbf{x}_j$. The corresponding decision function is given by:

$$g(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x} + b) = \text{sign}(\frac{1}{2}\sum_{i=1}^{N}a_{i,o}y_i\mathbf{x}_i^T\mathbf{S}_w^{-1}\mathbf{x} + b_o). \tag{15}$$

The optimal threshold b_o can be found by exploiting the fact that for all support vectors $\mathbf{x}_i$ with $0 < a_{i,o} < C$, their corresponding slack variables are zero, according to the KKT condition (12). Thus, for any support vector $\mathbf{x}_i$ with $i \in \mathcal{S} = \{i : 0 < a_i < C\}$, the following equation holds:

$$y_i(\frac{1}{2}\sum_{j=1}^{N} y_j a_{j,o}\mathbf{x}_j^T\mathbf{S}_w^{-1}\mathbf{x}_i + b_o) = 1. \tag{16}$$

Averaging over these patterns yields a numerically stable solution:

$$b_o = \frac{1}{N}\sum_{i\in\mathcal{S}}(\frac{1}{2}\sum_{j=1}^{N} y_j a_{j,o}\mathbf{x}_j^T\mathbf{S}_w^{-1}\mathbf{x}_i - y_i). \tag{17}$$

As can be seen, the described MWCVTCC [5] have been proposed for two class problems and define only linear classifiers. Actually, in [5] non-linear decision surfaces have been defined, but there were not the generalization of MWCVTCC in Hilbert spaces. These surfaces will be discussed in Section III-B.

D. Multi-class SVM

Many methods have been proposed for the extension of binary SVMs to multiclass problems [4], [24], [27], [28]. The multiclass SVMs classifiers in [4], [24], [27], [28] are the most elegant multiclass SVM algorithms closely aligned with the principle of always trying to solve problems directly. That principle entails the modification of the SVM objective in such a way that will simultaneously allow the computation of a multiclass classifier learning with kernels [4]. Nevertheless, the theory that will be presented in the next sections can be extended using other multiclass SVM classifiers in a straightforward manner. The interested reader can refer to [4], [24], [27], [29] and the references therein for the formulation and solution of multi-class SVM optimization problems.

Let the training dataset $\mathcal{U}$ to be separated to K disjoint classes $\mathcal{U}_1, \ldots, \mathcal{U}_K$. The training data are $(\mathbf{x}_1, l_1), \ldots, (\mathbf{x}_N, l_N)$ and $l_j \in \{1, \ldots, K\}$ are the class labels of the training vectors. The multi-class SVM problem solves only one optimization problem [27]. It constructs K classification rules, where the $k-$th function $\mathbf{w}_k^T\phi(\mathbf{x}_j) + b_k$ separates the training vectors of the class k from the rest of the vectors, by minimizing the objective function:

$$\min_{\mathbf{w},\mathbf{b},\boldsymbol{\xi}} \qquad \frac{1}{2}\sum_{k=1}^{K}\mathbf{w}_k^T\mathbf{w}_k + C\sum_{j=1}^{N}\sum_{k\neq l_j}\xi_j^k \tag{18}$$

subject to the constraints:

$$\mathbf{w}_{l_j}^T\mathbf{x}_j + b_{l_j} \geq \mathbf{w}_k^T\mathbf{x}_j + b_k + 2 - \xi_j^k \tag{19}$$

$$\xi_j^k \geq 0, \quad j = 1, \ldots, N, \quad k \in \{1, \ldots, K\}\backslash l_j$$

where C is the term that penalizes the training errors. The vector $\mathbf{b} = [b_1 \ldots b_K]^T$ is the bias vector and $\boldsymbol{\xi} = [\xi_1^1, \ldots, \xi_i^k, \ldots, \xi_N^K]^T$ is the slack variable vector. Then, the decision function is:

$$f(\mathbf{x}) = \underset{k=1,\ldots,K}{\operatorname{argmax}}(\mathbf{w}_k^T\mathbf{x} + b_k). \tag{20}$$

For the solution of the optimization problem (18), subject to the constraints (19), someone can refer to [4], [24], [27].

E. Relationship Between The Minimum Within Class Variance Classifiers and Support Vector Machines

In this subsection we will explore the relationship between MWCVTCC and maximum margin SVMs. Let that we define the following optimization problem:

$$\min_{\mathbf{w},b} \frac{1}{2}\mathbf{w}^T\mathbf{S}_w\mathbf{w} \tag{21}$$

under the separability constraints:

$$y_i(\mathbf{w}^T(\mathbf{x}_i - \mathbf{m}) + b) \geq 1, \tag{22}$$

which is the MWCVTCC (under some minor calculations i.e., subtracting the mean vector from all vectors).

Let that the matrix $\mathbf{S}_w$ is non-singular. We consider the transformed vectors $\mathbf{x}_i$ to the vectors $\mathbf{p}_i = \mathbf{S}_w^{-\frac{1}{2}}(\mathbf{x}_i - \mathbf{m})$ and by letting $\mathbf{w} = \mathbf{S}_w^{\frac{1}{2}}\mathbf{g}$, the above optimization problem is reformulated to a maximum margin classifier $(\mathbf{g}, b)$ such that:

$$\min_{\mathbf{g},b} \frac{1}{2}\mathbf{g}^T\mathbf{g} \tag{23}$$

subject to the separability constraints:

$$y_i(\mathbf{g}^T\mathbf{p}_i + b) \geq 1. \tag{24}$$

The above analysis shows that MWCVTCCs are equivalent to maximum margin classifiers when the within class scatter matrix is the identity matrix.

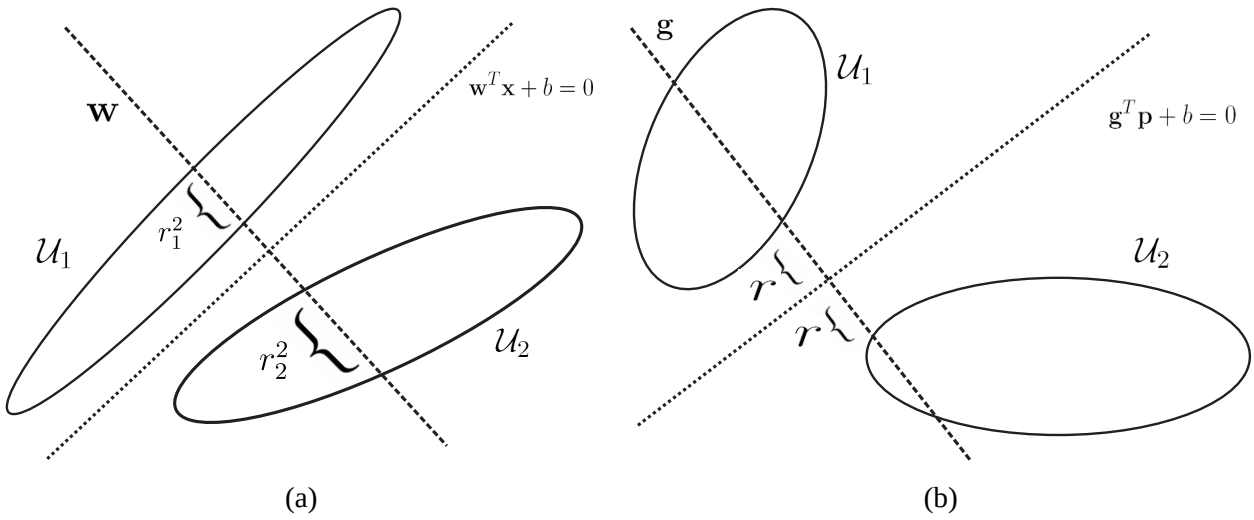


Fig. 1. The geometrical interpretation of minimum within class variance two class classifiers a) the optimization problem (21) subject to the constraints (22) finds the optimum hyperplane $\mathbf{w}^T\mathbf{x} + b$ such that the variances $r_1^2 + r_2^2$ is minimized subject to data separability b) the equivalent optimization problem (23) subject to the constraints (24) is to find a maximum margin SVM hyperplane in a space where $\mathbf{S}_w = \mathbf{I}$ (i.e., maximize $2r$ subject to separability).

The geometric interpretation of the optimization problem (21) subject to the constraints (22) and of the equivalent optimization problem (23) subject to (24) is pictorially described in Figures 1 a) and

b). The optimum hyperplane in the case of the optimization of (21) subject to (22) is demonstrated in Figure 1 a. The optimum hyperplane in this case is the one with normal vector such that $r_1^2 + r_2^2$ is minimized. The equivalent is a maximum margin hyperplane (maximize $2r$) in a normalized space where $\mathbf{S}_w = \mathbf{I}$, as described in Figure 1 b).

Another attempt to relate further MWCVTCCs, maximum margin SVM classifiers and the recently introduced Ellipsoidal kernel machines [30] is through the following. From VC dimension theory for a set of binary classifiers in $\Re^M$ with minimum margin ρ and under the assumption that the data are enclosed in a hypersphere with radius R then the VC dimension h is:

$$h_{Sphere} = \min\{\text{ceil}(\frac{R^2}{\rho^2}), M\} + 1 \tag{25}$$

ceil is the ceiling operator. The VC dimension is directly related to the generalization error [4], [31], [30]. The theory of SVMs has emerged from the above equation. That is, in SVM theory the family of classifiers obtained by the constraint optimization problem (6) maximize the margin, while the constraints (7) ensure empirical error minimization. As can be seen by the generalization error theory [4], [30] the VC dimension depends not only on the margin but also on the diameter of the enclosing hypersphere. The geometric area of a hypersphere in $\Re^M$ with radius R and center $\mathbf{m}$ is defined as $(\mathbf{x}-\mathbf{m})^T(\mathbf{x}-\mathbf{m}) \leq R^2$ or equivalently $(\mathbf{x}-\mathbf{m})^T\mathbf{A}(\mathbf{x}-\mathbf{m}) \leq 1$ with $\mathbf{A}$ being a $M \times M$ diagonal matrix with diagonal elements $\mathbf{A}_{i,i} = \frac{1}{R^2}$.

Let us now consider the enclosing hyperellipse with semi-major axis equal to R. The minimum enclosing hyperellipse is defined as $\mathcal{G}_R = \{\mathbf{x} : (\mathbf{x}-\mathbf{m})^T\mathbf{S}^{-1}(\mathbf{x}-\mathbf{m}) \leq 1\}$ where $\mathbf{S}$ is the covariance matrix of the hyperellipse. From the above observation, it is easy to show that for the VC dimension of a classifier defined in a hyperellipse it is valid that:

$$h_{\mathcal{G}_R} \leq h_{Sphere}. \tag{26}$$

The above can be easily proven by observing that the area defined by the hyperellipse is inside the hypersphere [30]. Suppose the two parallel hyperplanes that define the classifier can shatter l-points for a known margin in the hyperellipse. Then, the exact l-points can be shattered having the same margin in the hypersphere.

As has been shown by the above analysis, the so-called ellipsoidal classifiers in [30] have VC dimension less of equal to the dimension of maximum margin classifiers. The ellipsoidal classifiers minimize the functional $\mathbf{w}^T\mathbf{S}\mathbf{w}$ (instead of the functional $\mathbf{w}^T\mathbf{w}$ for SVMs and $\mathbf{w}^T\mathbf{S}_w\mathbf{w}$ for MWCVTCCs). Thus, the ellipsoidal classifiers [30] are equivalent to maximum margin classifiers subject to the transformation $\mathbf{p}_i = \mathbf{S}^{-\frac{1}{2}}(\mathbf{x}_i - \mathbf{m})$. In MWCVTCCs we use $\mathbf{S}_w^{-\frac{1}{2}}$ instead of $\mathbf{S}^{-\frac{1}{2}}$. The above is a first attempt to relate intuitively the proposed classifiers with maximum margin classifiers and the ellispoidal classifiers in [30].

III. Minimum Within-Class Variance Multiclass Classifiers using Mercer's Kernels

In this Section we describe the way the two class MWCVTCC (described in Section II-C) can be extended to multi-class classifications problems using the multi-class SVM formulation presented

in [4], [24], [27]. The procedure followed in order to generalize in arbitrary Hilbert spaces the optimization problem (9) subject to the constraints (10), using a non-linear function ϕ, so as to define decision surfaces, is also presented. The training data $\mathbf{x}_i$ are initial mapped to an arbitrary Hilbert space under the map $\phi : \Re^M \rightarrow \mathcal{H}$. In this section, only the case in which the mapping ϕ[1] satisfies the Mercer's condition [4] (or conditionality positive kernels) will be taken into consideration. It is not necessary to know the explicit form of the function ϕ, since all the algorithms that will be defined from now onwards require only the close form of the dot products in $\mathcal{H}$, the so called *kernel trick*:

$$h(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y}) \tag{27}$$

where h is called the *kernel function*. The typical kernels used in literature are the polynomial and the Radial Basis Functions (RBF) ones:

$$\begin{aligned} h(\mathbf{x}, \mathbf{y}) &= \phi(\mathbf{x})^T \phi(\mathbf{y}) = (\mathbf{x}^T \mathbf{y} + 1)^d \\ h(\mathbf{x}, \mathbf{y}) &= \phi(\mathbf{x})^T \phi(\mathbf{y}) = e^{\frac{(\mathbf{x}-\mathbf{y})^T (\mathbf{x}-\mathbf{y})}{\gamma^2}}, \end{aligned} \tag{28}$$

where d is a positive integer that is the degree of the polynomial and γ is the spread of the Gaussian kernel.

A. Solution of the optimization problem using Mercer's Kernels

The constrained optimization problem (9) subject to (10) is extended in Hilbert spaces using the multiclass SVM formulation in Section II-D. This novel multi-class classifier is the generalization of the two class problem defined in (9) in arbitrary Hilbert spaces. The within-class scatter matrix of the training vectors is defined in the K-class case as:

$$\mathbf{S}_w^{\Phi} = \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in \mathcal{U}_k} (\phi(\mathbf{x}_i) - \mathbf{m}_k^{\Phi})(\phi(\mathbf{x}_i) - \mathbf{m}_k^{\Phi})^T \tag{29}$$

where $\mathbf{m}_k^{\Phi}$ is the mean vector of the class $\mathcal{U}_k$ i.e, $\mathbf{m}_k^{\Phi} = \frac{1}{N_k} \sum_{\mathbf{x}_i \in \mathcal{U}_k} \phi(\mathbf{x}_i)$.

The modified constraint optimization problem is formulated as:

$$\min_{\mathbf{w}_k, \mathbf{b}, \boldsymbol{\xi}} \qquad \sum_{k=1}^{K} \frac{1}{2} \mathbf{w}_k^T \mathbf{S}_w^{\Phi} \mathbf{w}_k + C \sum_{j=1}^{N} \sum_{k \neq l_j} \xi_j^k \tag{30}$$

subject to the separability constraints in

$$\mathbf{w}_{l_j}^T (\phi(\mathbf{x}_j) - \mathbf{m}^{\Phi}) + b_{l_j} \geq \mathbf{w}_k^T (\phi(\mathbf{x}_j) - \mathbf{m}^{\Phi}) + b_k + 2 - \xi_j^k \ , \xi_j^k \geq 0, \quad j = 1, \ldots, N, \quad k \in \{1, \ldots, K\} \backslash l_j. \tag{31}$$

and inspired by the above constraints we propose a variant where we subtract the mean of each class from the vectors. In this case we have to solve the optimization problem (30) subject to:

$$\mathbf{w}_{l_j}^T (\phi(\mathbf{x}_j) - \mathbf{m}_{l_j}^{\Phi}) + b_{l_j} \geq \mathbf{w}_k^T (\phi(\mathbf{x}_j) - \mathbf{m}_k^{\Phi}) + b_k + 2 - \xi_j^k \ , \xi_j^k \geq 0, \quad j = 1, \ldots, N, \quad k \in \{1, \ldots, K\} \backslash l_j. \tag{32}$$

[1]The following discussion holds for the linear case as well, when $\phi(\mathbf{x}) = \mathbf{x}$ and is interesting since it provides solutions in linear cases when the number of samples is smaller than the dimensionality, i.e. the within-class scatter matrix is singular.

The solution of the constraint optimization problem (30) subject to the constraints (31) can be given by finding the saddle point of the Lagrangian:

$$\begin{aligned} L_1(\mathbf{w}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \sum_{k=1}^{K} \tfrac{1}{2}\mathbf{w}_k^T \mathbf{S}_w^{\Phi} \mathbf{w}_k + C \sum_{i=1}^{N}\sum_{k=1}^{K} \xi_i^k - \\ &- \sum_{i=1}^{N}\sum_{k=1}^{K} \alpha_i^k [(\mathbf{w}_{l_i} - \mathbf{w}_k)^T (\phi(\mathbf{x}_i) - \mathbf{m}^{\Phi}) + b_{l_i} - b_k - 2 + \xi_i^k] - \\ &- \sum_{i=1}^{N}\sum_{k=1}^{K} \beta_i^k \xi_i^k, \end{aligned} \tag{33}$$

where $\boldsymbol{\alpha} = [\alpha_1^1, \ldots, \alpha_i^k, \ldots, \alpha_N^K]$ and $\boldsymbol{\beta} = [\beta_1^1, \ldots, \beta_i^k, \ldots, \beta_N^K]$ are the Lagrangian multipliers for the constraints (31) with:

$$\alpha_i^{l_i} = 0,\ \xi_i^{l_i} = 2,\ \beta_i^{l_i} = 0,\ i = 1, \ldots, N \tag{34}$$

and constraints:

$$\alpha_i^k \geq 0,\ \beta_i^k \geq 0,\ i = 1, \ldots, N,\ k \in \{1, \ldots, K\} \backslash l_i. \tag{35}$$

For the second optimization problem of the variant MWCVMCs (i.e, (30) under the constraints (32)) the corresponding Lagrangian is:

$$\begin{aligned} L_2(\mathbf{w}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \tfrac{1}{2} \sum_{k=1}^{K} \mathbf{w}_k^T \mathbf{S}_w^{\Phi} \mathbf{w}_k + C \sum_{i=1}^{N}\sum_{k=1}^{K} \xi_i^k - \\ &- \sum_{i=1}^{N}\sum_{k=1}^{K} \alpha_i^k [\mathbf{w}_{l_i}^T (\phi(\mathbf{x}_i) - \mathbf{m}_{l_i}^{\Phi}) - \mathbf{w}_k^T (\phi(\mathbf{x}_i) - \mathbf{m}_k^{\Phi}) + b_{l_i} - b_k - 2 + \xi_i^k] - \\ &- \sum_{i=1}^{N}\sum_{k=1}^{K} \beta_i^k \xi_i^k. \end{aligned} \tag{36}$$

The Lagrangian equations (33) and (36) has to be maximized with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ and minimized with respect to $\mathbf{w}$ and $\boldsymbol{\xi}$. In order to produce a more compact equation form let us define the following variables:

$$A_i = \sum_{k=1}^{K} \alpha_i^k, \qquad c_i^k = \begin{cases} 1, & \text{if } l_i = k \\ 0, & \text{if } l_i \neq k. \end{cases} \tag{37}$$

One of the KKT conditions for the Lagrangian (33) requires:

$$\nabla_{\mathbf{w}_k} L_1|_{\mathbf{w}_k = \mathbf{w}_{k,o}} = 0 \Leftrightarrow \mathbf{S}_w^{\Phi} \mathbf{w}_{k,o} = \sum_{i=1}^{N} (c_i^k A_{i,o} - a_{i,o}^k)(\phi(\mathbf{x}_i) - \mathbf{m}^{\Phi}) \tag{38}$$

where $\mathbf{m}^{\Phi} = \frac{1}{N}\sum_{i=1}^{N} \phi(\mathbf{x}_i)$ is the mean vector of the projected samples, and for the second Lagrangian (36):

$$\nabla_{\mathbf{w}_k} L_2|_{\mathbf{w}_k = \mathbf{w}_{k,o}} = 0 \Leftrightarrow \mathbf{S}_w^{\Phi} \mathbf{w}_{k,o} = \sum_{i=1}^{N} (c_i^k A_{i,o} - a_{i,o}^k)(\phi(\mathbf{x}_i) - \mathbf{m}_k^{\Phi}) \tag{39}$$

where the subscript o denotes the optimal parameter choise. Since the Hilbert space $\mathcal{H}$ is of arbitrary dimension, the matrix $\mathbf{S}_w^{\Phi}$ is almost always singular. Thus, the optimal normal vector $\mathbf{w}_{k,o}$ cannot be directly found from (38) or from (39), since the matrix $\mathbf{S}_w^{\Phi}$ cannot be inverted. A solution of the optimization problem (30) subject to the separability constraints (31) (and of (30) subject to (32)) will be provided without having to assume that the within-class scatter matrix of the data is invertible, neither in the input space $\Re^M$, nor in the Hilbert space $\mathcal{H}$. The existence of a solution to this optimization problem will be justified by proving that we can find a mapping that makes the solution feasible. This mapping is the Kernel PCA (KPCA[2]) transform [32].

[2]This is particularly important for the small sample size problem in which the within-class scatter matrix is singular. In the linear case i.e., $\phi(\mathbf{x}) = \mathbf{x}$ the KPCA degenerates to the typical PCA transform.

Let the total scatter matrix $\mathbf{S}_t^{\Phi}$ in the Hilbert space $\mathcal{H}$ be defined as:

$$\mathbf{S}_t^{\Phi} = \sum_{i=1}^{N} (\phi(\mathbf{x}_i) - \mathbf{m}^{\Phi})(\phi(\mathbf{x}_i) - \mathbf{m}^{\Phi})^T. \tag{40}$$

The matrix $\mathbf{S}_t^{\Phi}$ is bounded, compact, positive and a self-adjoint operator in the Hilbert space $\mathcal{H}$. Thus, according to the Hilbert-Schmidt Theorem [26] its eigenvectors system is an orthonormal basis of $\mathcal{H}$. Let $\mathcal{B}^{\Phi}$ and $\mathcal{B}_{\perp}^{\Phi}$ be the complementary spaces spanned by the orthonormal eigenvectors of $\mathbf{S}_t^{\Phi}$ that correspond to non-zero and zero eigenvalues, respectively. An arbitrary vector $\mathbf{w} \in \mathcal{H}$, can be uniquely represented as $\mathbf{w} = \boldsymbol{\varphi} + \boldsymbol{\zeta}$ with $\boldsymbol{\varphi} \in \mathcal{B}^{\Phi}$ and $\boldsymbol{\zeta} \in \mathcal{B}_{\perp}^{\Phi}$. Let us define the linear mapping $L^{\Phi} : \mathcal{H} \rightarrow \mathcal{B}^{\Phi}$ as:

$$\mathbf{w} = \boldsymbol{\varphi} + \boldsymbol{\zeta} \rightarrow \boldsymbol{\varphi}. \tag{41}$$

The following proposition demonstrates that the optimization of the (30), subject to the constraints (31), can be performed in the space $\mathcal{B}^{\Phi}$, instead of $\mathcal{H}$, without any information loss.

Proposition 1. Under the mapping L^{Φ} the optimization problem (30) subject to the constraints (31) is equivalent to:

$$\min_{\boldsymbol{\varphi}_k, \mathbf{b}, \boldsymbol{\xi}} \quad \sum_{k=1}^{K} \frac{1}{2} \boldsymbol{\varphi}_k^T \mathbf{S}_w^{\Phi} \boldsymbol{\varphi}_k + C \sum_{j=1}^{N} \sum_{k \neq l_j} \xi_j^k, \tag{42}$$

subject to the constraints:

$$\begin{aligned} \boldsymbol{\varphi}_{l_j}^T (\phi(\mathbf{x}_j) - \mathbf{m}^{\Phi}) + b_{l_j} &\geq \boldsymbol{\varphi}_k^T (\phi(\mathbf{x}_j) - \mathbf{m}^{\Phi}) + b_k + 2 - \xi_j^k \\ \xi_j^k \geq 0, \quad j = 1, \ldots, N, \quad k &\in \{1, \ldots, K\} \backslash l_j. \end{aligned} \tag{43}$$

The corresponding optimization problem for the MWCVMCs variant is to optimize (42) subject to the constraints:

$$\begin{aligned} \boldsymbol{\varphi}_{l_j}^T (\phi(\mathbf{x}_j) - \mathbf{m}_{l_j}^{\Phi}) + b_{l_j} &\geq \boldsymbol{\varphi}_k^T (\phi(\mathbf{x}_j) - \mathbf{m}_k^{\Phi}) + b_k + 2 - \xi_j^k \\ \xi_j^k \geq 0, \quad j = 1, \ldots, N, \quad k &\in \{1, \ldots, K\} \backslash l_j. \square \end{aligned} \tag{44}$$

A proof of this Proposition can be found in Appendix I.

The optimal decision surfaces of the optimization problem (30) subject to the constraints (31) and of (30) subject to (32) can be found in the reduced space $\mathcal{B}^{\Phi}$ spanned by the non-zero eigenvectors of $\mathbf{S}_t^{\Phi}$. The number of the non-zero eigenvectors of $\mathbf{S}_t^{\Phi}$ is $m \leq N - 1$, Thus, the dimensionality of $\mathcal{B}^{\Phi}$ is $m \leq N - 1$. Therefore, according to the functional analysis theory [33], the space $\mathcal{B}^{\Phi}$ is isomorphic to the $(N-1)$-dimensional Euclidean space $\Re^{N-1}$. The isomorphic mapping is:

$$\boldsymbol{\varphi} = \mathbf{P}\boldsymbol{\eta}, \ \boldsymbol{\eta} \in \Re^{N-1}, \tag{45}$$

where $\mathbf{P}$ is the matrix having as columns the eigenvectors of $\mathbf{S}_t^{\Phi}$ that correspond to non-null eigenvalues. Equation (45) is an one-to-one mapping from $\Re^{N-1}$ onto $\mathcal{B}$.

Under this mapping, the optimization problem is reformulated to:

$$\min_{\boldsymbol{\eta}_k, \mathbf{b}, \boldsymbol{\xi}} \quad \sum_{k=1}^{K} \frac{1}{2} \boldsymbol{\eta}_k^T \tilde{\mathbf{S}}_w \boldsymbol{\eta}_k + C \sum_{j=1}^{N} \sum_{k \neq l_j} \xi_j^k, \tag{46}$$

where $\tilde{\mathbf{S}}_w$ is the within-class scatter matrix of the projected vectors at the non-null KPCA space given by $\tilde{\mathbf{S}}_w = \mathbf{P}^T \mathbf{S}_w^{\Phi} \mathbf{P}$, subject to the constraints:

$$\begin{aligned} \boldsymbol{\eta}_{l_j}^T(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}) + b_{l_j} \geq \boldsymbol{\eta}_k^T(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}) + b_k + 2 - \xi_j^k \\ \xi_j^k \geq 0, \quad j = 1, \ldots, N, \quad k \in \{1, \ldots, K\} \backslash l_j \end{aligned} \tag{47}$$

and for the variant the constraints are:

$$\begin{aligned} \boldsymbol{\eta}_{l_j}^T(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}_{l_j}) + b_{l_j} \geq \boldsymbol{\eta}_k^T(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}_k) + b_k + 2 - \xi_j^k \\ \xi_j^k \geq 0, \quad j = 1, \ldots, N, \quad k \in \{1, \ldots, K\} \backslash l_j \end{aligned} \tag{48}$$

where $\tilde{\mathbf{x}}_i = \mathbf{P}^T \phi(\mathbf{x}_i)$ and $\tilde{\mathbf{m}}_k = \mathbf{P}^T \mathbf{m}_k^{\Phi}$ are the projected vectors to the non-null KPCA space. More details on the calculation of the projections to the KPCA space can be found in [3], [32]. Under mapping (45), the optimal decision surface in $\mathcal{H}$ for the optimization problem (42), subject to (43), can be found by solving the optimization problem (46) subject to (47) in $\Re^{N-1}$. However, the matrix $\tilde{\mathbf{S}}_w$ may still be singular, since its rank is equal to $N - K$. If this is the case, i.e $\tilde{\mathbf{S}}_w$ is singular, it contains K null dimensions. Thus, in order to satisfy the invertibility of $\tilde{\mathbf{S}}_w$ along with the null eigenvectors of $\mathbf{P}$, K more eigenvectors are discarded, which correspond to the lowest non-zero eigenvalues. An alternative way here is to perform eigenanalysis on the singular matrix $\acute{\mathbf{S}}_w$ and remove the eigenvectors that correspond to null eigenvalues (the latter case requires a second eigenanalysis).

The Lagrangian of the optimization problem (46) subject to the constraints (47) is given by:

$$\begin{aligned} L_3(\boldsymbol{\eta}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \quad &= \textstyle\sum_{k=1}^{K} \frac{1}{2} \boldsymbol{\eta}_k^T \tilde{\mathbf{S}}_w \boldsymbol{\eta}_k + C \sum_{i=1}^{N} \sum_{k=1}^{K} \xi_i^k - \\ &- \textstyle\sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_i^k [(\boldsymbol{\eta}_{l_i} - \boldsymbol{\eta}_k)^T (\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}) + b_{l_i} - b_k - 2 + \xi_i^k] - \\ &- \textstyle\sum_{i=1}^{N} \sum_{k=1}^{K} \beta_i^k \xi_i^k. \end{aligned} \tag{49}$$

The search of the saddle point of the Lagrangian (49) is reformulated to the maximization of the Wolf dual problem:

$$W(\boldsymbol{\alpha}) = 2 \sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_i^k + \frac{1}{2} \sum_{i,j,k} (-\frac{1}{2} c_j^{l_j} A_i A_j + \alpha_i^k \alpha_i^{l_i} - \frac{1}{2} \alpha_i^k \alpha_j^k) (\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}) \tilde{\mathbf{S}}_w^{-1} (\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}) \tag{50}$$

which is a quadratic function in terms of $\boldsymbol{\alpha}$ with the linear constraints:

$$\sum_{i=1}^{N} a_i^k = \sum_{i=1}^{N} c_i^k A_i, \ k = 1, \ldots, K. \tag{51}$$

The above optimization problem can be solved using optimization software packages [27] or the MATLAB [34] function `quadprog`. The corresponding decision hyperplane is:

$$f(\mathbf{x}) = \underset{k=1,\ldots,K}{\operatorname{argmax}} (\mathbf{w}_k^T (\phi(\mathbf{x}) - \mathbf{m}^{\Phi}) + b_k) = \underset{k=1,\ldots,K}{\operatorname{argmax}} [\sum_{i=1}^{N} (c_i^k A_{i,o} - \alpha_{i,o}^k) (\phi(\mathbf{x}_i) - \mathbf{m}^{\Phi})^T \mathbf{P} \tilde{\mathbf{S}}_w^{-1} \mathbf{P}^T (\phi(\mathbf{x}) - \mathbf{m}^{\Phi}) + b_k], \tag{52}$$

as detailed in Appendix II.

For the variant (i.e., (46) subject to (48)) the corresponding Lagrangian multiplier is:

$$\begin{aligned} L_4(\boldsymbol{\eta}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \quad &= \sum_{k=1}^{K} \tfrac{1}{2} \boldsymbol{\eta}_k^T \tilde{\mathbf{S}}_w \boldsymbol{\eta}_k + C \sum_{i=1}^{N} \sum_{k=1}^{K} \xi_i^k - \\ &- \sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_i^k [\boldsymbol{\eta}_{l_i}^T (\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}_{l_i}) - \boldsymbol{\eta}_k^T (\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}_k) + b_{l_i} - b_k - 2 + \xi_i^k] - \\ &- \sum_{i=1}^{N} \sum_{k=1}^{K} \beta_i^k \xi_i^k. \end{aligned} \tag{53}$$

and as can be seen in Appendix III. The Wolf dual problem is the maximization of:

$$W(\boldsymbol{\alpha}) = 2 \sum_{i,k} \alpha_i^k + \sum_{k=1}^{K} \sum_{i=1}^{N} \sum_{j=1}^{N} \omega_{i,j,k} \mathbf{x}_i \tilde{\mathbf{S}}_w^{-1} \mathbf{x}_j \tag{54}$$

where $\omega_{i,j,k}$ is defined in Appendix III. The corresponding decision function for the variant is:

$$f(\mathbf{x}) = \underset{k=1,\ldots,K}{\operatorname{argmax}} (\mathbf{w}_k^T (\phi(\mathbf{x}) - \mathbf{m}_k^{\Phi}) + b_k) = \underset{k=1,\ldots,K}{\operatorname{argmax}} [\sum_{i=1}^{N} (c_i^k A_{i,o} - \alpha_{i,o}^k)(\phi(\mathbf{x}_i) - \mathbf{m}_k^{\Phi})^T \mathbf{P} \tilde{\mathbf{S}}_w^{-1} \mathbf{P}^T (\phi(\mathbf{x}) - \mathbf{m}_k^{\Phi}) + b_{k,o}]. \tag{55}$$

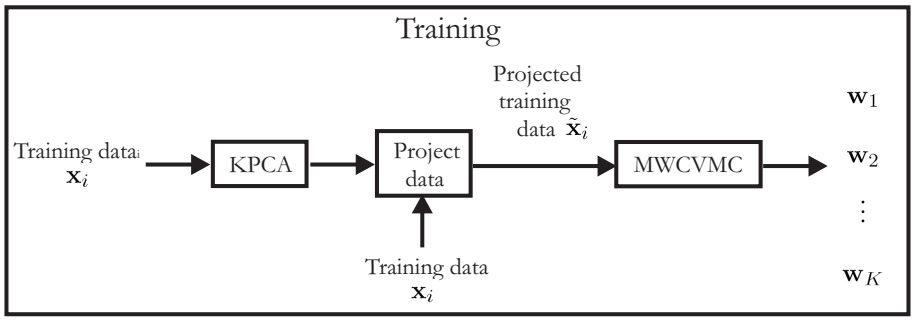


Fig. 2. Diagram of the MWCVMC training procedure.

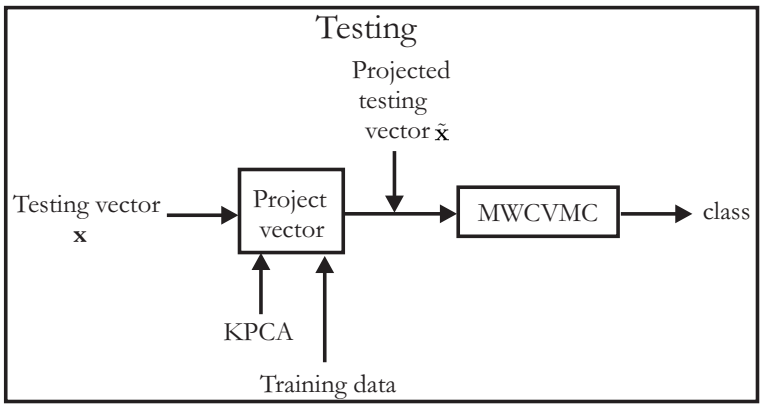


Fig. 3. Diagram of the MWCVMC testing procedure.

Summarizing, in the training phase the samples are first projected using KPCA. Afterwards, the optimization problem (46) subject to (47) (or the variant (46) subject to (48)) is solved. The training

phase is schematically described in Figure 2. When a test sample arrives, it is firstly projected using KPCA and afterwards is classified using (52) or (55). The test step is schematically described in Figure 3.

B. Alternative Multiclass Desicion Surfaces in [5] and [18]

The decision surfaces proposed in [5] and [7] have been inspired by the solution of the linear case where the term $\mathbf{x}_i^T\mathbf{S}_w^{-1}\mathbf{x}_j$ is employed in the dual optimization problem (14). Assuming that the original within-class scatter matrix of the data is not singular, this term has been expressed as an inner product of the form $(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_i)^T(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_j)$ (if $\mathbf{S}_w$ is invertible then it is a positive definite matrix). Then, in [5], instead of projecting $\mathbf{x}_i$ using ϕ (as described previously), the transformed vector $\mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_i$ is projected in the Hilbert space (also using ϕ) and the matrix $[\mathbf{Q}]_{i,j} = \frac{1}{2}y_iy_jh(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_i, \mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_j)$ is used for the solution of the dual optimization problem. Of course, the decision surface provided in [5] does not constitute the solution of the optimization problem of MWCVTCC in Hilbert spaces.

Following this strategy, the nonlinear multi-class decision surfaces proposed in [7] has been formulated. The fact that the term $\mathbf{x}_i^T\mathbf{S}_w^{-1}\mathbf{x}_j$ can be written in terms of dot products as $(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_i)^T(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_j)$ is taken under consideration. Then, kernels are applied in (50) as:

$$W(\boldsymbol{\alpha}) = 2\sum_{i=1}^{N}\sum_{k=1}^{K}\alpha_i^k + \frac{1}{2}\sum_{i,j,k}(-\frac{1}{2}c_j^{l_j}A_iA_j + \alpha_i^k\alpha_i^{l_i} - \frac{1}{2}\alpha_i^k\alpha_j^k)h(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_i, \mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_j). \tag{56}$$

The corresponding decision function is:

$$f(\mathbf{g}) = \underset{k=1,\ldots,K}{\operatorname{argmax}}\frac{1}{2}[\sum_{i=1}^{N}(c_i^kA_i - \alpha_i^n)h(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_i, \mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}) + b_k]. \tag{57}$$

The above decision surfaces are not the ones derived from the generalized MWCVMC optimization problem (30), subject to the constraints (19), which is described in Section III.It has been shown, in [7], that these surfaces outperform maximum margin SVM in facial expression recognition. Moreover, in [5], that the above surfaces outperform maximum margin SVMs in a two class problems for face verification. As we have already mentioned, we have generalized the methods and concepts presented in [5], [7] using arbitrary Mercer's kernel in multiclass problems (the two class problem is a special case of the treated problem).

C. Relationship with Complete Kernel Fisher Discriminant Analysis

In this section, the relationship of the proposed decision hyperplanes/surfaces with the ones derived through CKFD [3] is analyzed. Only the linear case will be considered, in our discussion, since the non-linear case is a direct generalization of the linear one using Mercer's kernels.

As it has been by the Proposition 1 in order to solve the linear or the generalized non-linear constraint optimization problems of MWCVMCs, the problem can be solved in $\Re^{N-1}$ using PCA (KPCA using a linear kernel becomes PCA) and solve an equivalent linear optimization problem there.

In the linear case (i.e., use linear kernel $h(\mathbf{x},\mathbf{y}) = \mathbf{x}^T\mathbf{y}$), in order to move from $\Re^{N-1}$ to $\Re^{N-K}$ we have removed K columns from the matrix $\mathbf{P}$ (the PCA matrix) which are the eigenvector that corresponds to the lowest non-zero eigenvalues of $\mathbf{S}_t$. If these columns are not removed from $\mathbf{P}$, then $\acute{\mathbf{S}}_w = \mathbf{P}^T\mathbf{S}_w\mathbf{P}$ contains K eigenvectors $\boldsymbol{\rho}_k$ that correspond to a null eigenvalues. Let $\boldsymbol{\upsilon}_k \in \Re^M$ be $\boldsymbol{\upsilon}_k = \mathbf{P}\boldsymbol{\rho}_k$, then, under the projection to $\boldsymbol{\upsilon}_k$, all the training samples are separated without an error, since $\boldsymbol{\upsilon}_k^T\mathbf{S}_w\boldsymbol{\upsilon}_k = 0$ and $\boldsymbol{\upsilon}_k^T\mathbf{S}_t\boldsymbol{\upsilon}_k > 0$. That is, $\boldsymbol{\upsilon}_k$ is a solution of the optimization problem (9) and since the data are projected to the one dimensional space it is very easy to find thresholds in order to to perfectly separate the projected vectors. This can be easily proven by observing that all samples after projecting to one of the directions $\boldsymbol{\upsilon}_k$ fall in center of each class [35].

Figure 4 describes pictorially the effect of the vectors $\mathbf{w}$ (K total vectors) for the cases, $\mathbf{w}^T\mathbf{S}_w\mathbf{w} = 0$ and $\mathbf{w}^T\mathbf{S}_t\mathbf{w} > 0$.

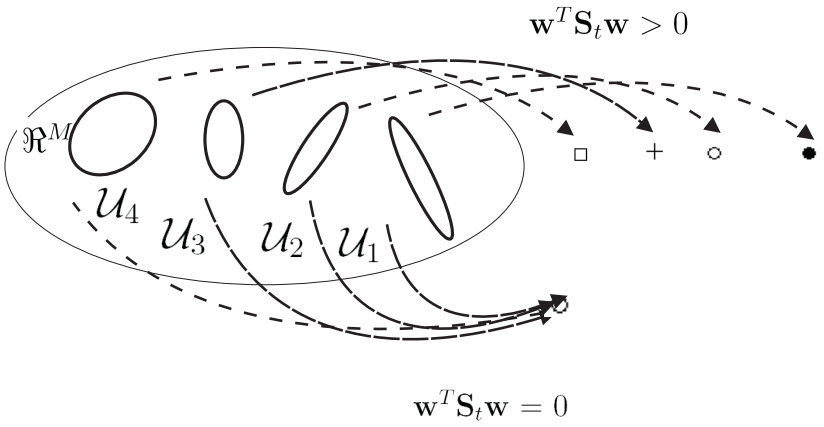


Fig. 4. Illustration of the effect of the projection to a vector $\mathbf{w}$ with $\mathbf{w}^T\mathbf{S}_w\mathbf{w} = 0$. If $\mathbf{w}^T\mathbf{S}_t\mathbf{w} > 0$ is valid for the vector $\mathbf{w}$ then all the training vectors of the different classes are projected to one vector different for each class, while if $\mathbf{w}^T\mathbf{S}_t\mathbf{w} = 0$ all the training vectors are projected to the same point.

It is interesting to notice that the vectors $\boldsymbol{\upsilon}_k$ are the same ones given by the irregular discriminant projection defined in [3], [36]. That is, the vectors $\boldsymbol{\upsilon}_k$ are the solution of the optimization problem:

$$\max_{\mathbf{w}_k \in \Re^M} \text{tr}[\mathbf{W}^T\mathbf{S}_b\mathbf{W}] \, (\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_K] \; ||\mathbf{w}_k|| = 1) \text{ subject to } \mathbf{w}_k^T\mathbf{S}_w\mathbf{w}_k = 0, \tag{58}$$

which is also a maximization point of the Fisher's discriminant ratio:

$$J(\mathbf{W}) = \frac{\text{tr}[\mathbf{W}^T\mathbf{S}_b\mathbf{W}]}{\text{tr}[\mathbf{W}^T\mathbf{S}_w\mathbf{W}]} \tag{59}$$

that makes $J(\mathbf{U}) \rightarrow +\infty$, ($\mathbf{U} = [\boldsymbol{\upsilon}_1, \ldots, \boldsymbol{\upsilon}_K]$). Summarizing, we can tell that we remove the K dimensions of the space $\Re^{N-1}$ due to the fact that the interesting vectors $\mathbf{w}_k$ with $\mathbf{w}_k^T\mathbf{S}_w\mathbf{w}_k = 0$ that provide fully class separability can be found by eigenanalysis only and not by solving a quadratic optimization problem. Hence, in the new space $\Re^{N-K}$ all the solutions $\boldsymbol{\eta}_k$ of the MWCVMC optimization problem satisfy $\boldsymbol{\eta}_k^T\mathbf{S}_w\boldsymbol{\eta}_k > 0$.

IV. Minimum Within-Class Variance Multiclass Classifiers in Pseudo-Euclidean Spaces

In the previous section only conditionally positive kernels have been considered [17]. In this section, the use of not-conditional positive kernels (i.e., indefinite kernels and dissimilarity measure) along with the MWCVMC will be presented. In [15], [37] a unified theory for (dis)similarity measures and kernels has been developed. In terms of kernels, the L_2 similarity measure between the two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ using a function ϕ can be written as:

$$\begin{aligned} d(\mathbf{x}_i, \mathbf{x}_j) &= ||\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)||^2 = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_i) - \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j) - \phi(\mathbf{x}_j)^T\phi(\mathbf{x}_i) + \phi(\mathbf{x}_j)^T\phi(\mathbf{x}_j) \\ &= h(\mathbf{x}_i, \mathbf{x}_i) - 2h(\mathbf{x}_i, \mathbf{x}_j) + h(\mathbf{x}_j, \mathbf{x}_j). \end{aligned} \tag{60}$$

Let us define the similarity (or dissimilarity) matrix $\mathbf{D} \in \Re^{N\times N}$ as:

$$[\mathbf{D}]_{i,j} = d(\mathbf{x}_i, \mathbf{x}_j). \tag{61}$$

The centered matrix $\mathbf{B}$ is defined as:

$$\mathbf{B} = -\frac{1}{2}\mathbf{JDJ} \tag{62}$$

where $\mathbf{J} = \mathbf{I}_{N\times N} - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T \in \Re^{N\times N}$ is the centering matrix, $\mathbf{I}_{N\times N}$ is the $N \times N$ identity matrix and $\mathbf{1}_N$ is the N-dimensional vector of ones. It can be proven that the matrix $\mathbf{B}$ is positive semidefinite, if and only if the kernel h is conditionally positive [37]. Many kernels exist, which have been used very successfully in pattern recognition applications like face recognition [16], [17], [18] that do not necessarily define positive semidefinite matrices $\mathbf{B}$. Typical examples of these kernels are the sigmoid kernels:

$$h(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\kappa(\mathbf{x}_i^T\mathbf{x}_j) + \theta) \tag{63}$$

with $\kappa > 0$ and $\theta < 0$, as well as the fractional polynomial models [16], [18]:

$$h(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T\mathbf{x}_j + 1)^d \tag{64}$$

with $0 < d < 1$. In the following, the MWCVMC using non-conditionally positive kernels will be defined for the general case where only the dissimilarity measure d is known and the explicit form of the kernel function h remains unknown. In the trivial case that the kernel function is known, the dissimilarity can be built using h. In this case, data representation is not strictly performed with vectors but possibly by other means as well (e.g. sets). A dissimilarity measure that can quantify the similarity between object representations $\mathcal{A}_i$[3] and obeys the following properties, should be available:

- reflectivity: $d(\mathcal{A}_i, \mathcal{A}_i) = 0$
- positivity: $d(\mathcal{A}_i, \mathcal{A}_j) > 0$ if $\mathcal{A}_i \neq \mathcal{A}_j$
- symmetry: $d(\mathcal{A}_i, \mathcal{A}_j) = d(\mathcal{A}_j, \mathcal{A}_i)$,

where $d(\mathcal{A}_i, \mathcal{A}_j)$ is a dissimilarity measure between the two object representations $\mathcal{A}_i, \mathcal{A}_j$.

[3]The object $\mathcal{A}_i$ can be a set/vetor but is not necessary to be explicitly defined as its definition is not of particular interest here. The only thing that should be defined is the dissimilarity measure.

A. Embedding function to pseudo-Euclidean spaces

The dissimilarity matrix $\mathbf{D}$ is used to define an embedding function $\mathbf{G} \in \Re^{k \times N}$, where $k \leq N$ is the dimensionality of the embedding. Therefore, the i-th column of $\mathbf{G}$, denoted by $\mathbf{g}_i$, corresponds to the features of the object $\mathcal{A}_i$ in the pseudo-Euclidean space. In order to find the embedding $\mathbf{G}$, the matrix $\mathbf{B}$ is defined as in (62). The matrix $\mathbf{J}$ projects the data so that the embedding $\mathbf{G}$ has zero mean. The eigendecomposition of the matrix $\mathbf{B}$ will give us the desired embedding. The matrix $\mathbf{B}$ is positive semi-definite (i.e., it has real and non-negative eigenvalues), if and only if the distance matrix $\mathbf{D}$ is Euclidean matrix [15]. Therefore, for a non-Euclidean $\mathbf{D}$, $\mathbf{B}$ has negative eigenvalues. For more details on pseudo-Euclidean embedding and dissimilarity based pattern recognition, the interested reader may refer to [15], [23], [38], [20]. Let the matrix $\mathbf{B}$ has p positive and q negative eigenvalues. Then, the matrix $\mathbf{B}$ can be written as:

$$\mathbf{B} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T = \mathbf{Q}|\mathbf{\Lambda}|^{\frac{1}{2}} \begin{bmatrix} \mathbf{M} & \\ & \mathbf{0} \end{bmatrix} |\mathbf{\Lambda}|^{\frac{1}{2}}\mathbf{Q}^T = \mathbf{G}^T\mathbf{M}\mathbf{G}, \tag{65}$$

where $\mathbf{\Lambda}$ is a diagonal matrix with the diagonal consisting of the p positive and q negative eigenvalues, which are presented in the following order: first, positive eigenvalues with decreasing values, then negative ones with decreasing magnitude and finally zero values. The matrix $\mathbf{Q}$ is an orthogonal matrix of the corresponding eigenvectors. The matrix $\mathbf{M}$ is equal to $\begin{bmatrix} \mathbf{I}_{p\times p} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}_{q\times q} \end{bmatrix}$ where $\mathbf{I}_{p\times p}$ and $\mathbf{I}_{q\times q}$ are the identity $p \times p$ and $q \times q$ matrices, and $k = p + q$. The matrix $\mathbf{G}$ is the embedding of the facial image database in the pseudo-Euclidean space $\Re^k = \Re^{(p,q)}$ [15]:

$$\mathbf{G} = |\mathbf{\Lambda}_k|^{\frac{1}{2}}\mathbf{Q}_k^T, \tag{66}$$

where $\mathbf{\Lambda}_k$ contains only the non null diagonal elements of $\mathbf{\Lambda}$. $\mathbf{Q}_k$ is the matrix with the corresponding eigenvectors.

Actually, the pseudo Euclidean-space $\Re^{(p,q)}$ consists of two Euclidean spaces, where the inner product is positive definite for the first one and negative definite for the second one. Using the previous remark, for the sake of completeness, a brief description of the procedure followed, when going back from the embedding $\mathbf{G}$ to the dissimilarity matrix $\mathbf{D}$, will be provided. The inner products in the pseudo-Euclidean space are defined as:

$$\langle \mathbf{g}, \mathbf{y} \rangle = \sum_{i=1}^{p} g_i y_i - \sum_{j=p+1}^{p+q} g_i y_i = \mathbf{g}^T\mathbf{M}\mathbf{y}. \tag{67}$$

The norm of a non-zero vector $\mathbf{g}$ in a pseudo-Euclidean space is defined as:

$$||\mathbf{g}||^2 = \langle \mathbf{g}, \mathbf{g} \rangle = \mathbf{g}^T\mathbf{M}\mathbf{g}, \tag{68}$$

which can be positive, negative or zero (contrary to the positive or zero norm value in an Euclidean space). The dissimilarity matrix $\mathbf{D}$ can now be retrieved from the embedding $\mathbf{G}$, using the notion of the inner products as:

$$\begin{aligned} [\mathbf{D}]_{i,j} &= ||\mathbf{g}_i - \mathbf{g}_j||^2 = \langle \mathbf{g}_i - \mathbf{g}_j, \mathbf{g}_i - \mathbf{g}_j \rangle = (\mathbf{g}_i - \mathbf{g}_j)^T\mathbf{M}(\mathbf{g}_i - \mathbf{g}_j) \\ &= d(\mathcal{A}_i, \mathcal{A}_j) = \mathbf{b}\mathbf{1}^T + \mathbf{1}\mathbf{b}^T - 2\mathbf{B} \end{aligned} \tag{69}$$

where $\mathbf{b}$ is a vector with the diagonal elements of the matrix $\mathbf{B}$.

Prior to proceeding to the description of the MWCVMC in pseudo-Euclidean spaces someone should notice that the matrix $\mathbf{G}$ has uncorrelated features with zero mean vector $\mathbf{m} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{g}_i = \mathbf{0}$. That is, if $\mathbf{S}_t$ is the total scatter matrix, then:

$$\mathbf{S}_t = \sum_i (\mathbf{g}_i - \mathbf{m})(\mathbf{g}_i - \mathbf{m})^T\mathbf{M} = \mathbf{G}\mathbf{G}^T\mathbf{M} = |\mathbf{\Lambda}|\mathbf{M} = \mathbf{\Lambda}. \tag{70}$$

Therefore, $\mathbf{G}$ can be considered to be the result of a mapping of a kernel-PCA (KPCA) projection procedure [32] using indefinite kernels [15], [17]. Thus, if a vectorial object representation is available (i.e., the representation of $\mathcal{A}_i$ is a vector) and $\mathbf{d}$ is defined as in (60) using conditionally positive kernels, then this embedding is the KPCA projection that has been used in section III prior to the optimization of the MWCVMC in Hilbert spaces.

Each object $\mathcal{A}_i$ is supposed to belong to one of the K object classes $\{\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_K\}$. For notation compactness, the set $\mathcal{U}_k$ will be used for referring both to the set of the object representations of the k-th object class and to the various feature vectors that are produced during the embedding and correspond to the objects of the k-th object class. The mean vector for the class r is denoted as $\acute{\mathbf{m}}_r$. Then, the within-class scatter for the vectors $\mathbf{g}_i$ is defined as:

$$\acute{\mathbf{S}}_w = \sum_{r=1}^{K}\sum_{\mathbf{g}_i \in \mathcal{U}_r} (\mathbf{g}_i - \acute{\mathbf{m}}_r)(\mathbf{g}_i - \acute{\mathbf{m}}_r)^T\mathbf{M}. \tag{71}$$

As seen previously, the dimensions that correspond to the null eigenvalues of $\mathbf{B}$ have not been taken into consideration for the definition of the embedding $\mathbf{G}$ and the matrix $\acute{\mathbf{S}}_w$, since they offer no information for the optimization of the MWCVMCs (as described in the previous section). Now we should take care of the dimensions of the embedding that correspond to negative eigenvalues. The problem of these dimensions is that they lead to Hessian matrices that are not positive semidefinite. Hence the optimization problems are not convex and generally NP-complete. Two alternatives exist regarding the dimensions of the embedding $\mathbf{G}$ that correspond to negative eigenvalues:

- to remove the dimensions that correspond to negative eigenvalues. In this case the embedding $\mathbf{G}$ degenerates to:

  $$\mathbf{G}_p = \mathbf{\Lambda}_p^{\frac{1}{2}}\mathbf{Q}_p^T \tag{72}$$

  where $\mathbf{G}_p \in \Re^{p,N}$. This step is preferred when the negative eigenvalues are few in number and very small in magnitude, in comparison to the magnitude of the positive eigenvalues (i.e., the dissimilarity measure is almost Euclidean). Such embedding has been successfully used for face recognition when using KPCA with fractional polynomial kernels [16], [18].
- To use only the magnitude of the negative eigenvalues. This step is preferred when the magnitude of the negative eigenvalues is not small, or when there are many dimensions that correspond to negative eigenvalues in the embedding. In this case the new embedding is:

  $$\mathbf{G}_l = \mathbf{\Delta}_l^{\frac{1}{2}}\mathbf{Q}_l^T \tag{73}$$

  where $\mathbf{\Delta}_l$ is a diagonal matrix having as diagonal elements the magnitude of the diagonal elements of $\mathbf{\Lambda}_l$, in descending magnitude order. The matrix $\mathbf{Q}_l$ contains the corresponding

eigenvectors. For the dimensionality l of the new embedding, it is valid that $l \leq k \leq N$. This step is preferred for the definition of the Hessian matrix of the quadratic optimization problem of SVMs in pseudo-Euclidean spaces [15], [17].

In both cases, the new embedding $\mathbf{G}_l$ is purely Euclidean. Without loss of generality, the embedding $\mathbf{G}_l$ will be considered for the description of the MWCVMC. Let the vector $\mathbf{g}_i^l$ be the i-th column of the matrix $\mathbf{G}_l$. The mean vector for the class r is denoted by $\acute{\mathbf{m}}_\mathbf{r}$ and the mean of all classes by $\acute{\mathbf{m}}$ (which, in the case under examination, is a zero vector). Since there are no dimensions that correspond to negative eigenvalues, the within-class scatter matrix of the embedding $\mathbf{G}_l$ is defined as:

$$\mathbf{S}_w^l = \sum_{r=1}^{K} \sum_{\mathbf{g}_i^l \in \mathcal{U}_r} (\mathbf{g}_i^l - \acute{\mathbf{m}}_r)(\mathbf{g}_i^l - \acute{\mathbf{m}}_r)^T. \tag{74}$$

The dimensionality of the embedding is $l \leq k \leq N$, while the rank of $\mathbf{S}_w^l$ is less than, or equal to $N - K$. Thus, there is not a guarantee that the within-class scatter matrix $\mathbf{S}_w^l$ will be invertible. Two alternatives exist regarding the solution of this problem:

- to avoid initially eigenvectors corresponding to the smallest eigenvalues of $\mathbf{B}$, when defining the pseudo-Euclidean space (i.e., $l \leq N - K$);
- perform eigenanalysis to $\mathbf{S}_w^l$ and remove the null eigenvectors.

Without loss of generality, let us follow the first approach, by choosing $l \leq N - K$. The MWCVMC is defined in the pseudo-Euclidean space as:

$$\min_{\mathbf{w}_k, \mathbf{b}, \boldsymbol{\xi}} \qquad \sum_{k=1}^{K} \frac{1}{2} \mathbf{w}_k^T \mathbf{S}_w^l \mathbf{w}_k + C \sum_{j=1}^{N} \sum_{k \neq l_j} \xi_j^k \tag{75}$$

subject to the constraints:

$$\begin{aligned} \mathbf{w}_{l_j}^T (\mathbf{g}_j^l - \acute{\mathbf{m}}) + b_{l_j} &\geq \mathbf{w}_k^T (\mathbf{g}_j^l - \acute{\mathbf{m}}) + b_k + 2 - \xi_j^k \\ \xi_j^k &\geq 0, \quad j = 1, \ldots, N, \quad k \in \{1, \ldots, K\} \backslash l_j, \end{aligned} \tag{76}$$

for the MWCVMCs the variant is:

$$\begin{aligned} \mathbf{w}_{l_j}^T (\mathbf{g}_j^l - \acute{\mathbf{m}}_{l_j}) + b_{l_j} &\geq \mathbf{w}_k^T (\mathbf{g}_j^l - \acute{\mathbf{m}}_k) + b_k + 2 - \xi_j^k \\ \xi_j^k &\geq 0, \quad j = 1, \ldots, N, \quad k \in \{1, \ldots, K\} \backslash l_j. \end{aligned} \tag{77}$$

The corresponding hyperplanes $(\mathbf{w}_1, b_1), \cdots, (\mathbf{w}_K, b_K)$ are found by solving the optimization problem (75) subject to the constraints (76) as in Appendix II, and for the variant solving (75) subject to (77) as presented in Appendix III.

B. Classifying Novel Object Representations using pseudo-Euclidean embedding and MWCVMC

Let $\{\mathcal{B}_1, \cdots, \mathcal{B}_n\}$ be a set of n objects. The matrix $\mathbf{D}_n \in \Re^{n \times N}$ is created: $[\mathbf{D}_n]_{i,j} = d(\mathcal{B}_i, \mathcal{A}_i)$ which represents the similarity between the n test object and all the training object representations.

The matrix $\mathbf{B}_n \in \Re^{n \times N}$ of inner products relating all new data to all data from the training set can be found as follows:

$$\mathbf{B}_n = -\frac{1}{2}(\mathbf{D}_n\mathbf{J} - \mathbf{U}\mathbf{D}\mathbf{J}), \tag{78}$$

where $\mathbf{J}$ is the centering matrix and $\mathbf{U} = \frac{1}{N}\mathbf{1}_n\mathbf{1}_N^T \in \Re^{n \times N}$. The embedding of the test object representations $\mathbf{G}_n \in \Re^{l \times n}$ that is used for classification is:

$$\mathbf{G}_n = \mathbf{\Delta}_l^{-\frac{1}{2}}\mathbf{Q}_l^T\mathbf{B}_n^T. \tag{79}$$

The columns of the matrix $\mathbf{G}_n$ are the features used for classification. Let $\mathbf{g}_{i,n} \in \Re^l$ be the i-th column of the matrix $\mathbf{G}_n$. For more details about the embedding of novel data in pseudo-Euclidean spaces, the interested reader may refer to [15]. After the embedding, the classification of $\mathcal{B}_i$ to one of the K-object classes is performed by using the decision function:

$$f(\mathcal{B}_i) = \arg\max_{k=1,\ldots,7}(\mathbf{w}_k^T(\mathbf{g}_{i,n} - \acute{\mathbf{m}}) + b_k), \tag{80}$$

or for the variant

$$f(\mathcal{B}_i) = \arg\max_{k=1,\ldots,7}(\mathbf{w}_k^T(\mathbf{g}_{i,n} - \acute{\mathbf{m}}_k) + b_k), \tag{81}$$

where $\mathbf{w}_k$ and b_k have been found during training.

V. EXPERIMENTAL RESULTS

Three set of experiments have been conducted in order to test the proposed methods:

- Multiclass classification experiments using Hausdorff distances for the facial grids in order to recognize the seven basic facial expressions (i.e., test the MWCVMCs in pseudo-Euclidean spaces).
- Multiclass classification experiments using polynomial Mercer's kernels for face recognition (i.e., test the MWCVMCs in Hilbert spaces).
- Multiclass classification experiments with various Mercer kernels using datasets from UCI repository [25].

Moreover, we compare the two MWCVMCs variants presented in Section III (i.e., the one that optimized (30) subject to the constraints (31) and the one that optimizes the same functional subject to (32)). Since, these two MWCVMCs variants minimize the same functional and have about the same separability constraints with a small difference (i.e., in the first we subtract the total mean vector, like a normalization, while in the second we subtract the mean of the class to be classified), we anticipate small performance difference between them.

A. Multiclass Classification Experiments in Face Expression Recognition

1) Database description: The database used for the experiments was created using the Cohn-Kanade database. This database is annotated with FAUs. These combinations of FAUs were translated into facial expressions according to [39], in order to define the corresponding ground truth for the facial expressions. The facial expressions under examination are the six basic ones (anger, disgust,

fear, happiness, sadness and surprise) plus the neutral state. All the available subjects were taken under consideration to form the database for the experiments.

The geometrical information vector taken under consideration is the deformed Candide grid produced by the grid tracking system as described in [40]. In Figure 5, a sample of an image for every facial expression for one poser from this database and the corresponding deformed grid, is shown. The deformed grids were afterwards normalized in order to have the same scale and orientation.

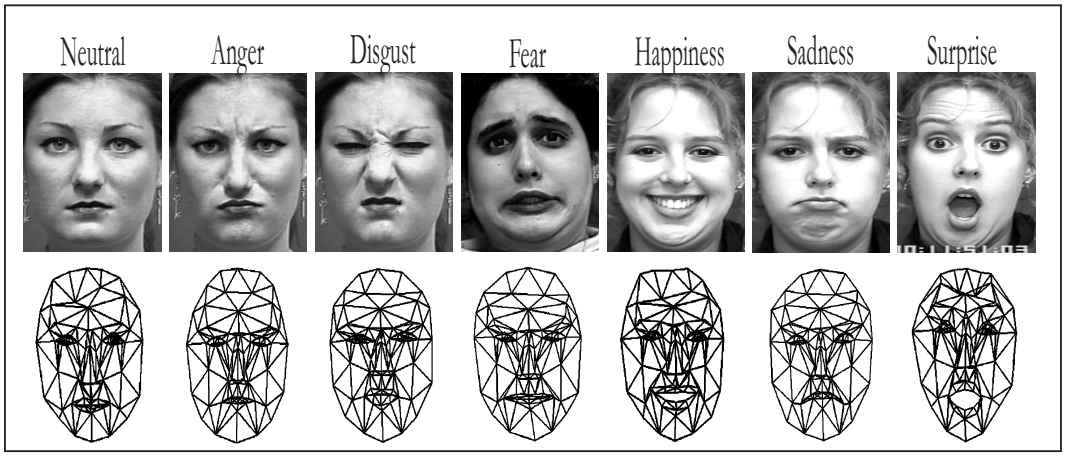


Fig. 5. The facial expression image and the corresponding grid for a poser of the Cohn-Kanade database.

Facial expression recognition was also studied in the presence of partial facial occlusion. A pair of black glasses and a mouth mask, as well as left and right face area masks were created using a graphics computer program, to be superimposed on the eyes or mouth regions respectively to simulate partial occlusion. The glasses were similar to black sun glasses, while the mouth mask was similar to a medical mask that covers the nose, cheeks, mouth and chin. The Candide nodes corresponding to the occluded facial area were discarded. Figure 6 presents one expresser from Cohn-Kanade database posing for the 6 basic facial expressions. On each image, the Candide grid has been superimposed and deformed to correspond to the depicted facial expression, as it is used for the facial expression classification using shape information. The first and last row show the facial part that is taken under consideration when mouth and eyes occlusion is present. The equivalent subset of the Candide grid used for classification is also depicted. In Figure 7 one expresser is depicted from the Cohn-Kanade database for the 6 basic facial expressions under partial occlusion.

2) Hausdorff distance: In order to calculate the distance between two grids, the Hausdorff distance has been used. More specifically, given two finite point sets: $\mathcal{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_p\}$ and $\mathcal{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_p\}$ (in our case this set of points is the set of Candide nodes), the Hausdorff distance is defined as:

$$H(\mathcal{A}, \mathcal{B}) = \max\{d(\mathcal{A}, \mathcal{B}), d(\mathcal{B}, \mathcal{A})\}, \tag{82}$$

where

$$d(\mathcal{A}, \mathcal{B}) = \sup_{\mathbf{a} \in \mathcal{A}} \inf_{\mathbf{b} \in \mathcal{B}} \parallel \mathbf{a} - \mathbf{b} \parallel \tag{83}$$

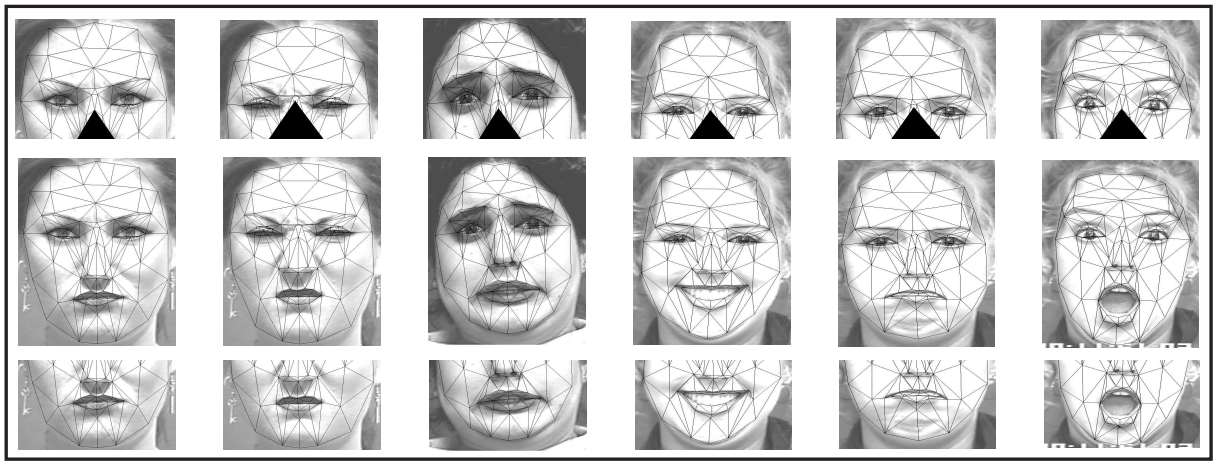


Fig. 6. A poser example from the Cohn-Kanade database, depicting the grid taken under consideration in the original image (second row) and when mouth and eyes occlusion is present (first and last row, respectively).

Fig. 7. A poser example from the Cohn-Kanade database, depicting the original images (second row) and eyes and mouth occlusion (first and last row, respectively).

$\|\cdot\|$ represents some underlying norm defined in the space of the two point sets, which is generally required to be an L_p norm, usually the L_2 or Euclidean norm.

In the proposed method, a robust alternative of the Hausdorff distance, the so-called mean Hausdorff distance [41] is used in order to measure the similarity between facial grids. The mean Hausdorff

distance $d_{MH}(\mathcal{A},\mathcal{B})$ from $\mathcal{A}$ to $\mathcal{B}$ is defined as:

$$d_{MH}(\mathcal{A},\mathcal{B}) = \frac{1}{N(\mathcal{A})}\sum_{\mathbf{a}\in\mathcal{A}}\min_{\mathbf{b}\in\mathcal{B}}\|\mathbf{a}-\mathbf{b}\| \tag{84}$$

where $N(\mathcal{A})$ is the number of points in $\mathcal{A}$. The mean Hausdorff distance is used to create a feature space, using pseudo-Euclidean embedding, as described in Section IV, so as to define later a multiclass SVM classifier in this space. It should be noted here that in the setup used in this paper, where the same grid (the Candide grid) is tracked in all cases over facial video frames, the correspondences between the grid nodes $\mathbf{a}_i, \mathbf{b}_i, i = 1,\ldots,p$ $(p = q)$ in the two grid sets are known. Thus, the sum of Euclideans $\sum_i \|\mathbf{a}_i - \mathbf{b}_i\|$ would suffice. However, the use of Hausdorff distance makes the proposed system applicable to other scenarios, e.g. when different grids are used or when part of the grid is not available ($p \neq q$ e.g. due to image cropping). This may occur when a tracking algorithm is applied and some nodes are lost or considered unreliable. Thus, the general Hausdorff distance is adopted. Another measure that we are currently investigating is the angle of the Candide points between the neutral and the expressed grids. Using the angle of points in a sequence of grids the dynamics of facial expression could be described. But this approach has the same disadvantage as the one proposed in [7], in which deformation vectors have been used for facial expression recognition, and require the initial detection of the neutral state (the neutral state is not required in the proposed procedure).

3) Experimental protocol: The most frequently used approach for testing the generalization performance of a classifier is the leave-one cross-validation approach [42]. It was devised in order to make maximal use of the available data and produce averaged classification accuracy results. The term leave-one out cross-validation does not correspond to the classical leave-one-out definition, as a variant of leave-one-out was used (i.e., leave 20% of the samples out) for the formation of the test dataset in our experiments. However, the procedure followed will be called leave-one-out from now on for notation simplicity without loss of generalization. More specifically, all image sequences contained in the database are divided into 7 facial expression classes. Five sets containing 20% of the data for each class, chosen randomly, were created. One set containing 20% of the samples for each class is used as the test set, while the remaining sets form the training set. After the classification procedure is performed, the samples forming the test set are incorporated into the current training set, and a new set of samples (20% of the samples for each class) is extracted to form the new test set. The remaining samples create the new training set. This procedure is repeated five times. A diagram of the leave-one-out cross-validation method can be seen in Figure 8. The average classification accuracy is defined as the mean value of the percentages of the correctly classified facial expressions over all data presentations. The accuracy achieved for each facial expression is averaged over all

facial expressions and does not provide any information with respect to a particular expression. The confusion matrices [7] have been computed to handle this problem. The confusion matrix is a $n \times n$ matrix containing information about the actual class label lab_{ac} (in its columns) and the label obtained through classification lab_{cl} (in its rows). The diagonal entries of the confusion matrix are

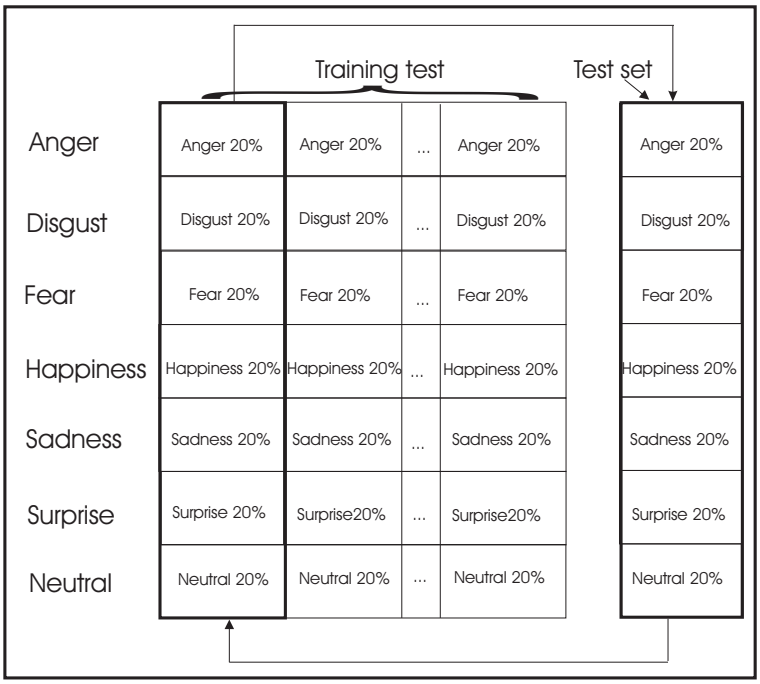


Fig. 8. Diagram of leave-one-out method used in classification assessment for facial expression and FAUs recognition.

the percentages that correspond to the cases when facial expressions are correctly classified, while the off-diagonal entries correspond to misclassifications. The abbreviations an, di, fe, ha, sa, su and ne represent anger, disgust, fear, happiness, sadness, surprise and neutral, respectively. We have experimented with various numbers of the C parameter (from $C = 10^{-6}$ until $C = 10^{6}$ in log scale) and the best setup has been when using $C = 100$ for all tested classifiers. Only the best accuracies achieved for any method used are taken under consideration to make the final conclusions.

4) Experiments regarding the entire Candide grid: The confusion matrix obtained when maximum margin SVMs were used taking under consideration the deformed Candide grids, is presented in Table Ia. The accuracy achieved was equal to 85.2%. As can be seen from the confusion matrix, fear seems to be the most ambiguous facial expression having the lowest correct classification ration (71.2%). The overall facial expression recognition accuracy rates achieved for different number of dimensions of the pseudo-Euclidean space of the Hausdorff distances taken under consideration when maxim margin SVMs, MWCVMCs and MWCVMCs variant were used are depicted in Figure 9a. The highest overall accuracy rate achieved was equal to 99% (achieved by MWCVMC and the variant). The confusion matrix calculated in this case is presented in Table Ib. As can be seen from the confusion matrix, almost all previous misclassifications are now eliminated. The only misclassification remaining is the one between fear and happiness, which was actually the most usual misclassification appearing when the maximum margin SVMs were used.

A comparison of the recognition rates achieved for each facial expression with the state of the

TABLE I

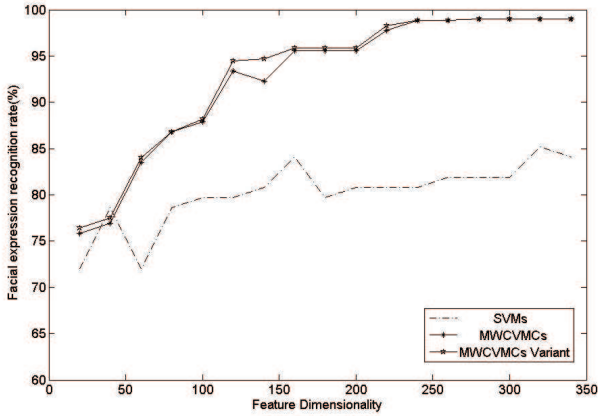
CONFUSION MATRICES WHEN USING A) MAXIMUM MARGIN SVMS AND B) MWCVMCS.

$lab_{ac}\% \backslash lab_{cl}\%$	an	di	fe	ha	sa	su	ne
an	91	14.3	0	0	10.8	0	4.8
di	6	85.7	7.3	0	0	0	0
fe	0	0	71.2	0	0	7.1	2.4
ha	0	0	8.8	91	4.6	0	0
sa	0	0	5.5	0	80	0	2.4
su	3	0	0	0	0	92.9	5.8
ne	0	0	7.2	9	4.6	0	84.6

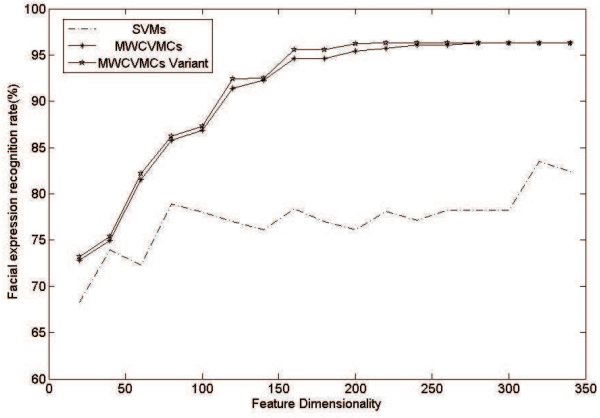
(a)

$lab_{ac}\% \backslash lab_{cl}\%$	an	di	fe	ha	sa	su	ne
an	100	0	0	0	0	0	0
di	0	100	0	0	0	0	0
fe	0	0	93	0	0	0	0
ha	0	0	7	100	0	0	0
sa	0	0	0	0	100	0	0
su	0	0	0	0	0	100	0
ne	0	0	0	0	0	0	100

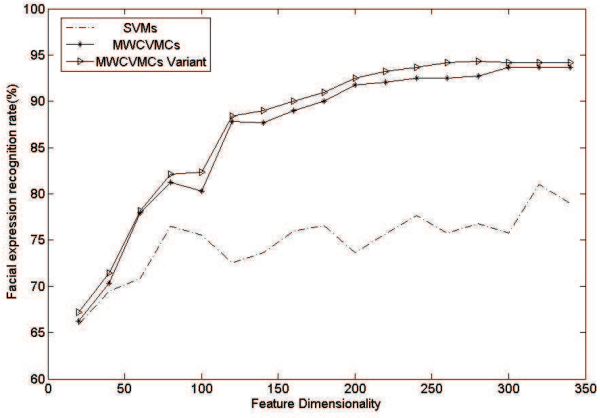
(b)



(a)



(b)



(c)

Fig. 9. Recognition accuracies obtained for facial expression recognition using maximum margin SVMs and MWCVMC in the pseudo-Euclidean space when a) all the grid nodes were used b) eyes occlusion is present (mouth nodes discarded) and c) mouth occlusion is present (eyes nodes discarded).

art [42]-[45], when six facial expression were examined (the neutral state was not taken under consideration) is depicted in Figure 10, where the recognition rate of each of the six basic facial expressions is depicted. As can be seen, our recognition rates are the highest for each facial expression.

The second best reported results are the ones in [45], where a 97% total recognition rate has been reported. Moreover, the proposed method has been tested for the recognition of the neutral state, unlike the methods in [43], [44], [45], that have been tested only for the recognition of the six expression. That is, the error that will be introduced by the inclusion of the neutral state to the other expressions remains unknown. The method in [42] has been tested for the recognition of neutral state and has achieved 78.59% (our method had 100% performance for the neutral state). To the best of the authors knowledge these are the best results achieved in Cohn-Kanade database for the recognition of the seven facial expressions.

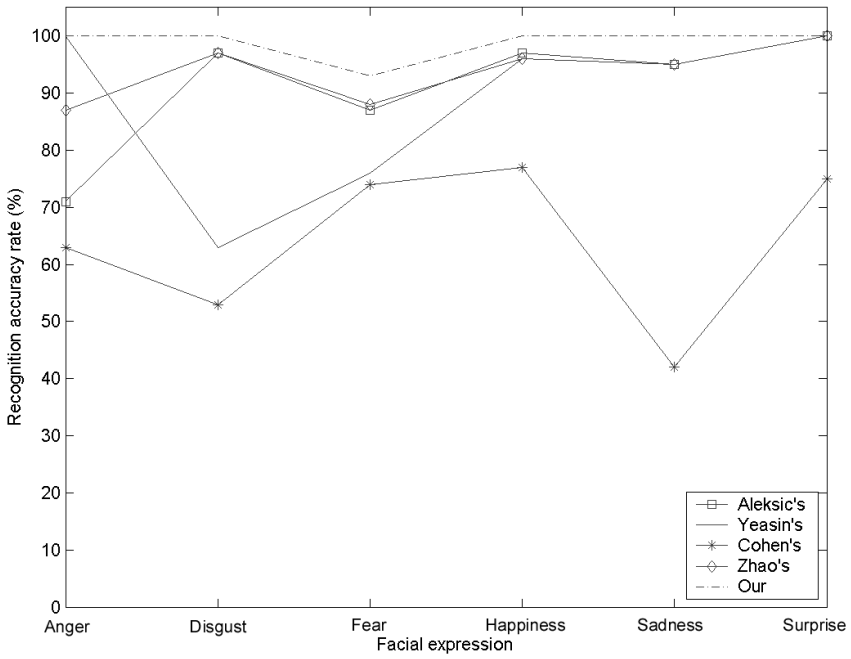


Fig. 10. Comparison of the recognition rate for every of the six basic facial expression of various state-of-the-art facial expression recognition methods.

5) Experiments in the presence of eyes occlusion: The recognition accuracy rate achieved when eyes occlusion was present and the maximum margin SVMs were used was equal to 83.5%. Thus, the introduction of eyes occlusion results in a 1.7% recognition accuracy rate drop. The equivalent recognition accuracy rate achieved when MWCVMC (or MWCVMC variant) were used was equal to 96.3% (2.7% drop in recognition accuracy due to eyes occlusion). The recognition accuracy rates achieved for different number of dimensions of the pseudo-Euclidean space of the Hausdorff distances taken under consideration when maximum margin SVMs and the two MWCVMC were used are depicted in Figure 9b.

6) Experiments in the presence of mouth occlusion: The recognition accuracy rate achieved when mouth occlusion was present and the maximum margin SVMs were used was equal to 79.8%. Thus, eyes occlusion results in a 5.4% recognition accuracy rate drop. The equivalent recognition accuracy rate achieved when MWCVMC (or MWCVMC variant) were used was equal to 93.7% (5.3% accuracy drop due to eyes occlusion presence). The recognition accuracy rates achieved for different number of dimensions of the pseudo-Euclidean space of the Hausdorff distances taken under consideration when maximum margin SVMs and MWCVMC were used are depicted in Figure 9c.

B. Multiclass Classification Experiments in Face Recognition

The face recognition problem has been performed in order to assess the proposed method using Mercer's kernels. Experiments were performed using the ORL (Olivetti Research Laboratory) database. This database includes ten different images of 40 distinct subjects. For some of them, the images were taken at different times and there are variations in facial expression (open/closed eyes, smiling/nonsmiling) and facial details (glasses/no glasses). The original face images were all sized 92×112 pixels. The gray scale was linearly normalized to lie within the range $[-1, 1]$. The experiments were performed with five training images and five test images per person for a total of 200 training images and 200 test images. There was no overlap between the training and test sets. Since the recognition performance is affected by the selection of the training images, the reported results were obtained by training 5 non-overlapping repetitions with different training examples (random selection of five images from ten ones per subject, out of a total of selections) and selecting the average error over all the results. In Figure 11 the mean error rates for the proposed approach and the maximum margin SVM are depicted. The tested kernels have been the polynomial kernels with degrees from 1 to 4. The best error rate of the proposed method has been measured at about 1.5% for the proposed methods (both MWCVMC variants gave the same mean recognition rate in this experiment) was an average of 5 simulations. However, individual experiments had given error rates as low as 0%. The SVM classifier in this problem achieved a best error rate at about 3%.

For completeness, we should note here that the proposed MWCVMCs classifiers are similar to the classifiers tested for face recognition in the ORL database using a KPCA plus SVM scheme. That is, the method for finding the MWCVMCS classifier is comprised of an initial KPCA step, and afterwards a minimum within class variance multiclass system is trained. The method of the KPCA plus SVM classifier in [46] has shown superior results in face recognition in comparison to the other tested methods. Actually, the successful application of a KPCA plus SVM scheme has motivated the application of MWCVMCs for face recognition in ORL database. We have experimented with a KPCA plus SVM approach as in [46] and the best mean recognition rate has been 2.5%. As can be seen our method outperforms KPCA plus SVMs in ORL database.

C. Experimental Results in Other Databases

Apart from facial expression and face recognition we have applied the proposed classifier to other problems. To do so we have used benchmark data sets from the University of California at Irvine

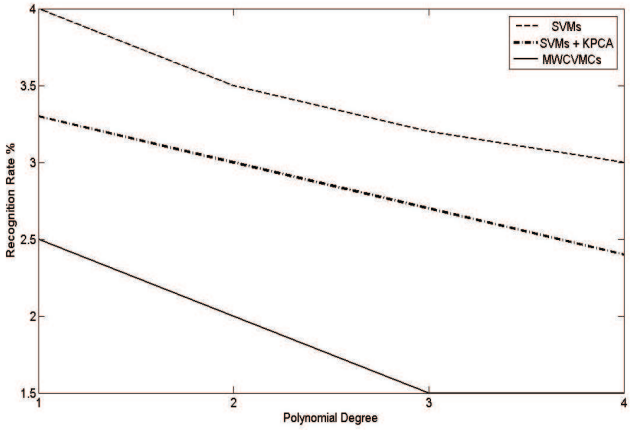


Fig. 11. Mean face recognition error rates in ORL database.

(UCI) Repository database [25]. More precisely, we have used the Balance-scale, Glass, Iris and Wine database. We have used a similar testing protocol as the one used in facial expression recognition experiments but in this time we have considered 70% for training and the remaining 30% for testing. This procedure has been repeated five times. The average classification accuracy is defined as the mean value of the percentages of the correctly classified samples over all data presentations. We have tested various kernels (i.e., polynomial and RBF kernels) but we will report only the best results for all the tested kernels and for all the tested approaches. The C values that we have tested had been from $C = 10^{-6}$ to $C = 10^{6}$ in log scale. For case of RBF kernels in order to choose the parameter γ (spread) we have used a simple euristic method. That is, on the training set, we calculate the average of the distance from each instance to its nearest neighbor and call this γ_0 .We used in the experiments $\gamma = \{\gamma_0, 2\gamma_0, 4\gamma_0\}$.

The balance-scale is separated into three classes with a total of 625 four dimensional vectors. For this dataset the linear kernel (i.e., $k(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^T\mathbf{x}_2$) has given the best results that have been 87.7% for typical SVMs, 92.9% for the MWCVMCs and 93.5% for the second variant of MWCVMCs (in this case the within class scatter matrix was invertible). The second dataset has been the Glass dataset that is separated into 6 classes giving a total of 214 9-dimensional vectors. For this dataset the best kernel has been an RBF kernel with variance $\gamma = \gamma_0$ for SVMs and an RBF with variance $\gamma = 2\gamma_0$ for the MWCVMCs. The best mean error rate for SVMs has been 58.4%, for MWCVMCs and for the second variant has been 63% and 64%, respectively. The third dataset has been Iris which is separated into 3 classes of a total of 150 four dimensional vectors. The best kernel for this dataset has been an RBF with variance $\gamma = 2\gamma_0$ for all the tested classifiers. The best results have been 96.07% for SVMs and 96.73% for both MWCVMCs and for the second variant. The final dataset has been the Wine dataset which is separated into 3 classes containing a total of 178 13-dimensional vectors. The RBF kernel has given the best results for all the tested classifiers, with $\gamma = 2\gamma_0$. In this dataset SVMs

have given 93.3%, the MWCVMCs have achieved 96.67% and the variant of MWCVMCs 97.1%.

The best results are summarized in Table II. As can be seen the proposed classifiers outperform maximum margin classifiers in all cases.

TABLE II

MEAN ERROR RATES A) BALANCE-SCALE B) GLASS C) IRIS AND D) WINE.

Method	Kernel	Mean Error Rate %
SVMs	linear	87.7
MWCVMCs	linear	92.9
MWCVMCs Variant	linear	**93.5**

(a)Balance-Scale

Method	Kernel	Mean Error Rate %
SVMs	RBF	58.4
MWCVMCs	RBF	63.01
MWCVMCs Variant	RBF	**64**

(b) Glass

Method	Kernel	Mean Error Rate%
SVMs	RBF	96
MWCVMCs	RBF	**96.7**
MWCVMCs Variant	RBF	**96.7**

(c) Iris

Method	Kernel	Mean Error Rate%
SVMs	RBF	93.3
MWCVMCs	RBF	96.67
MWCVMCs Variant	RBF	**97.1**

(d) Wine

VI. CONCLUSIONS

In this paper novel multiclass decision hyperplanes/surfaces have been proposed based on the minimization of within-class variance in Hilbert spaces subject to separability constraints. We have provided robust solutions for the optimization problem. We have related the proposed classifiers with SVMs and we have provided insights why the proposed surfaces can outperform maximum margin classifiers. Moreover, we have tried to related the proposed classifiers with Fisher Kernel Discriminant Analysis. We have extended the proposed classifiers in pseudo-Euclidean spaces (i.e., defining the proposed classifiers with indefinite kernels). We have shown the usefulness of this extension by applying the proposed classifiers in a space defined by Hausdorff distances and we have applied the method for the classification of seven facial expressions, where state-of-art facial expression recognition rates have been achieved. We have applied the proposed classifiers to other classification problems where it is shown that they outperform typical maximum margin classifiers. Further, research on the topic includes the explicitly measurement of the VC dimension of the proposed classifiers and find surfaces with VC dimension strictly less than the one of maximum margin classifiers. Another subject for research on the topic is the robust calculation of the enclosing hyperellipse of every of the classes. This can be achieved by the robust calculation of the covariance and the mean of each of the classes. Moreover, the proposed classifiers can be applied in a straightforward manner to other multiclass SVM approaches apart the one described in this paper [29], [47]. Furthermore, it would be an interesting topic to make the training procedure of the classifiers an online one. This requires the use of both iterative KPCA and SVM algorithms. Thus, another possible research topic would be the combination of algorithms such as [48], [49] for iterative KPCA and such as [50] for online SVM training in order to make online minimum within class variance classifiers. Finally, it would

be a very interesting topic to compare the proposed classifiers to recently introduced SVM variants that consider class statistics, as well [30], [51], [52].

APPENDIX I
PROOF OF PROPOSITION

Proposition 2. If for some $\boldsymbol{\zeta} \in \mathcal{H}$, $\boldsymbol{\zeta}^T \mathbf{S}_t^\Phi \boldsymbol{\zeta} = 0$, then under the projection $\boldsymbol{\zeta}$ for all training vectors $\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)$ with $\phi(\mathbf{x}_i) \neq \phi(\mathbf{x}_j)$, the following holds $\boldsymbol{\zeta}^T \phi(\mathbf{x}_i) = \boldsymbol{\zeta}^T \phi(\mathbf{x}_j)$. In other words, under the projection $\boldsymbol{\zeta}$ all the training vectors $\phi(\mathbf{x}_i)$ fall in the same point. Thus, $r = \boldsymbol{\zeta}^T \phi(\mathbf{x}_i)$ is a constant $\forall\ \mathbf{x}_i \in \mathcal{U}$.

Let the matrix $\mathbf{X}^\Phi = [\phi(\mathbf{x}_1) \dots \phi(\mathbf{x}_N)]$ that has as columns the projected training vectors. The total scatter matrix $\mathbf{S}_t^\Phi$ can be written as:

$$\mathbf{S}_t^\Phi = \sum_{i=1}^{N} (\phi(\mathbf{x}_i) - \mathbf{m}^\Phi)(\phi(\mathbf{x}_i) - \mathbf{m}^\Phi)^T = (\mathbf{X}^\Phi - \mathbf{X}^\Phi \mathbf{G}_N)(\mathbf{X}^\Phi - \mathbf{X}^\Phi \mathbf{G}_N)^T \tag{85}$$

where $\mathbf{G}_N$ is a matrix with elements equal to N^{-1}. Let $\mathbf{I}_N$ be the identity $N \times N$ matrix. The following holds:

$$\begin{aligned} &\boldsymbol{\zeta}^T \mathbf{S}_t^\Phi \boldsymbol{\zeta} = 0 \Leftrightarrow \boldsymbol{\zeta}^T (\mathbf{X}^\Phi - \mathbf{X}^\Phi \mathbf{G}_N)(\mathbf{X}^\Phi - \mathbf{X}^\Phi \mathbf{G}_N)^T \boldsymbol{\zeta} = 0 \\ &||(\mathbf{I}_N - \mathbf{G}_N) {\mathbf{X}^\Phi}^T \boldsymbol{\zeta}||^2 = 0 \Leftrightarrow \boldsymbol{\zeta}^T \phi(\mathbf{x}_i) = \boldsymbol{\zeta}^T \phi(\mathbf{x}_j) = \boldsymbol{\zeta}^T \mathbf{m}^\Phi \ \blacksquare \end{aligned} \tag{86}$$

Let $\mathcal{B}^\Phi$ and $\mathcal{B}_\perp^\Phi$ be the complementary spaces spanned by the orthonormal eigenvectors of $\mathbf{S}_t^\Phi$ that correspond to non-zero eigenvalues and to zero eigenvalues, respectively. Let $\boldsymbol{\varphi} \in \mathcal{B}^\Phi$ and $\boldsymbol{\zeta} \in \mathcal{B}_\perp^\Phi$. Thus, $\mathbf{w}^T \mathbf{S}_w^\Phi \mathbf{w} = \boldsymbol{\varphi}^T \mathbf{S}_w^\Phi \boldsymbol{\varphi}\ \forall\ \mathbf{w} \in \mathcal{H}$. A proof of the above proposition can be found in [3]. The normal vector $\mathbf{w}_k$ of the decision surface can be written as $\mathbf{w}_k = \boldsymbol{\varphi}_k + \boldsymbol{\zeta}_k$ with $\boldsymbol{\varphi}_k \in \mathcal{B}^\Phi$ and $\boldsymbol{\zeta}_k \in \mathcal{B}_\perp^\Phi$.

Taking under consideration that $\mathbf{w}_k = \boldsymbol{\varphi}_k + \boldsymbol{\zeta}_k$, the Lagrangian of the optimization problem (30) subject to the separability constraints (19) can be written as:

$$\begin{aligned} L_1(\mathbf{w}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \textstyle\sum_{k=1}^{K} \mathbf{w}_k^T \mathbf{S}_w^\Phi \mathbf{w}_k + C \sum_{i=1}^{N} \sum_{k=1}^{K} \xi_i^k - \\ &- \textstyle\sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_i^k [(\mathbf{w}_{l_i} - \mathbf{w}_k)^T (\phi(\mathbf{x}_i) - \mathbf{m}^\Phi) + b_{l_i} - b_k - 2 + \xi_i^k] - \\ &- \textstyle\sum_{i=1}^{N} \sum_{k=1}^{K} \beta_i^k \xi_i^k \\ &= \textstyle\sum_{k=1}^{K} \boldsymbol{\varphi}_k^T \mathbf{S}_w^\Phi \boldsymbol{\varphi}_k + C \sum_{i=1}^{N} \sum_{k=1}^{K} \xi_i^k - \\ &- \textstyle\sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_i^k [(\boldsymbol{\varphi}_{l_i} + \boldsymbol{\zeta}_{l_i} - \boldsymbol{\varphi}_k - \boldsymbol{\zeta}_k)^T (\phi(\mathbf{x}_i) - \mathbf{m}^\Phi) + b_{l_i} - b_k - 2 + \xi_i^k] - \\ &- \textstyle\sum_{i=1}^{N} \sum_{k=1}^{K} \beta_i^k \xi_i^k. \end{aligned} \tag{87}$$

Taking under consideration the Proposition 2, since for $\boldsymbol{\zeta}_{k,o} \in \mathcal{B}_\perp$, $\boldsymbol{\zeta}_{k,o}^T \mathbf{S}_t^\Phi \boldsymbol{\zeta}_{k,o} = 0$, then $\boldsymbol{\zeta}_{k,o}^T \phi(\mathbf{x}_i)$ is a constant for all $\phi(\mathbf{x}_i)$. That is, $\boldsymbol{\zeta}_{l_i} \phi(\mathbf{x}_i) = \boldsymbol{\zeta}_{l_i} \mathbf{m}^\Phi$ and $\boldsymbol{\zeta}_k \phi(\mathbf{x}_i) = \boldsymbol{\zeta}_k \mathbf{m}^\Phi$. Thus, L_1 becomes:

$$\begin{aligned} L_1(\mathbf{w}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \textstyle\sum_{k=1}^{K} \boldsymbol{\varphi}_k^T \mathbf{S}_w^\Phi \boldsymbol{\varphi}_k + C \sum_{i=1}^{N} \sum_{k=1}^{K} \xi_i^k - \\ &- \textstyle\sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_i^k [(\boldsymbol{\varphi}_{l_i} - \boldsymbol{\varphi}_k)^T (\phi(\mathbf{x}_i) - \mathbf{m}^\Phi) + b_{l_i} - b_k - 2 + \xi_i^k] - \\ &- \textstyle\sum_{i=1}^{N} \sum_{k=1}^{K} \beta_i^k \xi_i^k. \end{aligned} \tag{88}$$

The optimum hyperplane $\mathbf{w}_o$ can be written as $\mathbf{w}_{k,o} = \boldsymbol{\varphi}_{k,o} + \boldsymbol{\zeta}_{k,o}$. Then:

$$\nabla_{\mathbf{w}_k} L_1|_{\mathbf{w}_k=\mathbf{w}_{k,o}} = \nabla_{\boldsymbol{\varphi}_k} L_1|_{\boldsymbol{\varphi}_k=\boldsymbol{\varphi}_{k,o}} = \mathbf{0} \Leftrightarrow \mathbf{S}_w^{\Phi}\boldsymbol{\varphi}_{k,o} - \sum_{i=1}^{N}(c_i^k A_{i,o} - a_{i,o}^k)(\phi(\mathbf{x}_i) - \mathbf{m}^{\Phi}) = \mathbf{0}. \quad (89)$$

It can be shown in a straightforward way that the gradient in (89) is the same as the gradient of the optimization problem (42) subject to the constraints (43). Hence, the separability constraints (31) can be safely replaced by the separability constraints (43). Thus, the part $\boldsymbol{\zeta}_{k,o}$ of the vector $\mathbf{w}_{k,o}$ does not play any role in the separability constraints (since an arbitrary vector $\boldsymbol{\zeta}_{k,o}$ can be chosen, the vector $\boldsymbol{\zeta}_{k,o} = \mathbf{0}$ is selected) and the Proposition 1 has been proven. A similar approach can be used for proving the equivalent proposition for the MWCVMCs variant.

APPENDIX II
WOLF DUAL PROBLEM FOR THE OPTIMIZATION OF LAGRANGIAN (49)

In order to find the optimum separating hyperplanes for the optimization problem (46) subject to the constraints (47), we have to define the saddle point of the Langragian (49). At the saddle point, the solution should satisfy the KKT conditions, for $k = 1, \ldots, K$:

$$\nabla_{\boldsymbol{\eta}_k} L_3|_{\boldsymbol{\eta}_k=\boldsymbol{\eta}_{k,o}} = 0 \Leftrightarrow \boldsymbol{\eta}_{k,o} = \tilde{\mathbf{S}}_w^{-1}\sum_{i=1}^{N}(c_i^k A_{i,o} - a_{i,o}^k)(\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}) \quad (90)$$

$$\frac{\partial L_3}{\partial b_k}|_{b_k=b_{k,o}} = 0 \Leftrightarrow \sum_{i=1}^{N}\alpha_{i,o}^k = \sum_{i=1}^{N} c_i^k A_{i,o} \quad (91)$$

$$\frac{\partial L_3}{\partial \xi_k}|_{\xi_k=\xi_{k,o}} = 0 \Leftrightarrow \beta_{j,o}^k + \alpha_{j,o}^k = C \quad \text{and} \quad 0 \leq \alpha_{j,o}^k \leq C. \quad (92)$$

Substituting (90) back into (49) we obtain:

$$\begin{aligned} L_3(\boldsymbol{\eta}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \quad &= \textstyle\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(c_i^k A_i - \alpha_i^k)(c_j^k A_j - \alpha_j^k)((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}})^T\mathbf{S}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}})) - \\ &- \textstyle\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_i^k[\sum_{j=1}^{N}(c_j^{l_i} A_j - \alpha_j^{l_i})((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}})^T\tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}})) - \\ &- \textstyle\sum_{j=1}^{N}(c_j^k A_j - \alpha_j^k)((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}})^T\tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}})) + b_{l_i} - b_k - 2] - \\ &- \textstyle\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_i^k\xi_i^k + C\sum_{k=1}^{K}\sum_{i=1}^{N}\xi_i^k - \sum_{i=1}^{N}\sum_{k=1}^{K}\beta_i^k\xi_i^m. \end{aligned} \quad (93)$$

Adding the constraint (92), the terms in $\boldsymbol{\xi}$ disappear. Only the two terms in $\boldsymbol{\beta}$ are considered:

$$B_1 = \sum_{i,k}\alpha_i^k b_{l_i} = \sum_k b_k(\sum_i c_i^k A_i) \text{ and } B_2 = -\sum_{i,k}\alpha_i^k b_k = -\sum_k b_k(\sum_i \alpha_i^k). \quad (94)$$

But, from (91)

$$\sum_{i=1}^{N}\alpha_i^k = \sum_{i=1}^{N} c_i^k A_i \quad (95)$$

so $B_1 = B_2$ and the two terms cancel each other, giving:

$$\begin{aligned} L_3(\boldsymbol{\eta}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = W(\boldsymbol{\alpha}) = \quad &2\textstyle\sum_{i,k}\alpha_i^k + \frac{1}{2}\sum_{i,j,k}(\frac{1}{2}c_i^k c_j^k A_i A_j - \frac{1}{2}c_i^k A_i\alpha_j^k - \frac{1}{2}c_j^k A_i\alpha_i^k + \frac{1}{2}\alpha_i^k\alpha_j^k - \\ &- c_j^{l_i} A_j\alpha_i^k + \alpha_i^k\alpha_j^{l_i} + c_j^k A_j\alpha_i^k - \alpha_i^k\alpha_j^k)((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}})^T\tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}})) \end{aligned} \quad (96)$$

Since $\sum_k c_i^k A_i \alpha_j^k = \sum_k c_j^k A_j \alpha_i^k$ we have:

$$W(\boldsymbol{\alpha}) = 2\sum_{i,k}\alpha_i^k + \frac{1}{2}\sum_{i,j,k}(\frac{1}{2}c_i^k c_j^k A_i A_j - c_j^{l_i} A_i A_j + \alpha_i^k \alpha_j^{l_i} - \frac{1}{2}\alpha_i^k \alpha_j^k)((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}})^T \tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}))$$

but $\sum_k c_i^k c_j^k = c_i^{l_i} = c_j^{l_j}$ so:

$$W(\boldsymbol{\alpha}) = 2\sum_{i,k}\alpha_i^k + \frac{1}{2}\sum_{i,j,k}[-\frac{1}{2}c_j^{y_i} A_i A_j + \alpha_i^k \alpha_j^{y_i} - \frac{1}{2}\alpha_i^k \alpha_j^k]((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}})^T \tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}})) \tag{97}$$

which is a quadratic function in terms of alpha with linear constraints:

$$\textstyle\sum_{i=1}^N \alpha_i^k = \sum_{i=1}^N c_i^k A_i, \quad k = 1, \ldots, K \text{ and} \tag{98}$$

$$0 \leq \alpha_i^k \leq C,\ i = 1, \ldots, N\ \ a_i^{l_i} = 0,\ \ k \in \{1, \ldots, K\} \backslash l_i. \tag{99}$$

The combination of (90) with the fact that $\boldsymbol{\eta}_{k,o} = \mathbf{P}^T \boldsymbol{\phi}_{k,o}$ (from the isomorphic mapping (45)) and the results of the Proposition 1, provides the following decision function:

$$\begin{aligned} f(\mathbf{x}) &= \text{argmax}_{k=1,\ldots,K}[\mathbf{w}_{k,o}^T(\phi(\mathbf{x}) - \mathbf{m}^\Phi) + b_{k,o}] = \text{argmax}_{k=1,\ldots,K}[\boldsymbol{\phi}_{k,o}^T(\phi(\mathbf{x}) - \mathbf{m}^\Phi) + b_{k,o}] \\ &= \text{argmax}_{k=1,\ldots,K}[\textstyle\sum_{i=1}^N (c_i^k A_{i,o} - \alpha_{i,o}^k)(\phi(\mathbf{x}_i) - \mathbf{m}^\Phi)^T \mathbf{P}\tilde{\mathbf{S}}_w^{-1}\mathbf{P}^T(\phi(\mathbf{x}) - \mathbf{m}^\Phi) + b_{k,o}] \end{aligned} \tag{100}$$

or equivalently:

$$\begin{aligned} f(\mathbf{x}) = \ & \text{argmax}_{k=1,\ldots,K}[\textstyle\sum_{i:y_i=k} A_{i,o}(\phi(\mathbf{x}_i) - \mathbf{m}^\Phi)^T \mathbf{P}\tilde{\mathbf{S}}_w^{-1}\mathbf{P}^T(\phi(\mathbf{x}) - \mathbf{m}^\Phi) \\ & - \textstyle\sum_{i:y_i \neq k} \alpha_{i,o}^k (\phi(\mathbf{x}_i) - \mathbf{m}^\Phi)^T \mathbf{P}\tilde{\mathbf{S}}_w^{-1}\mathbf{P}^T(\phi(\mathbf{x}) - \mathbf{m}^\Phi) + b_{k,o}]. \end{aligned} \tag{101}$$

APPENDIX III
WOLF DUAL PROBLEM FOR THE OPTIMIZATION OF LAGRANGIAN (53)

At the saddle point, the solution should satisfy the KKT conditions, for $k = 1, \ldots, K$:

$$\begin{aligned} \nabla_{\boldsymbol{\eta}_k} L_4|_{\boldsymbol{\eta}_k = \boldsymbol{\eta}_{k,o}} &= 0 \Leftrightarrow \\ \boldsymbol{\eta}_{k,o} &= \tilde{\mathbf{S}}_w^{-1} \textstyle\sum_{i=1}^N (c_i^k A_{i,o} - a_{i,o}^k)(\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}_k). \end{aligned} \tag{102}$$

The other conditions are the same as (91) and (92).

By substituting (102) back into (53) we obtain:

$$\begin{aligned} L_4(\boldsymbol{\eta}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \ &= \textstyle\sum_{k=1}^K \sum_{i=1}^N \sum_{j=1}^N (c_i^k A_i - \alpha_i^k)(c_j^k A_j - \alpha_j^k)((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}_k)^T \tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}_k)) - \\ &- \textstyle\sum_{k=1}^K \sum_{i=1}^N \alpha_i^k [\sum_{j=1}^N (c_j^{l_i} A_j - \alpha_j^{l_i})((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}_{l_i})^T \tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}_{l_i})) - \\ &- \textstyle\sum_{j=1}^N (c_j^k A_j - \alpha_j^k)((\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}_k)^T \tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}_k))) + b_{l_i} - b_k - 2] - \\ &- \textstyle\sum_{k=1}^K \sum_{i=1}^N \alpha_i^k \xi_i^k + C \sum_{k=1}^K \sum_{i=1}^N \xi_i^k - \sum_{i=1}^N \sum_{k=1}^K \beta_i^k \xi_i^m. \end{aligned} \tag{103}$$

as in Appendix II the terms in $\boldsymbol{\xi}$ disappear and (103) becomes:

$$\begin{aligned} L_4(\boldsymbol{\eta}_k, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \ &= 2\textstyle\sum_{i,k} \alpha_i^k + \sum_{k=1}^K \sum_{i=1}^N \sum_{j=1}^N \delta_{i,j,k} (\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}_k)^T \tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}_k) \\ &- \textstyle\sum_{k=1}^K \sum_i^N \sum_j^N \lambda_{i,j,k} (\tilde{\mathbf{x}}_i - \tilde{\mathbf{m}}_{l_i})^T \tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j - \tilde{\mathbf{m}}_{l_i}) \end{aligned} \tag{104}$$

where $\delta_{i,j,k} = (c_i^k A_i - \alpha_i^k)(c_j^k A_j - \alpha_j^k) - \alpha_i^k (c_j^k A_j - \alpha_j^k)$ and $\lambda_{i,j,k} = \alpha_i^k (c_j^{l_i} A_j - \alpha_j^{l_i})$.

In order to isolate $\tilde{\mathbf{x}}_i^T\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{x}}_j$ in equation (104) we expanded $\tilde{\mathbf{m}}_k$ as $\tilde{\mathbf{m}}_k = \frac{1}{N_k}\sum_{\tilde{\mathbf{x}}\in\mathcal{U}_k}\tilde{\mathbf{x}} = \sum_{\rho=1}^{N}\nu_{k,\rho}\tilde{\mathbf{x}}_\rho$ where $\nu_{k,\rho} = \frac{1}{N_k}$ if $\tilde{\mathbf{x}}_\rho \in \mathcal{U}_k$ and $\nu_{k,\rho} = 0$ if $\tilde{\mathbf{x}}_\rho \in \mathcal{U}_k$. We expand the term as:

$$\begin{array}{l}\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\delta_{i,j,k}(\tilde{\mathbf{x}}_i-\tilde{\mathbf{m}}_k)^T\tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j-\tilde{\mathbf{m}}_k)\\ =\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\delta_{i,j,k}(\tilde{\mathbf{x}}_i\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{x}}_j-\tilde{\mathbf{m}}_k\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{x}}_j+\tilde{\mathbf{m}}_k^T\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{m}}_k-\tilde{\mathbf{x}}_i^T\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{m}}_k)\\ =\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(\delta_{i,j,k}-\sum_{\rho=1}^{N}\nu_{k,rho}\delta_{i,\rho,k}-\\ -\sum_{\rho=1}^{N}\nu_{k,rho}\delta_{\rho,j,k}+\sum_{\rho=1}^{N}\sum_{m=1}^{N}\nu_{K,\rho}\nu_{k,m}\delta_{\rho,m,k})\tilde{\mathbf{x}}_i^T\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{x}}_j\end{array} \tag{105}$$

while the other one is expanded as:

$$\begin{array}{l}\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_{i,j,k}(\tilde{\mathbf{x}}_i-\tilde{\mathbf{m}}_{l_i})^T\tilde{\mathbf{S}}_w^{-1}(\tilde{\mathbf{x}}_j-\tilde{\mathbf{m}}_{l_i})=\\ =\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(\tilde{\mathbf{x}}_i^T\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{x}}_j-\tilde{\mathbf{m}}_{l_i}^T\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{x}}_j-\tilde{\mathbf{x}}_j^T\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{m}}_{l_i}+\tilde{\mathbf{m}}_{l_i}^T\tilde{\mathbf{m}}_{l_i})\\ =\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(\lambda_{i,j,k}-\sum_{\rho=1}^{N}\nu_{l_\rho,\rho}\lambda_{\rho,j,k}-\sum_{\rho=1}^{N}\nu_{l_\rho,\rho}\lambda_{i,\rho,k}+\\ +\sum_{\rho=1}^{N}\sum_{m=1}^{N}\nu_{l_\rho,\rho}\nu_{l_m,m})\tilde{\mathbf{x}}_i\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{x}}_j.\end{array} \tag{106}$$

Thus, the Wolf dual problem is:

$$L_4(\boldsymbol{\eta}_k,\mathbf{b},\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\beta}) = W(\boldsymbol{\alpha}) = 2\sum_{i,k}\alpha_i^k + \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\omega_{i,j,k}\tilde{\mathbf{x}}_i\tilde{\mathbf{S}}_w^{-1}\tilde{\mathbf{x}}_j \tag{107}$$

where

$$\begin{array}{rl}\omega_{i_j,k} & =\delta_{i,j,k}-\sum_{\rho=1}^{N}\nu_{k,\rho}\delta_{i,\rho,k}-\sum_{\rho=1}^{N}\nu_{k,\rho}\delta_{\rho,j,k}+\sum_{\rho=1}^{N}\sum_{m=1}^{N}\nu_{K,\rho}\nu_{k,m}\delta_{\rho,m,k}+\\ & -\lambda_{i,j,k}+\sum_{\rho=1}^{N}\nu_{l_\rho,\rho}\lambda_{\rho,j,k}+\sum_{\rho=1}^{N}\nu_{l_\rho,\rho}\lambda_{i,\rho,k}-\sum_{\rho=1}^{N}\sum_{m=1}^{N}\nu_{l_\rho,\rho}\nu_{l_m,m}.\end{array} \tag{108}$$

After, solving the quadratic optimization problem (108) the decision function is:

$$\begin{array}{rl} f(\mathbf{x}) & =\mathrm{argmax}_{k=1,\ldots,K}[\mathbf{w}_{k,o}^T(\phi(\mathbf{x})-\mathbf{m}_k^\Phi)+b_{k,o}] = \mathrm{argmax}_{k=1,\ldots,K}[\boldsymbol{\phi}_{k,o}^T(\phi(\mathbf{x})-\mathbf{m}_k^\Phi)+b_{k,o}]\\ & =\mathrm{argmax}_{k=1,\ldots,K}[\sum_{i=1}^{N}(c_i^kA_{i,o}-\alpha_{i,o}^k)(\phi(\mathbf{x}_i)-\mathbf{m}_k^\Phi)^T\mathbf{P}\tilde{\mathbf{S}}_w^{-1}\mathbf{P}^T(\phi(\mathbf{x})-\mathbf{m}_k^\Phi)+b_{k,o}].\end{array} \tag{109}$$

REFERENCES

[1] K. Fukunaga, *Statistical Pattern Recognition*. San Diego: CA: Academic, 1990.

[2] S. Mika, R. G., J. Weston, B. Scholkopf, A. Smola, and K.-R. Muller, "Constructing descriptive and discriminative nonlinear features: Rayleigh coefficients in kernel feature spaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 623 – 628, 2003.

[3] J. Yang, A. Frangi, J. Yang, D. Zhang, and Z. Jin, "KPCA plus LDA: A complete kernel Fisher discriminant framework for feature extraction and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 230–244, 2005.

[4] V. Vapnik, *Statistical Learning Theory*. New York: J.Wiley, 1998.

[5] A. Tefas, C. Kotropoulos, and I. Pitas, "Using support vector machines to enhance the performance of elastic graph matching for frontal face authentication," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 7, pp. 735–746, 2001.

[6] K.-R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 181–201, 2001.

[7] I. Kotsia and I. Pitas, "Facial expression recognition in image sequences using geometric deformation features and support vector machines," *IEEE Transactions on Image Processing*, vol. 16, no. 1, pp. 172–187, Jan. 2007.

[8] Z. Zeng, M. Pantic, G. Roisman, and T. Huang, "A survey of affect recognition methods: Audio, visual and spontaneous expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence, accepted for publication*, 2008.

[9] P. Ekman and W. V. Friesen, *Emotion in the Human Face*. New Jersey: Prentice Hall, 1975.

[10] M. Pantic and L. Rothkrantz, "Automatic analysis of facial expressions: The state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1424–1445, December 2000.

[11] B. Fasel and J. Luettin, "Automatic facial expression analysis: A survey," *Pattern Recognition*, vol. 36, no. 1, pp. 259–275, 2003.

[12] M. Rydfalk, "CANDIDE: A parameterized face," Linkoping University, Tech. Rep., 1978.

[13] P. Michel and R. Kaliouby, "Real time facial expression recognition in video using support vector machines," in *Proceedings of 5th international conference on Multimodal interfaces*, Vancouver, British Columbia, Canada, 2003, pp. 258–264.

[14] O. Martin, F.-X. Fanard, and B. Macq, "From feature detection to facial expression recognition: An integrated probabilistic approach," in *7th International Workshop on Image Analysis for Multimedia Interactive Services*, Incheon, South Korea, April 2006 2006.

[15] E. Pekalska, P. Paclik, and R. Duin, "A generalized kernel approach to dissimilarity-based classification," *Journal of Machine Learning Research*, vol. 2, pp. 175–211, 2001.

[16] L. Chengjun, "Gabor-based kernel PCA with fractional power polynomial models for face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 572 – 581, May 2004.

[17] B. Haasdonk, "Feature space interpretation of SVMs with indefinite kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 482 – 492, 2005.

[18] L. Chengjun, "Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 5, pp. 725–737, 2006.

[19] J. Gower, "Euclidean distance geometry," *Mathematical Scientist*, vol. 7, pp. 1–14, 1982.

[20] L. Goldfarb, "A unified approach to pattern recognition," *Pattern Recognition*, vol. 17, pp. 575–582, 1984.

[21] ——, "A new approach to pattern recognition," *L.N. Kanal and A. Rosenfeld, editors, Progress in Pattern Recognition, Elsevier Science Publishers*, vol. 2, pp. 241–402, 1985.

[22] J. Gower, "Metric and euclidean properties of dissimilarity coefficients," *Journal of Classification*, vol. 3, pp. 5–48, 1986.

[23] I. Borg and P. Groenen, *Modern Multidimensional Scaling*. Springer-Verlag, New York, 1997.

[24] J. Weston and C. Watkins, "Multi-class Support Vector Machines," in *Proceedings of ESANN99*, Brussels, Belgium, 1999.

[25] C. L. Blake and C. J. Merz, "UCI repository of machine learning databases," Available: http://www.ics.uci.edu/mlearn/MLRepository.html, Dept. Inf. Comput. Sci., Univ. California, Tech. Rep., 1998, Tech. Rep., 1998.

[26] V. Hutson and J. Pym, *Applications of Functional Analysis and Operator Theory*. London: Academic Press, 1980.

[27] J. Weston and C. Watkins, "Multi-class Support Vector Machines, Tech. Rep. Techical report CSD-TR-98-04, 1998.

[28] B. Scholkopf and A. Smola, *Learning with Kernels*. Cambridge, MA: MIT Press, 2002.

[29] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass Support Vector Machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, March 2002.

[30] P. Shivaswamy and T. Jebara, "Ellipsoidal kernel machines," in *Artificial Intelligence in Statistics (AISTATS)*, March 2007.

[31] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Minining and Knowledge Discovery*, vol. 2, pp. 121–167, 1998.

[32] A. Scholkopf, B. Smola and K. R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, pp. 1299–1319, 1998.

[33] E. Kreyszig, *Introductory Functional Analysis with Applications*. John Wiley & Sons, 1978.

[34] MATLAB, *Users Guide*. The MathWorks, Inc., http:// www.mathworks.com, 1994-2001.

[35] H. Cevikalp, M. Neamtu, M. Wilkes, and A. Barkana, "Discriminative common vectors for face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 4–13, 2005.

[36] J. Yang and J.-Y. Yang, "Why can LDA be performed in PCA transformed space?" *Pattern Recognition*, vol. 36, no. 2, pp. 563–566, 2003.

[37] B. Scholkopf, "The kernel trick for distances," in *NIPS*, 2000.

[38] W. Greub, *Modern Linear Algebra*. Springer-Verlag, 1975.

[39] M. Pantic and L. J. M. Rothkrantz, "Expert system for automatic analysis of facial expressions," *Image and Vision Computing*, vol. 18, no. 11, pp. 881–905, August 2000.

[40] S. Krinidis and I. Pitas, "Statistical analysis of facial expressions for facial expression synthesis," *submitted in IEEE Transactions on Multimedia*, 2005.

[41] Y.Zhang and Q.Ji, "Active and dynamic information fusion for facial expression understanding from image sequences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 699–714, May 2005.

[42] I. Cohen, N. Sebe, S. Garg, L. S. Chen, and T. S. Huanga, "Facial expression recognition from video sequences: temporal and static modelling," *Computer Vision and Image Understanding*, vol. 91, pp. 160–187, 2003.

[43] S. Aleksic and K. Katsaggelos, "Automatic facial expression recognition using facial animation parameters and multi-stream hmms," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 1, pp. 3–11, 2006.

[44] M. Yeasin, B. Bullot, and R. Sharma, "Recognition of facial expressions and measurement of levels of interest from video," vol. 8, no. 3, June 2006, pp. 500–508.

[45] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 915–928, June 2007.

[46] K. I. Kim, K. Jung, and H. J. Kim, "Face recognition using kernel principal component analysis," *IEEE Signal Processing Letters*, vol. 9, no. 2, pp. 40–42, 2002.

[47] M. Gonen, A. Tanugur, and E. Alpaydin, "Multiclass posterior probability support vector machines," *IEEE Transactions on Neural Networks*, vol. 19, no. 1, pp. 130 – 139, 2008.

[48] K. Kim, M. Franz, and B. Scholkopf, "Iterative kernel principal component analysis for image modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 9, pp. 1351 – 1366, 2005.

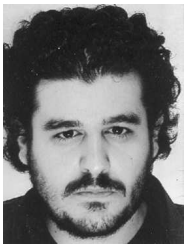
[49] T. Chin and D. Suter, "Incremental kernel principal component analysis," *IEEE Transactions on Image Processing*, vol. 16, no. 6, pp. 1662 – 1674, 2007.

[50] J. Kivinen, A. Smola, and R. Williamson, "Online learning with kernels," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2165–2176, 2004.

[51] K. Huang, H. Yang, I. King, and M. Lyu, "Maxi-Min Margin Machine: Learning large margin classifiers locally and globally," *IEEE Transactions on Neural Networks*, vol. 19, no. 2, pp. 260 – 272, 2008.

[52] D. Wang, D. Yeung, and E. Tsang, "Weighted mahalanobis distance kernels for support vector machines," *IEEE Transactions on Neural Networks*, vol. 18, no. 5, pp. 1453 – 1462, 2007.

Irene Kotsia was born in Kastoria, Greece. She received the B.Sc. and the Ph.D. degrees, from the Department of Informatics of Aristotle University of Thessaloniki, Greece, in 2002 and 2008, respectively. She has coauthored more than 19 journal and conference publications. She is currently a senior researcher at the Artificial Intelligence and Information Analysis (AIIA) Laboratory of the Department of Informatics. Her current research interests lie in the areas of image and signal processing, statistical pattern recognition especially for facial expression recognition from static images and image sequences as well as in the area of graphics and animation.



Stefanos Zafeiriou was born in Thessaloniki, Greece in 1981. He received the B.Sc, degree in Informatics with highest honors in 2003 and the Ph.D degree in Informatics in 2007, both from the Aristotle University of Thessaloniki, Thessaloniki, Greece. He has co-authored over than 30 journal and conference publications. During 2007-2008 he was a senior researcher at the Department of Informatics at the Aristotle University of Thessaloniki. Currently, he is a senior researcher at the Department of Electrical and Electronic Engineering at Imperial College London, UK. His current research interests lie in the areas of signal and image processing, computational intelligence, pattern recognition, machine learning, computer vision and detection and estimation theory. Dr. Zafeiriou received various scholarships and awards during his undergraduate, Ph.D. and postdoctoral studies.

Ioannis Pitas received the Diploma of Electrical Engineering in 1980 and the PhD degree in Electrical Engineering in 1985 both from the Aristotle University of Thessaloniki, Greece. Since 1994, he has been a Professor at the Department of Informatics, Aristotle University of Thessaloniki. From 1980 to 1993 he served as Scientific Assistant, Lecturer, Assistant Professor, and Associate Professor in the Department of Electrical and Computer Engineering at the same University. He served as a Visiting Research Associate at the University of Toronto, Canada, University of Erlangen- Nuernberg, Germany, Tampere University of Technology, Finland, as Visiting Assistant Professor at the University of Toronto and as Visiting Professor at the University of British Columbia, Vancouver, Canada. He was lecturer in short courses for continuing education. He has published over 600 journal and conference papers and contributed in 22 books in his areas of interest. He is the co-author of the books Nonlinear Digital Filters: Principles and Applications (Kluwer, 1990), 3-D Image Processing Algorithms (J. Wiley, 2000), Nonlinear Model-Based Image/Video Processing and Analysis (J. Wiley, 2001) and author of Digital Image Processing Algorithms and Applications (J. Wiley, 2000). He is the editor of the book Parallel Algorithms and Architectures for Digital Image Processing, Computer Vision and Neural Networks (Wiley, 1993). He has also been an invited speaker and/or member of the program committee of several scientific conferences and workshops. In the past he served as Associate Editor of the IEEE Transactions on Circuits and Systems, IEEE Transactions on Neural Networks, IEEE Transactions on Image Processing, EURASIP Journal on Applied Signal Processing and co-editor of Multidimensional Systems and Signal Processing. He was general chair of the 1995 IEEE Workshop on Nonlinear Signal and Image Processing (NSIP95), technical chair of the 1998 European Signal Processing Conference and general chair of IEEE ICIP 2001. His current interests are in the areas of digital image and video processing and analysis, multidimensional signal processing, watermarking and computer vision.