Pattern Recognition 46 (2013) 3441–3451

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Using robust dispersion estimation in support vector machines

N. Vretos *, A. Tefas, I. Pitas

Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece

ARTICLE INFO

Article history:
Received 11 July 2012
Received in revised form
20 March 2013
Accepted 4 May 2013
Available online 23 May 2013

Keywords:
Support vector machines
Minimum covariance determinant
Robust dispersion estimation

ABSTRACT

In this paper, a novel Support Vector Machine (SVM) variant, which makes use of robust statistics, is proposed. We investigate the use of statistically robust location and dispersion estimators, in order to enhance the performance of SVMs and test it in two-class and multi-class classification problems. Moreover, we propose a novel method for class specific multi-class SVM, which makes use of the covariance matrix of only one class, i.e., the class that we are interested in separating from the others, while ignoring the dispersion of other classes. We performed experiments in artificial data, as well as in many real world publicly available databases used for classification. The proposed approach performs better than other SVM variants, especially in cases where the training data contain outliers. Finally, we applied the proposed method for facial expression recognition in three well known facial expression databases, showing that it outperforms previously published attempts.

1. Introduction

The most recent and successful classification methods, in terms of generalization capabilities, are the margin machines [1], which maximize the margin between the data and the empirically calculated separation hyperplane. Support vector machines (SVMs) are a special case of margin machines that achieve good classification performance by maximizing the margin of the planes separating the different classes. SVMs has been used so far in many and diverse applications such as [2–10]. Many alternatives exist that make use of the margin maximization technique for classification purposes [1,11–13]. The Fisher ratio was embedded in the SVM optimization problem, so that class dispersion can be taken into account for a more accurate calculation of the separation hyperplane and, subsequently, the support vectors in [13]. In [14], a solution for the nonlinear class separation, using kernel PCA, was proposed. A multi-class version was proposed in [15]. Based on a similar idea, the so-called maximum relative margin machines (RMMs) were introduced in [1], proposing a trade-off between standard and dispersion-based SVMs. In [1], it is argued that even a slight affine data transformation can produce major changes in the SVM classification performance and, thus, embedding the data dispersion data in the SVM framework can tackle such problems. However, when first and second order statistics are used in order to calculate the location and dispersion and the location, an underlying Gaussianity assumption is made on the data, which is not always true. Based on these two remarks, a trade off between standard and dispersion-based SVMs is proposed in [1]. It has to be mentioned, that although in a theoretical point of view a change in regularization will have benefits in limited cases only, it has been shown in many publications, as the ones before mentioned, that such changes are useful in a vast area of applications.

There has been some attempts to tackle the outlier problem in SVMs so far. In [16], the authors propose a novel soft margin SVM formulation that includes an outlier indicator function to remove outliers. This approach differs from the one proposed here since in our approach we do not remove outliers, which can potentially be support vectors, but we only use them to better estimate the covariance matrix. On the other hand in [17], an approach that makes use of the Mahalanobis distance to calculate the covariance matrix is used to tackle outlying sample. In this approach the authors propose an one-class dispersion based SVM and show that they achieve better results by handling outliers with the mahalanobis formulation of the covariance. The proposed method is different in two aspects from the one proposed in [17] : first, in the robust calculation of the dispersion measure, where in our case we use the minimum covariance determinant and second, the fact that we propose one-class, multi-class as well as a specific-class SVM variants.

It is well known that data dispersion estimators suffer in the presence of outliers [18]. Sample covariance estimation may vastly degenerate in the presence of only one outlier. Many approaches exist, which deal with data outliers [19–21]. Their main goal is, firstly, to find a robust location and/or dispersion estimator with high breakdown value (i.e., the percentage of outlying samples that the estimator can tolerate) and, secondly, to construct an

* Corresponding author. Tel./fax: +30 2310996304.
E-mail addresses: vretos@iti.gr (N. Vretos), tefas@aiia.csd.auth.gr (A. Tefas), pitas@aiia.csd.auth.gr (I. Pitas).

0031-3203/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2013.05.016

estimator possessing the affine equivariance property [18]. A good survey for multivariate robust estimators can be found in [18]. The *minimum covariance determinant* (MCD) estimator is a robust multivariate dispersion estimator proposed in [22]. Given an initial data set $\mathcal{X}$, the objective is to find a data subset $\mathcal{E} \subseteq \mathcal{X}$, which possesses minimum covariance determinant. Then, its sample covariance estimation is used for covariance matrix estimation of the initial set $\mathcal{X}$. It is proven that MCD possess the affine equivariance property and, also, outperforms other estimators, in terms of breakdown value [23]. The main drawback of this approach is that it has a proven NP complexity. Hence, MCD was not applicable, until a fast version was developed in [24].

In this paper, we integrate the MCD robust covariance estimator in the SVM optimization problem. It will be proven that such an integration makes sense only when using robust estimators that possess the affine equivariance property. In the opposite case, no physical interpretation of the integration can be deduced, as will be detailed later on. This interpretation summarizes in that in the projective space the data dispersion needs to be optimized through the SVM optimization problem. If affine equivariance does not hold for the dispersion estimator, the corresponding criterion does not represent the robustly estimated projected data dispersion. Other dispersion estimators can be used as long as affine equivariance property holds. Although, MCD was chosen in that it has a fast implementation as well as better breakdown value than other. Moreover, we shall demonstrate that the use of MCD addresses the problematic Gaussian data assumption in a systematic way. Furthermore, we show that using class-specific covariance matrices outperforms the more global scatter matrix model, which takes into consideration all data classes. We shall also demonstrate that the solution of the SVM optimization problem for the multi-class case, using class-specific minimum covariance SVMs, follows the solution of the standard multi-class SVMs to some extent. This provides easy implementation schemes, due to the fact that a simple transformation of the input space can be made and the use of standard multi-class SVMs implementations can be used.

The novelty of this paper lies mainly in the integration of robust covariance estimators in the SVM optimization problem. More specifically, the MCD estimator is used to tackle problems of outlying samples in the training data space. We show that the use of robust statistics can be applied in such a framework, only under the affine equivariance property of the dispersion and location estimation. Class-specific minimum covariance SVMs are proposed, in contrast to the use of the standard scatter matrix firstly proposed in [13]. Based on these two major contributions, we also propose a novel multi-class SVM in a class-specific minimum covariance framework. We prove in this paper that the class-specific minimum variance SVMs optimization problem can be solved analytically. Finally, we prove that the nonlinear separation hyperplanes can be deduced in the case of robust statistics SVMs, under a mild assumption on the kernel function. This assumption is that the kernel function conforms to the Mercer conditions, which is the case for the most common kernel functions such as RBF and polynomial ones.

In summary the paper novelties are the following:

- Use of the minimum covariance determinant estimation of the covariance based SVMs methods as the ones proposed in [13,14].
- Extension to the nonlinear case with robust estimation of the mean and the covariance in the feature space by using the same approach as in [14] but with the main difference of the robust calculation in the feature space.
- Introducing the class specific minimum covariance SVMs. In contrast to the multi-class minimum covariance based SVMs proposed in [15], which make use of the scatter matrix of all classes.
- Minimum covariance determinant estimation for the class specific minimum covariance SVM.

The paper is organized as follows. The Minimum Covariance Determinant procedure is analyzed in 2.1. The SVM optimization problem is outlined in Section 2.2. The integration of a robust dispersion estimator and the corresponding SVM optimization problem is solved in Section 3 for the two-class problem. In the same Section, we show that nonlinear kernel SVMs can be handled using MCD. In Section 4, the optimization problem for the multi-class cases of the class-specific minimum variance SVMs is solved. It is proven that the solution is similar to that of the standard multi-class SVMs. Experimental results are shown and discussed in Section 5. Finally, conclusions are drawn in Section 6.

2. Problem statement

2.1. Minimum covariance determinant

Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ be a sample of N observations in $\mathbb{R}^d$ also forming the data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T$. Moreover, take $h \leq N$. By definition, the MCD problem consists of finding a subset $\mathcal{E}$ such that

$$\mathcal{E} = \arg \min_{\mathcal{E} \subset \mathcal{X}, |\mathcal{E}| = h} \det(\hat{\mathbf{S}}_{\mathcal{E}}), \tag{1}$$

where $|\cdot|$ the set cardinality, $det(\cdot)$ the determinant of a matrix and $\hat{\mathbf{S}}_{\mathcal{E}}$ is the sample covariance matrix based on the observations $x_i \in \mathcal{E}$. The MCD estimates [25] are then given by

$$\hat{\boldsymbol{\mu}} = \sum_{\mathbf{x}_i \in \mathcal{X}} p_i \mathbf{x}_i \tag{2}$$

$$\hat{\mathbf{S}}_{\mathcal{X}} = \sum_{\mathbf{x}_i \in \mathcal{X}} p_i (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^T, \tag{3}$$

where $p_i = 1/h$ for i such that $\mathbf{x}_i \in \mathcal{E}$ and $p_i = 0$ for $\mathbf{x}_i \notin \mathcal{E}$.

The minimum covariance determinant dispersion and location estimator finds the subset $\mathcal{E} \subseteq \mathcal{X}$, that has a covariance matrix with minimal determinant among all possible subsets of $\mathcal{X}$ [22]. Therefore, the sample covariance matrix and the sample mean of this subset $\mathcal{E}$ are returned as dispersion and location estimations of the original data. As it is proven in [22], MCD is optimal for the class of elliptical distributions, which is a broaden class of distribution than the Gaussian. In Fig. 1, the robust and classical tolerance

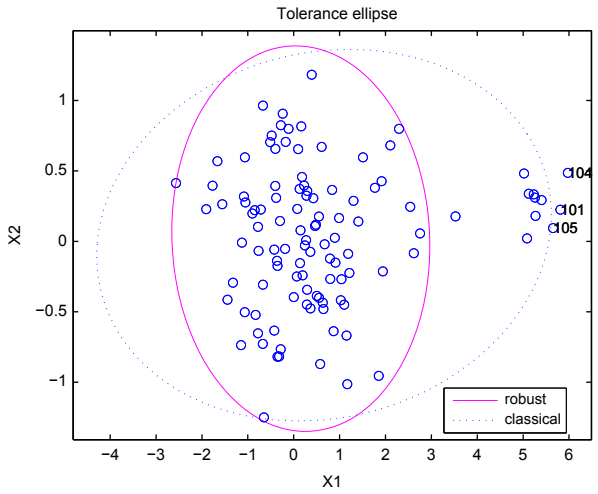

Fig. 1. Tolerance ellipse for a bivariate distribution.

ellipse of a bivariate distribution are shown. The tolerance ellipse is defined as the set of points $\mathbf{x}_i$ whose Mahalanobis distance equals the square root of the 0.975 quantile of the χ^2 distribution with 2 degrees of freedom $\sqrt{\chi^2_{2,0.975}}$ [22].

The value h is considered as the minimum number of points which must not outlay and it is an input parameter of the MCD algorithm. The MCD has its highest possible breakdown at $h=\lceil (n+d+1)/2 \rceil$, where $\lceil \cdot \rceil$ is the "smallest integer greater than" function. Although MCD calculation is proven to be NP-hard [26], a fast version has been proposed in [24]. Since then, MCD has attracted a lot of attention [27–29].

Our aim is to use this definition of the covariance matrix for a robust estimation of the covariance matrices, which are involved in the case of dispersion based SVMs, such as in [13,14].

2.2. Minimum covariance support vector machines

In a two-class classification problem, SVM classification is defined as follows: let $\{(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2),\ldots,(\mathbf{x}_N,y_N)\}$, be a set of N labeled d-dimensional vectors $\mathbf{x}_i \in \mathbb{R}^d$ and their corresponding class labels $y_i \in \{-1,1\}$, $i=[1,..,N]$. The aim is to find an optimal separation hyperplane in $\mathbb{R}^d$, which can separate the two-class data with a maximum margin, leading to the following optimization problem [30]:

$$\min_{\mathbf{w},b} \tfrac{1}{2}\mathbf{w}^T\mathbf{w}, \tag{4}$$

subject to (s.t.)

$$y_i(\mathbf{w}^T\mathbf{x}_i+b)\geq 1, \text{ with } i=1,..,N, \tag{5}$$

where $\mathbf{w}$,b are the normal vector and the corresponding bias term of the SVM hyperplane. Throughout the paper, we have omitted proofs related to error bounds (slack variables) for better paper readability, since all proofs that follow can be easily transformed to integrate slack variables. However, we have to mention that the slack variables trick has some well known implication on the Lagrange multipliers, which, in the case of nonseparable classes, will be bounded from above by a positive constant [30]. In [13], the authors, inspired from the Fisher discriminant ratio [31], proposed its integration in the SVM optimization problem, by modifying (4) to become:

$$\min_{\mathbf{w},b} \mathbf{w}^T\mathbf{S}_w\mathbf{w}. \tag{6}$$

The minimization of (6) subject to (5) leads to the so-called minimum covariance SVMs (MCVSVMs), where $\mathbf{S}_w$ is the within class scatter matrix, as defined in [13]. That is

$$\mathbf{S}_w=\sum_{k=1}^{K}\sum_{i=1}^{N_k}(\mathbf{x}_i-\boldsymbol{\mu}_k)(\mathbf{x}_i-\boldsymbol{\mu}_k)^T, \tag{7}$$

where K the number of classes in the classification problem, N_k is class cardinality and $\boldsymbol{\mu}_k$ is the mean vector of each class.

By using practically the same idea, i.e., using training data dispersion, a generalized approach was proposed in [1], which measure margins relative only to the data spread in any projection direction. Therein, the Gaussian assumption on the data is dropped and a more general optimization problem is defined consisting of (4) and (5) and a constraint on the radius of the projected data [1]:

$$\min_{\mathbf{w},b} \tfrac{1}{2}\mathbf{w}^T\mathbf{w}, \tag{8}$$

s.t.

$$y_i(\mathbf{w}^T\mathbf{x}_i+b)\geq 1 \tag{9}$$

$$\frac{1}{2}(\mathbf{w}^T\mathbf{x}_i+b)^2\leq\frac{B^2}{2}, \tag{10}$$

where B bounds the projected data. In their approach, the quadratic constraint in (10) is simplified and replaced by two linear constraints, for implementation purposes. Although this method provides better results, data outliers are not treated systematically in the statistical sense. However, a solution is proposed to relax the minimization problem using an arbitrary multivariate variable [1]. Although this approach is proposed to handle outliers and also deals in some way with the non-Gaussianity of data, it is based on assumptions which may or may not hold in all cases. On the other hand, our method tries to solve these issues in a more systematic way by using solid, statistically correct, robust techniques.

In our approach, we modify the SVM optimization problem (6) using the MCD estimator in order to handle outlying training data in a more systematic way. Moreover, the use of MCD assumes only that the data follow an elliptical distribution, which is a much larger class of distributions than the Gaussians ones. This relaxes the Gaussian assumption in [1] to the more general class of elliptical distributions.

The general SVM optimization can be written as follows:

$$\min_{\mathbf{w},b} \mathbf{w}^T\mathbf{A}\mathbf{w}, \tag{11}$$

under the same separability criterion as in (5), where $\mathbf{A}$ is a dispersion matrix of the initial data, e.g., the intra-scatter matrix [13], or a whitening operator [1], or MCD (in our case). It can be solved using the Lagrangian function:

$$\mathcal{L}(\mathbf{A},\mathbf{w},b,\boldsymbol{\alpha})=\mathbf{w}^T\mathbf{A}\mathbf{w}-\sum_{i=1}^{N}\alpha_i y_i(\mathbf{w}_i^T\mathbf{x}_i+b-1). \tag{12}$$

Its gradient with respect to $\mathbf{w}$ must be equal to $\mathbf{0}$:

$$\frac{\partial\mathcal{L}(\mathbf{A},\mathbf{w},b,\boldsymbol{\alpha})}{\partial\mathbf{w}}=2\mathbf{A}\mathbf{w}-\sum_{i=1}^{N}\alpha_i y_i\mathbf{x}_i=\mathbf{0}, \tag{13}$$

in order to find the optimal hyperplane $\mathbf{w}_o$, assuming that $\mathbf{A}$ is nonsingular:

$$\mathbf{w}_o=\frac{1}{2}\mathbf{A}^{-1}\sum_{i=1}^{N}\alpha_{i,o}y_i\mathbf{x}_i, \tag{14}$$

where $\alpha_{i,o}$ are the Lagrange multipliers for the optimal solution that can be found by the dual problem defined as

$$\max_{\boldsymbol{\alpha}}\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j\mathbf{x}_i^T\mathbf{A}^{-1}\mathbf{x}_j, \tag{15}$$

subject to the constraints:

$$\sum_{i=1}^{N}\alpha_i y_i=0 \text{ and } \alpha_i\geq 0. \tag{16}$$

By writing (15) in a matrix form, we obtain:

$$\max_{\boldsymbol{\alpha}}\boldsymbol{\alpha}^T\mathbf{1}-\tfrac{1}{2}\boldsymbol{\alpha}^T\mathbf{G}\boldsymbol{\alpha}, \tag{17}$$

where matrix $\mathbf{G}$, has elements $G_{ij}=y_i y_j\mathbf{x}_i^T\mathbf{A}^{-1}\mathbf{x}_j$ and $\mathbf{1}$ is a vector containing ones. By comparing the dual of the SVM optimization problem (15) with the one in the standard SVM case, we deduce that samples $\mathbf{x}_i$ can be transformed with the use of $\mathbf{A}$:

$$\mathbf{x}'_i=\mathbf{A}^{-1/2}\mathbf{x}_i, \tag{18}$$

since in the standard SVM formulation, the matrix $\mathbf{G}'$ is used, having elements $G'_{ij}=y_i y_j\mathbf{x}_i^T\mathbf{x}_j$.

The criterion of the dual problem is derived from the well known Karush–Kuhn–Tucker conditions (KKT) [32]. These conditions imply that, for a saddle point, the followings equations must hold:

$$\nabla_{\mathbf{w}}\mathcal{L}|_{\mathbf{w}=\mathbf{w}_o}=\mathbf{0}$$

$$\frac{\partial\mathcal{L}}{\partial b}\Big|_{b=b_o}=0$$

$$\alpha_{i,o} \geq 0, b_{i,o} \geq 0 \tag{19}$$

Finally, the optimal separation hyperplane parameter b_o can be easily deduced by averaging Eq. (5) for all support vectors $\mathbf{x}_i,\ i \in D = \{i : a_{o,i} > 0\}$:

$$b_o = \frac{1}{N} \sum_{i \in D} \left(y_i - \frac{1}{2} \sum_{j=1}^{N} y_j \alpha_j \mathbf{x}_j^T \mathbf{A}^{-1} \mathbf{x}_i \right) \tag{20}$$

As can be seen in (14), matrix $\mathbf{A}$ must be nonsingular. This constraint, though, is very restrictive in many problems especially in image processing due to the so-called "curse of dimensionality", when we have less training samples than data dimensions. We shall see later on, in detail, how to overcome this issue in the proposed SVM scheme. A second more subtle constraint for $\mathbf{A}$ is that it has to operate in the same way on both the data and on their projections, in order to be able to formulate the SVM optimization problem as in (11) and minimize the robust dispersion of the projected samples. This interpretation summarizes in that in the projective space the data dispersion needs to be optimized through the SVM optimization problem. If affine equivariance does not hold for the dispersion estimator, the corresponding criterion does not represents the robustly estimated projected data dispersion. $\mathbf{A}$ is an affine equivariant estimator of the data dispersion [18], if it can be properly transformed under data rotation, translation and scaling. If $\mathbf{S}(\cdot)$, $\mathbf{m}(\cdot)$ are a dispersion and location estimator respectively, they are affine equivariant iff:

$$\mathbf{m}(\mathbf{RX} + \mathbf{t}) = \mathbf{R}\mathbf{m}(\mathbf{X}) + \mathbf{t} \tag{21}$$

$$\mathbf{S}(\mathbf{RX} + \mathbf{t}) = \mathbf{R}\mathbf{S}(\mathbf{X})\mathbf{R}^T, \tag{22}$$

where $\mathbf{R}$ is a nonsingular matrix and $\mathbf{t}$, is arbitrary translation vector, respectively, while $\mathbf{X}$ denotes the matrix of all samples $\mathbf{x}_i$, $i = 1, .., N$. Unfortunately, most of the univariate robust estimators for location and dispersion are not affine equivariant in their multivariate form. As an example, the median and the associated Median Absolute Deviation (MAD) location and dispersion estimators, which are the best robust estimators in terms of breakdown values, are not affine equivariant in their multivariate version. A good review for statistical robust dispersion and location estimators can be found in [18]. On the other hand, the usual sample covariance matrix, is affine equivariant but not robust [18]. Many robust affine equivariant estimators have been proposed in the literature [18]. In our case we shall use the MCD covariance estimator, due to its high breakdown value, in comparison to other covariance estimators and then integrate it to the support vector minimization problem (11). We propose two novel variants for the SVM, one that makes use of the robust estimation of the covariance matrix and therefore follows the same path as in [13], and a second one where we are only interested in the covariance matrix of just one data class.

3. MCVSVM with robust statistics (RobMCVSVM)

In the case of RobMCVSVM for the two class problem, the inner-class scatter matrix $\mathbf{A}_R$ is defined as

$$\mathbf{A}_R = MCD(\mathcal{C}_1) + MCD(\mathcal{C}_2), \tag{23}$$

where $\mathcal{C}_k$ (with $k \in \{1,2\}$) is the subset of $\mathcal{X}$ containing samples from the first and second class respectively, and, $MCD(\mathcal{C}_k)$ is the robustly estimated dispersion of the first and second class respectively calculated by the MCD estimator. It has to be noted that summing the covariance matrix estimates of each class estimated using MCD, is equivalent to calculating an MCD-wise robust within class scatter matrix. It is straightforward to apply $\mathbf{A}_R$ to the SVM minimization problem (11). There is a subtle difference between outlier trimming before using MCVSVM and RobMCVSVM. The latter uses (23) for robust dispersion estimation and, therefore, uses all training samples in (15), whereas outlier trimming followed by MCVSVM can possibly result in support vector trimming, since true support vectors are in class periphery and tend to outly. Therefore, it can be easily guessed that outlying samples are probable candidates for support vectors. Therefore, their exclusion may lead in classification accuracy drop.

3.1. Nonlinear SVMs with robust statistics

We shall show that robustly calculated covariance matrices can be used as well with nonlinear decision surfaces. Nonlinear SVMs are formulated as a minimization problem in a Hilbert space $\mathcal{H}$ (feature space) induced by a kernel function under the Mercer conditions [30]. Let us define a mapping $\phi : \mathbb{R}^d \to \mathcal{H}$, which maps the samples from $\mathcal{X}$ to an arbitrary Hilbert space. For the case of RobMCVSVM, Eq. (23) takes the form:

$$\mathbf{A}_R^{\phi} = \frac{1}{h_1} \sum_{\mathbf{x} \in \mathcal{E}_1} (\phi(\mathbf{x}) - \hat{\boldsymbol{\mu}}_{\mathcal{E}_1}^{\phi})(\phi(\mathbf{x}) - \hat{\boldsymbol{\mu}}_{\mathcal{E}_1}^{\phi})^T + \frac{1}{h_2} \sum_{\mathbf{x} \in \mathcal{E}_2} (\phi(\mathbf{x}) - \hat{\boldsymbol{\mu}}_{\mathcal{E}_2}^{\phi})(\phi(\mathbf{x}) - \hat{\boldsymbol{\mu}}_{\mathcal{E}_2}^{\phi})^T, \tag{24}$$

where $\mathcal{E}_k$ with $k \in \{1,2\}$ are the subsets calculated from the MCD optimization problem in (1) for each class with cardinality h_1 and h_2 respectively, and $\hat{\boldsymbol{\mu}}_{\mathcal{E}_k}^{\phi}$ is the robustly calculated mean vector of each class in the feature space $\mathcal{H}$ according to (2). In [14], kernel Principal Components Analysis (KPCA) is performed and it is proven that the application of linear MCVSVM with KPCA scores is equivalent with kernel MCVSVM. In our case though, we want to create a robust SVM framework, which is not the case in [14], due to the non-robust nature of KPCA. To handle kernel versions, we propose the following procedure. As it is discussed in [33], in order for kernelized methods to be able to classify data in the feature space, samples must preserve their topographic ordering in the feature space as well. Therein the authors claim that this is true if the mapping ϕ is smooth and continuous, which is the case for all Mercer kernels. Under this assumption, it is reasonable to consider that outliers detected through MCD in the sample space will be equivalently MCD wise outliers in the feature space as well. In order to solve the problem in the kernel space, we first perform the MCD algorithm in the sample space, in order to find the subset $\mathcal{E}$ of the samples and subsequently, we calculate the mean and covariance matrix in the feature space using the previously calculated $\mathcal{E}$ and the same approach as in [14]. As discussed in [14], to do so, we need to find a transformation $\mathbf{P}$ which projects samples from the feature space to another of lower dimensions as $\mathbf{x}'_i = \mathbf{P}^T \phi(\mathbf{x}_i)$, $\phi(\mathbf{x}_i) \in \mathcal{H}$. They prove that this is equivalent with projecting the sample space through the KPCA transform. Thus, $\mathbf{x}'_i$ are the KPCA score. Under these conditions, the mean vector in the feature space can be calculated as

$$\hat{\boldsymbol{\mu}}_{\mathcal{E}_k}^{\phi} = \frac{1}{h_k} \sum_{\mathbf{x}'_i \in \mathcal{E}_k} \mathbf{x}'_i, \tag{25}$$

where $\mathcal{E}_k$ is the subset of samples belonging to $\mathcal{X}$ which are not outliers of class k, according to the MCD algorithm. In a second step we map all samples through the transformation operator $\mathbf{P}$ (i.e., the KPCA) and, in the minimization problem, we use the mean vectors found earlier to calculate the scatter matrix using (24).

3.2. The "curse" of dimensionality

One often refers to the "curse" of dimensionality for cases, where the number of training samples is less than the data dimension. Its implication is the singularity of the sample covariance matrix. A typical way to tackle such problems is the use of

dimensionality reduction methods such as Principal Components Analysis (PCA). In [14], the authors prove that an equivalent minimization problem can be deduced through PCA projections, which subsequently leads to the optimal separating hyperplanes. Since PCA uses the non-robust sample covariance matrix, in the case of robust SVMs, we propose to use a robust version of PCA proposed in [21]. Therein, a MCD-based calculation of PCA is proposed, with an additional criterion of a projection pursuit calculated mean, firstly proposed in [34]. This robust PCA method, uses the same approach as the one in MCD, in order to find the covariance matrix for PCA. Therefore, this approach is better suited to our case. Nevertheless, we use this method only for dimensionality reduction, while the rest of the RobMCVSVM method is applied with the MCD as previously described. This is done, because, once we robustly find the principal components, we project all (i.e., outliers included) data samples to the principal axes. Therefore, RobMVCSVM can be calculated efficiently using (23), with the robust PCA-wise projected samples.

4. MultiClass robust support vector machines

In the case of multi-class SVMs, the minimization problem is defined so that all separation hyperplanes are integrated in the same minimization problem.

In [15], the minimization problem was enriched with the data dispersion information in the same manner as in [14], with the exception that multi-class SVMs were considered. Therein, the minimization problem for multiclass SVMs is solved and decision functions are extracted for the multi class case. In the case of RobMCVSVM, the application is straightforward by simply using an equivalent equation to (23) to calculate the multi class $\mathbf{A}_R$ in the place of $\mathbf{S}_w$.

4.1. Class specific minimum variance support vector machines (CSMVSVM)

As discussed earlier, the use of intra-class scatter matrix in the SVM optimization creates a condensed class representation along a specific projection direction. We propose to create a more specific classification framework to treat each class separately, by using k SVMs, one for each class. In this approach, each class specific SVM can ignores the dispersion of the other classes and, at the same time, condense only the projected samples of this specific class. To this end, we estimate the sample covariance matrix for each class as

$$\mathbf{A}_k = \frac{1}{|C_k|}\sum_{\mathbf{x}\in C_k}(\mathbf{x}-\boldsymbol{\mu}_{C_k})(\mathbf{x}-\boldsymbol{\mu}_{C_k})^T, \tag{26}$$

to be integrated in the minimization framework:

$$\min_{\mathbf{w}_k,b_k}\sum_{k=1}^{K}\mathbf{w}_k^T\mathbf{A}_k\mathbf{w}_k, \tag{27}$$

s.t.

$$(\mathbf{w}_{l_i}^T\mathbf{x}_i + b_{l_i}) \geq (\mathbf{w}_k^T\mathbf{x}_i + b_k) + 2 \quad i=1,..,N \quad k\in\{1,..,K\}\backslash l_i. \tag{28}$$

Trying to solve (27), under separability constraints in (28), results to the following formulation of the Lagrangian function:

$$\mathcal{L}(\mathbf{A}_k,\mathbf{w}_k,b_k,\boldsymbol{\alpha}_k) = \sum_{k=1}^{K}\mathbf{w}_k^T\mathbf{A}_k\mathbf{w}_k - \sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}((\mathbf{w}_{l_i}-\mathbf{w}_k)^T\mathbf{x}_i + (b_{l_i}-b_k-2)). \tag{29}$$

We shall introduce two notations for better readability of the equations:

$$R_i = \sum_{k=1}^{K}\alpha_{k,i}, \tag{30}$$

$$c_{k,i} = \begin{cases} 1, & \text{if } k = l_i \\ 0, & \text{if } k \neq l_i \end{cases} \tag{31}$$

The gradient of the Lagrangian function can be set equal to 0:

$$\frac{\partial\mathcal{L}(\mathbf{A}_k,\mathbf{w}_k,b_k,\boldsymbol{\alpha}_k)}{\partial\mathbf{w}_k} = 2\mathbf{A}_k\mathbf{w}_k + \sum_{i=1}^{N}(c_{k,i}R_i-\alpha_{k,i})\mathbf{x}_i = \mathbf{0}, \tag{32}$$

leading to the solution:

$$\mathbf{w}_k = \frac{1}{2}\mathbf{A}_k^{-1}\sum_{i=1}^{N}(c_{k,i}R_i-\alpha_{k,i})\mathbf{x}_i, \tag{33}$$

The second KKT condition:

$$\frac{\partial\mathcal{L}(\mathbf{A}_k,\mathbf{w}_k,b_k,\boldsymbol{\alpha}_k)}{\partial\mathbf{b}_k} = \mathbf{0}, \tag{34}$$

yields:

$$\sum_{i=1}^{N}\alpha_{k,i} = \sum_{i=1}^{N}c_{k,i}R_i \tag{35}$$

By substituting (33) and (34) in (29) we have

$$\mathcal{L}(\mathbf{A}_k,\mathbf{w}_k,b_k,\boldsymbol{\alpha}_k) = 2\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i} + \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\frac{1}{4}c_{l_i,j}R_iR_j - \frac{1}{2}\alpha_{k,i}\alpha_{l_i,j} + \frac{1}{4}\alpha_{k,i}\alpha_{k,j}\right)\cdot\mathbf{x}_i^T\mathbf{A}_k^{-1}\mathbf{x}_j. \tag{36}$$

Details for the calculation of (36) can be found in Appendix A. From (36), we can form the dual problem as:

$$\max_{\boldsymbol{\alpha}_k}\mathcal{D}(\boldsymbol{\alpha}_k) = 2\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i} + \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\frac{1}{4}c_{l_i,j}R_iR_j - \frac{1}{2}\alpha_{k,i}\alpha_{l_i,j} + \frac{1}{4}\alpha_{k,i}\alpha_{kj}\right)\cdot\mathbf{x}_i^T\mathbf{A}_k^{-1}\mathbf{x}_j, \tag{37}$$

s.t.

$$\sum_{i=n}^{N}\alpha_{k,i} = \sum_{i=1}^{N}c_{k,i}R_i \text{ and } \alpha_{k,i}\geq 0. \tag{38}$$

Eq. (37) is a quadratic function of $\boldsymbol{\alpha}$ and its matrix form can be deduced in the same way as in [35], by replacing the kernel matrix therein with $\mathbf{A}_k$, in order to solve the optimization problem as a single problem. This problem can be decoupled in K different SVM problems, one for each class [35]. For each SVM, we transform the data samples as in (18). Classification is then performed using the simple voting method between the results of the K class-specific SVMs. In case of equal votes, for some classes, we arbitrarily choose one out of the ones having equal votes. However, as will be seen in Section 5, such cases seldom occur. The robust version of this method can be obtained using the MCD algorithm for the calculation of $\mathbf{A}_k$. The algorithm for training and testing the RobCSMVSVM are presented in Algorithms 1 and 2. In these algorithms, *SVMTRAIN* and *SVMPREDICT* can be any linear SVM implementation.

Algorithm 1. Algorithm outline for the RobCSMVSVM training.

1: **Input:** Samples data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^T$, labels vector $\mathbf{y} = [y_1, y_2, \ldots, y_N]$ and SVM parameters
2: **Output:** The k trained SVM models $\mathbf{M}_k$
3: **Initialize: forall** k $\mathbf{E}_k \leftarrow \{\}$ and $\mathbf{X'}_k \leftarrow []$.
// Calculate the robust covariance matrix $\mathbf{A}_k$, for each class based on the vector $\mathbf{y}$ and matrix $\mathbf{X}$ and then project the samples with the equivalent inverse matrix
4: **for** $i = 1 \rightarrow N$ **do**

5: $\mathbf{E}_{y_i} \leftarrow \mathbf{E}_{y_i} \cup \{\mathbf{x}_i\}$
6: **end for**
7: **for** $k = 1 \rightarrow K$ **do**
8: $\mathbf{A}_k = MCD(\mathbf{E}_k)$
9: **end for**
// transform the samples with (18)
10: **for** $k = 1 \rightarrow K$ **do**
11: $\mathbf{X}' = \mathbf{A}_k^{-\frac{1}{2}}\mathbf{X}$
12: $\mathbf{M}_k \leftarrow SVMTRAIN(\mathbf{y}, \mathbf{X}', params)$
13: **end for**

Algorithm 2. Algorithm outline for the RobCSMVSVM testing.

1: **Input:** the robust covariance matrices $\mathbf{A}_k$, the trained models $\mathbf{M}_k$ and the test sample vector $\mathbf{x}_{test}$
2: **Output:** The predicted class label *label*
3: **for** $k = 1 \rightarrow K$ **do**
4: $\mathbf{x}'_{test} = \mathbf{A}_k^{-\frac{1}{2}}\mathbf{x}_{test}$
5: $label_k \leftarrow SVMPREDICT(\mathbf{M}_k, \mathbf{x}'_{test})$
6: **end for**
7: majority voting on $label_k$ gives the final *label*

5. Experimental results

We have conducted experiments on artificial data, as well as in real application, in order to provide evidence that the proposed method performs better in cases where outliers are present. We performed experiments on publicly available databases for classification benchmarking to show that our method outperforms other SVM variants on these databases. Finally, we have conducted facial expression recognition experiments on 3 different databases: Cohn–Kanade [36], BU-3DFE [37] and JAFFE [38]. We have used the *libsvm* software in our experiments, a very efficient SVM implementation [39].

5.1. Experiments with artificial data

We used artificially created data to pinpoint the fact that, in the presence of outliers our method performs better than both classical SVMs and MCVSVMs. First, two different two class problems are investigated. One using half-moon shaped classes, as in Fig. 2 and one with two Gaussian classes as in Fig. 4. In Figs. 3 and 5, the classes are shown, after adding a uniform noise. In the first case we add noise on each class in a way that the outliers degenerate the sample covariance matrix of the original data and in the second case we add a uniform noise in the whole space and therefore randomly chose from outlying samples for each class. A 5-fold cross validation test is performed in 100 different experiments, where the noisy data were created by reinitializing the noise random seed. In each experiment the mean accuracy over all folds is taken as the final classification accuracy. The results for the 5 different classifiers are depicted in Table 1 as the average of the 100 different experiments. Fig. 4

Both CSMVSVM and MCVSVM methods, which use the sample covariance matrix have reduced performance in the presence of noise, compared to their robust counterparts. The standard SVM, which uses no information on data dispersion, achieves better classification accuracy rate than CSMVSVMs and MCVSVMs in the presence of outliers, since it can handle them in a better way as it does not employ poor dispersion estimations. The robust versions outperform all others, because the data dispersion is much more accurately calculated, once the outliers are handled. In case of nonlinear artificial data such as co-centric cycles, our framework will perform as classical SVMs since in that case the covariance matrix will approximate the identity matrix and thus it will have no effect in the SVM algorithm. Such extreme cases of nonlinearity that can not make use of the dispersion of each class as a supplementary information has no effect in our framework and thus the proposed framework results to classical SVMs.

For comparison reasons, we have implemented the artificial data experiment described in [16]. It consists of two Gaussian distributions contaminated with a uniform noise in a ring of radius R to $R+1$. In there, the authors create two bivariate distributions with $\mu = [3, -3]$, $\Sigma = \begin{bmatrix} 20 & 16 \\ 16 & 20 \end{bmatrix}$ and $-\mu$ respectively. The outliers are

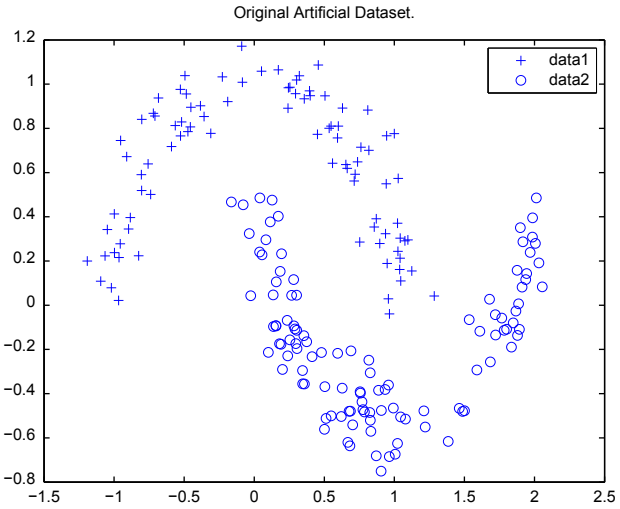

Fig. 2. Two balanced half-moon data classes.

Table 1
Mean classification accuracy on artificial data in 100 experiments.

Method	First experiment (%)	Second experiment (%)
RobCSMVSVMs	**99.64**	**95.34**
RobMCVSVMs	98.52	94.45
CSMVSVMs	90	88.93
MCVSVMs	89.28	87.52
SVMs	97.85	91.31

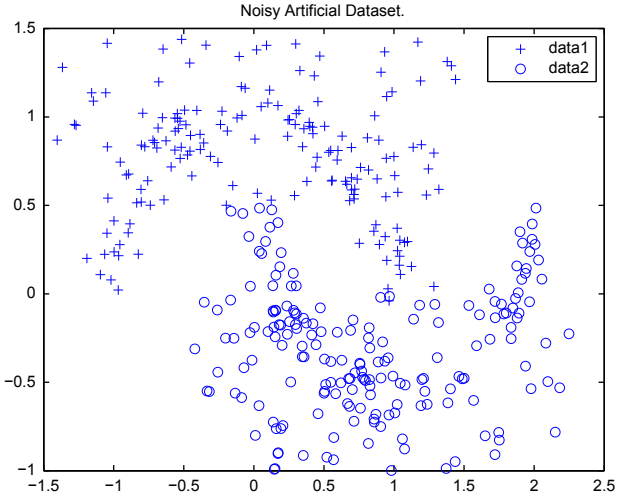

Fig. 3. Contaminated data set with uniformly distributed noise.

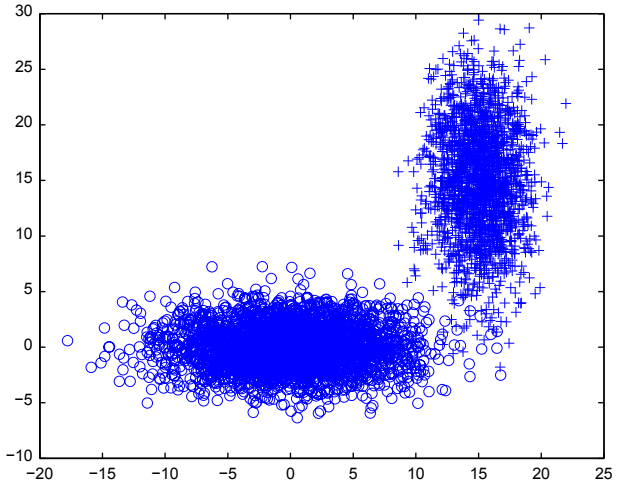

Fig. 4. Two Gaussian data classes.

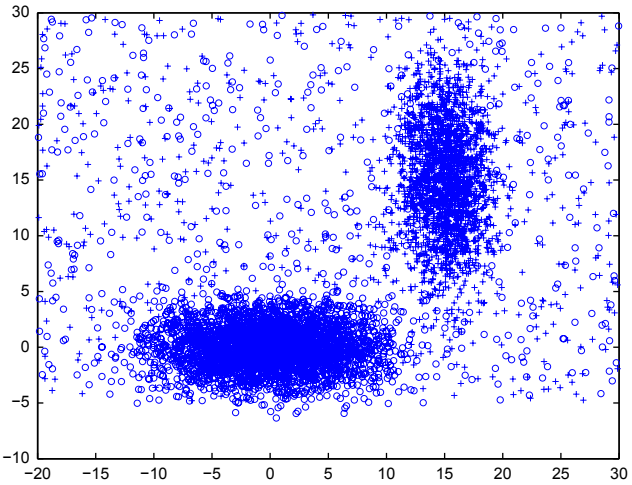

Fig. 5. Contaminated data set with uniformly distributed noise.

selected from different rings with $R=15,35,55,75$. In all experiments, the training set contains 50 samples, 20 from each distribution and 10 randomly labeled from the ring. Finally, the test set contains 2000 samples, 1000 from each class. The experiment was conducted 50 times with reinitialization of both the training and the test set. The results of the classical SVM and the RobCSMVSVM are drawn in Fig. 6. For comparison the reader may refer to [16] to see how our method performs in comparison with several others reported in there. It is obvious that the proposed approach performs better than the ones mentioned in [16] as can be seen from Fig. 6 in both, classification error as well as the robustness towards elevated limits of noise.

Based on the same dataset we have conducted experiments with different values for the parameter h of the *MCD* procedure. The results are shown in Fig. 7. As can be seen, the parameter h is optimal for values near $0.75|\mathcal{X}|$. This result was expected since h is a threshold between statistical significance and outlier robustness. Moreover, this result is also reported in [20]. For small values of h (e.g., near $0.55|\mathcal{X}|$), the samples that are used to calculate the covariance matrix are not enough for a correct estimation of the real data dispersion. On the other hand, for the value $h=1$, the outliers are not treated at all. It has to be noted, though, that the parameter h is data dependent. If an estimation of the outliers is a priori known, h can be initialized accordingly. The results shown in Fig. 7 are the result of 50 runs as described before where 5 different values of h were used in each run. The results are the

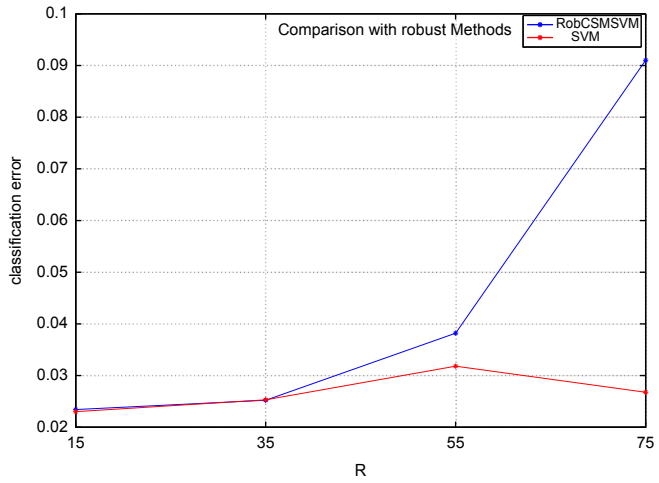

Fig. 6. Comparison with methods reported in [16]. The *Y*-axis represents classification error, while the *X*-axis the radius of the uniform noise.

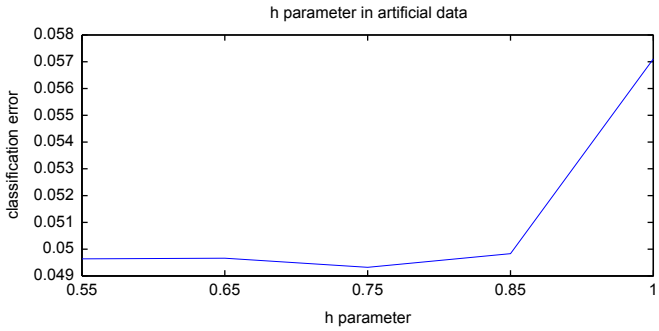

Fig. 7. Influence of parameter *h*.The *Y*-axis represents classification error, while the *X*-axis the values of parameter *h*.

mean values of the accuracies over 50 runs for each value of the parameter h.

5.2. Experiments with standard classification databases

For these experiments, we have used the publicly available datasets from the machine learning repository of University of California Irvine (UCI) [40]. More precisely, we have used the wine, statlog and iris datasets. For the wine and iris dataset 5-fold cross validation has been performed, while for the statlog dataset, there is an already provided training and testing dataset. The wine dataset consists of 178 samples of dimension 13 and 3 different classes of wines. The iris dataset has 150 samples of dimension 4 and 3 different classes. Finally, statlog is a set of satellite images with 6435 samples of dimension 36. From this dataset, 4435 samples are used for training and the rest 2000 are used for testing. Overall, 6 different classes are contained in the statlog dataset. Classification results from these datasets for different classifiers are shown in Table 2. We see that RobCSMVSVMs achieve the best classification accuracy.

5.3. Experiments on facial expression recognition

We have performed facial expression recognition on three publicly available facial expressions databases, namely, BU3DFE [37], Cohn–Kanade [36] and JAFFE [38] databases using previously mentioned SVM variants. BU3DFE consists of 100 subjects (56 male and 44 female) with six expressions, each at four different intensity levels. This results in a total of 2400 different facial expressions images and another 100 facial images with the neutral

Table 2
Classification accuracy on various classification databases for RobCSMVSVMs, RobMCVSVMs, CSMVSVMs, MCVSVMs and SVMs.

Method	Wine (%)	Iris (%)	Statlog (%)
RobCSMVSVMs	**97.19**	**98**	**85.25**
RobMCVSVMs	97	97.2	84.1
CSMVSVMs	97.17	96.66	84.1
MCVSVMs	96.6	96	81.65
SVMs	95.5	95.3	83.75

expression. In our experiments, we have used only the high level intensity expression (700 images). The Cohn–Kanade database is a video database of facial expressions. We have extracted 407 different facial expression images (with different number of entries in each facial expression class) from 100 different individuals. In this database, people vary in age, race and sex. Finally, the JAFFE database consists of 10 different Japanese females and a total of 213 facial images of all facial expressions.

For each database we have conducted a person-out based fivefold cross validation. More specifically, we exclude 20% of the individuals present in these databases and therefore we train our classifiers with the remaining 80% of the individuals and test with the excluded set. Moreover, based on the work in [41], we have used the so called enriched databases. They contain translated revisions of facial images that are appended in the training set, in order to enrich it. Experiments in [41], have proven that this approach does ameliorate the classification rates with respect to the original training datasets. Even in this case though, the person-out rule is followed. That is, transformed images of the excluded 20% individuals are also excluded from the training set in each fold of the cross-validation.

Due to the high data dimensionality (1200 dimensions for each image), we first perform PCA and retain 94% of the total energy for the enriched dataset. This results in a lower data dimensionality, since only 70 dimensions are retained. PCA, though, is very sensitive to outliers and, therefore, our robust SVMs cannot be compared in a fair way. For this reason, we use ROBPCA, a robust version of the PCA [34].

For the simple case, where only the original datasets are used, the performed principal components analysis must be more restricted. Only 80% of the total energy is retained, due to MCD calculation inside every class, since the number of samples therein should be at least twice the data dimension [24]. For instance, in the Cohn–Kanade database, the cardinality of the class "Anger" in a fivefold cross-validation process does not exceed 34, which limits the dimensionality to at most 17. This drawback cannot be attributed to the proposed method, but rather to the small size of the database. In real situations, at least for the training set, one should create a convenient training dataset.

Classification accuracies, for all the above mentioned experiments, are illustrated in Table 3. It is clear that in all cases, the RobCSMVSVM method outperforms all other ones. Compared with the other classifiers, we conclude that using robust covariance matrix calculation and a class specific approach, boosts the SVMs performance in facial expression recognition.

Moreover, for comparison purposes, Table 4 contains recognition accuracies comparison of the proposed method and the ones reported in [41]. As experiments were conducted on the same test sets, the comparison between results is fair, with slight differences, due to the randomness in the folds creation for the 5-fold cross validation process. In most cases, the proposed SVM method outperforms other subspace techniques with the only exception being the JAFFE database, where nearest centroid classification with PCA and LDA performs better.

Table 3
Person out 5-fold cross validation results for 3D facial expressions databases.

Method	Cohn–Kanade [36] (%)	BU3DFE [37] (%)	JAFFE [38] (%)
Robust CSMVSVMs	**69.20**	**67.28**	**60.18**
Robust MCVSVMs	67.45	65.43	58.80
CSMVSVMs	65.87	62.57	59.25
MCVSVMs [13]	63.40	61.40	58.80
SVMs	62.15	61.70	57.50

Table 4
Accuracy of various subspace techniques in facial expression recognition.

Method	Cohn–Kanade [36] (%)	BU3DFE [37] (%)	JAFFE [38] (%)
RobCSMVSVMs	**69.20**	**67.28**	60.18
NC PCA + LDA [41]	68.80	64.90	**63.50**
KNN PCA + LDA [41]	67.60	62.10	58.50

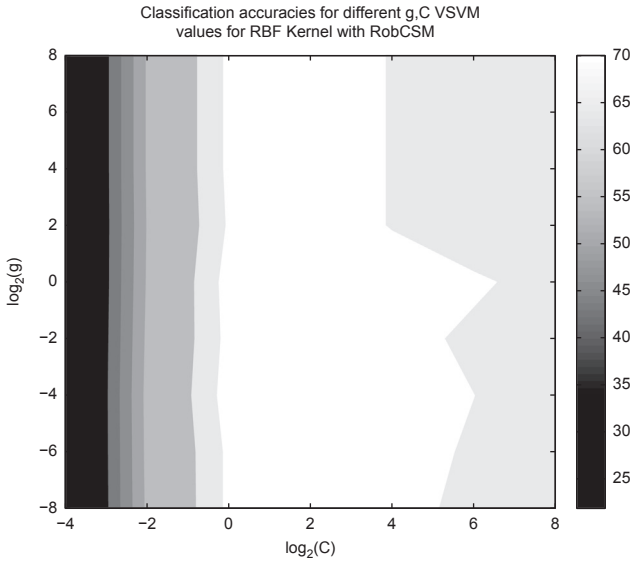

Fig. 8. Classification accuracies for different RBF kernel parameters for RobCSMSVM.

Finally, in order to test our method with various kernel configurations, we provide experiments in the Cohn–Kanade database [36]. The conducted experiments, perform a grid search for kernel parameters selection for RobCSMSVM and classical SVMs. In Figs. 8 and 9, the results of the different kernel parameters are shown for RBF kernels for RobCSMVSVM and SVM respectively. The classification accuracies versus the respective kernel parameters are shown with the different grey scales, with black been the lowest and white the highest accuracy. As can be seen in Figs. 10 and 11 in all cases our method outperforms simple kernel SVMs, since its best performance shown in Fig. 10 is in the range of 72%, whereas the best performance of classical SVM shown in Fig. 11 is in the range of 65%. Moreover, we have performed the same experiments for a polynomial kernel. Results are drawn in Figs. 10 and 11.

5.4. Statistical significance tests

We have used the McNemar test [42] to decide whether the difference between the proposed method and competing ones, is statistically significant. This method has been widely used for this purpose by several authors [43–45]. To do so, we constructed the

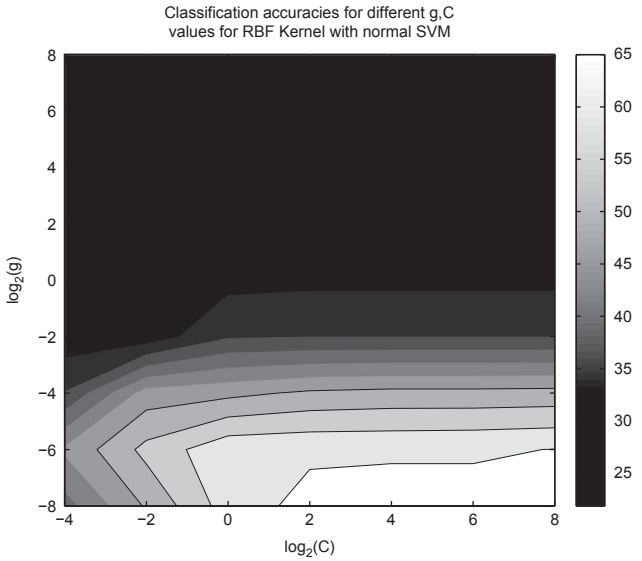

Fig. 9. Classification accuracies for different RBF kernel parameters for classical SVMs.

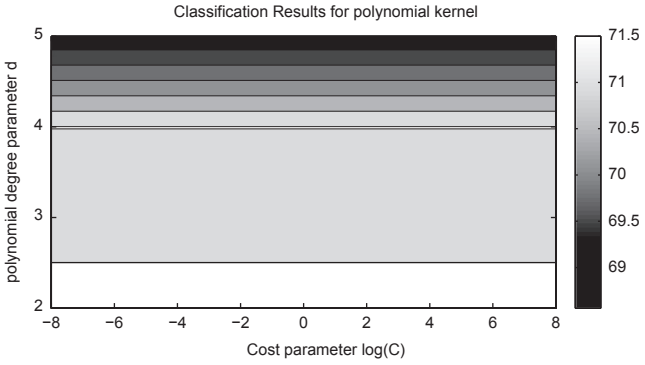

Fig. 10. Classification accuracies for different polynomial kernel parameters for RobCSMSVM.

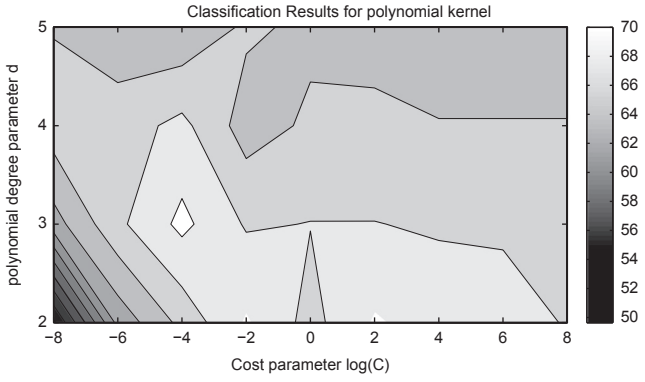

Fig. 11. Classification accuracies for different polynomial kernel parameters for classical SVMs.

contingency matrices for each classifier with the RobCSMVSVM classifier. The matrices are constructed based on the best facial expression recognition rate for each classifier. Table 5, shows the actual contingency tables from the conducted experiments on the BU3DFE database.

In all three cases the resulting p-value from McNemar test is less than the desired significance level (in our case p is set to $p \ll 0.02$). Therefore, it is safe to conclude that the difference of the classification accuracies is statistically significant.

Table 5
Contingency tables for all classifiers versus RobCSMVSVM for the BU3DFE database.

RobCSMVSVM(1st) vs. MCVSVM(2nd)	Misclassified by 1st	Classified by 1st
Misclassified by 2nd	493	30
Classified by 2nd	10	807
RobCSMVSVM(1st) vs. CSMVSVM(2nd)		
Misclassified by 2nd	500	25
Classified by 2nd	10	900
RobCSMVSVM(1st) vs. RobMCVSVM(2nd)		
Misclassified by 2nd	499	30
Classified by 2nd	10	901

6. Conclusions

A novel robust SVM framework has been proposed in this paper. The use of robust statistics in the calculation of the data dispersion provides better classification results in all tested applications (i.e., artificial data, UCI databases and facial expression recognition databases). Thorough benchmarking provided evidence that the proposed method performs better than previously published methods in various classification tasks. Moreover, class specific minimum variance SVM has been studied and analytically derived in the case of classical, as well as in the case of robust SVM framework. We also compared the proposed methods to other classification techniques, such as linear discriminant analysis and principal components analysis combined with nearest neighbor and nearest centroid classification. In most cases, our method outperforms the various competing techniques.

Conflict of Interest

None declared.

Acknowledgments

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant agreement no 248434 (MOBISERV).

Appendix A. Details of calculation of the Multiclass SVM problem

In order to solve Eq. (36) first we substitute (33), (34) to (29). For better readability of the equations we do the substitution in parts. That is, we partition Eq. (36) in two parts, which represent the two summation terms. For the first term we have the following:

$$\begin{aligned}\sum_{k=1}^{K}\mathbf{w}_k^T\mathbf{A}_k\mathbf{w}_k &= \frac{1}{4}\sum_{k=1}^{K}\left(\sum_{i=1}^{N}(c_{k,i}R_i-\alpha_{k,i})\mathbf{A}_k^{-1}\mathbf{x}_i\right)^T \\ &\quad\cdot\mathbf{A}_k\cdot\left(\sum_{i=1}^{N}(c_{k,j}R_j-\alpha_{k,j})\mathbf{A}_k^{-1}\mathbf{x}_j\right) \\ &= \frac{1}{4}\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(c_{k,i}R_i-\alpha_{k,i})\cdot(c_{k,j}R_j-\alpha_{k,j})\cdot\cdot\mathbf{x}_i^T\mathbf{A}_k^{-1}\mathbf{x}_j \\ &= \frac{1}{4}\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(c_{k,i}c_{k,j}R_iR_j-c_{k,i}R_i\alpha_{k,j}\end{aligned}$$

$$-c_{k,j}R_j\alpha_{k,i}+\alpha_{k,i}\alpha_{k,j})\mathbf{x}_i^T\mathbf{A}_k^{-1}\mathbf{x}_j. \quad (A.1)$$

It can be easily proven that

$$\sum_{k=1}^{K} c_{k,i}c_{k,j}=c_{l_i,j}=c_{l_j,i} \quad \text{and} \quad (A.2)$$

$$\sum_{k=1}^{K} c_{k,i}R_i\alpha_{k,j}=\sum_{k=1}^{K} c_{k,j}R_j\alpha_{k,i}, \quad (A.3)$$

from (A.2) and (A.3), Eq. (A.1) takes on its final form:

$$\sum_{k=1}^{K}\mathbf{w}_k^T\mathbf{A}_k\mathbf{w}_k=\frac{1}{4}\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(c_{l_i,j}R_iR_j-2c_{k,i}R_i\alpha_{k,j}+\alpha_{k,i}\alpha_{k,j})\mathbf{x}_i^T\mathbf{A}_k^{-1}\mathbf{x}_j, \quad (A.4)$$

for the second term of Eq. (36) we need the second KKT condition, by which the term on b_k and b_{l_i} annulate with the Lagrange multipliers. That is

$$\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}b_{l_i}=\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}b_k. \quad (A.5)$$

Thus, from (A.5) we have for the second term of (36)

$$\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}\mathbf{w}_{l_i}^T\mathbf{x}_i-\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}\mathbf{w}_k^T\mathbf{x}_i, \quad (A.6)$$

substituting (33) into (A.5) we have

$$\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}\cdot\frac{1}{2}\left(\mathbf{A}_{l_i}^{-1}\sum_{j=1}^{N}(c_{l_i,j}R_j-a_{l_i,j})\mathbf{x}_j\right)^T\mathbf{x}_i-\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}\frac{1}{2}\left(\mathbf{A}_k^{-1}\sum_{j=1}^{N}(c_{k,j}R_j-a_{k,j})\mathbf{x}_j\right)^T\mathbf{x}_i. \quad (A.7)$$

Once again for better readability of the equations we break down into its terms Eq. (A.7). For the first part we have

$$\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}\left(\mathbf{A}_{l_i}^{-1}\sum_{j=1}^{N}(c_{l_i,j}R_j-\alpha_{l_i,j})\mathbf{x}_j\right)^T\mathbf{x}_i=\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}\left(\sum_{j=1}^{N}(c_{l_i,j}R_j-\alpha_{l_i,j})\mathbf{x}_j\right)^T a_{k,i}\mathbf{A}_{l_i}^{-1}\mathbf{x}_i. \quad (A.8)$$

But it can be proven that

$$\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}\mathbf{A}_{l_i}^{-1}=\sum_{k=1}^{K}\mathbf{A}_k^{-1}\sum_{i=1}^{N}c_{k,i}R_i. \quad (A.9)$$

Thus (A.7) takes on its final form due to (A.9)

$$=\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}\left(\sum_{j=1}^{N}(c_{l_i,j}R_j-\alpha_{l_i,j})\mathbf{x}_j^T\right)\mathbf{A}_k^{-1}c_{k,i}R_i\mathbf{x}_i=\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(c_{l_i,j}c_{k,i}R_iR_j-c_{k,i}R_i\alpha_{l_i,j})\mathbf{x}_j^T\mathbf{A}_k^{-1}\mathbf{x}_i. \quad (A.10)$$

Equivalently, the second part of Eq. (A.7) takes on the form:

$$=-\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}(c_{k,j}R_j\alpha_{k,i}-\alpha_{k,j}\alpha_{k,i})\mathbf{x}_j^T\mathbf{A}_k^{-1}\mathbf{x}_i. \quad (A.11)$$

Adding all together we have the final form of (36)

$$\mathcal{L}(\mathbf{A}_k,\mathbf{w}_k,b_k,\boldsymbol{\alpha}_k)=2\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_{k,i}+\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\frac{1}{4}c_{l_i,j}R_iR_j-\frac{1}{2}\alpha_{k,i}\alpha_{l_i,j}+\frac{1}{4}\alpha_{k,i}\alpha_{k,j}\right)\cdot\mathbf{x}_i^T\mathbf{A}_k^{-1}\mathbf{x}_j. \quad (A.12)$$

References

[1] P. Shivaswamy, T. Jebara, Maximum relative margin and data-dependent regularization, The Journal of Machine Learning Research 11 (2010) 747–788.

[2] M. Adankon, M. Cheriet, Model selection for the LS-SVM application to handwriting recognition, Pattern Recognition 42 (12) (2009) 3264–3270.

[3] N. Khan, R. Ksantini, I. Ahmad, B. Boufama, A novel SVM+NDA model for classification with an application to face recognition, Pattern Recognition 45 (1) (2012) 66–79.

[4] X. Wang, L. SHU-XIA, J. Zhai, Fast fuzzy multicategory SVM based on support vector domain description, International Journal of Pattern Recognition and Artificial Intelligence 22 (01) (2008) 109–120.

[5] Y. Wu, Y. Lee, J. Yang, Robust and efficient multiclass SVM models for phrase pattern recognition, Pattern Recognition 41 (9) (2008) 2874–2889.

[6] R. Liu, Y. Wang, T. Baba, D. Masumoto, S. Nagata, SVM-based active feedback in image retrieval using clustering and unlabeled data, Pattern Recognition 41 (8) (2008) 2645–2655.

[7] Z. Xue, D. Ming, W. Song, B. Wan, S. Jin, Infrared gait recognition based on wavelet transform and support vector machine, Pattern Recognition 43 (8) (2010) 2904–2910.

[8] M. Tahir, A. Khan, A. Majid, Protein subcellular localization of fluorescence imagery using spatial and transform domain features, Bioinformatics 28 (1) (2012) 91–97.

[9] A. Khan, S.F. Tahir, A. Majid, T.-S. Choi, Machine learning based adaptive watermark decoding in view of anticipated attack, Pattern Recognition 41 (8) (2008) 2594–2610.

[10] X. Wang, T. Wang, J. Bu, Color image segmentation using pixel wise support vector machine classification, Pattern Recognition 44 (4) (2011) 777–787.

[11] R. Williamson, A. Smola, B. Scholkopf, Generalization performance of regularization networks and support vector machines via entropy numbers of compact operators, IEEE Transactions on Information Theory 47 (6) (2002) 2516–2532.

[12] K. Huang, H. Yang, I. King, M. Lyu, Maxi–Min margin machine: learning large margin classifiers locally and globally, IEEE Transactions on Neural Networks 19 (2) (2008) 260–272.

[13] A. Tefas, C. Kotropoulos, 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 23 (7) (2002) 735–746.

[14] S. Zafeiriou, A. Tefas, I. Pitas, Minimum class variance support vector machines, IEEE Transactions on Image Processing 16 (10) (2007) 2551–2564.

[15] I. Kotsia, I. Pitas, Facial expression recognition in image sequences using geometric deformation features and support vector machines, IEEE Transactions on Image Processing 16 (1) (2006) 172–187.

[16] L. Xu, K. Crammer, D. Schuurmans, Robust support vector machine training via convex outlier ablation, in: Proceedings of the National Conference on Artificial Intelligence, vol. 21, Menlo Park, CA; Cambridge, MA; London, AAAI Press; MIT Press, 1999, 2006, p. 536.

[17] I. Tsang, J. Kwok, S. Li, Learning the kernel in mahalanobis one-class support vector machines, in: IEEE International Joint Conference on Neural Networks, IJCNN'06. 2006, pp. 1169–1175.

[18] R. Wilcox, Introduction to Robust Estimation and Hypothesis Testing, Academic Press, 2005.

[19] A. Azzalini, M. Genton, Robust likelihood methods based on the skew-t and related distributions, International Statistical Review 76 (1) (2008) 106–129.

[20] M. Hubert, M. Debruyne, Minimum covariance determinant, Wiley Interdisciplinary Reviews: Computational Statistics 2 (1) (2010) 36–43.

[21] M. Hubert, P. Rousseeuw, K. Vanden Branden, ROBPCA: a new approach to robust principal component analysis, Technometrics 47 (1) (2005) 64–79.

[22] P. Rousseeuw, A. Leroy, Robust Regression and Outlier Detection, John Wiley & Sons Inc, 1987.

[23] C. Croux, G. Haesbroeck, Influence function and efficiency of the minimum covariance determinant scatter matrix estimator, Journal of Multivariate Analysis 71 (2) (1999) 161–190.

[24] P. Rousseeuw, K. Van Driessen, A fast algorithm for the minimum covariance determinant estimator, Technometrics 41 (3) (1999) 212–223.

[25] M. Schyns, G. Haesbroeck, F. Critchley, Relaxmcd: smooth optimisation for the minimum covariance determinant estimator, Computational Statistics & Data Analysis 54 (4) (2010) 843–857.

[26] T. Bernholt, P. Fischer, The complexity of computing the MCD-estimator, Theoretical Computer Science 326 (1–3) (2004) 383–398.

[27] F. Yang, Z. Shan, F. Kruggel, White matter lesion segmentation based on feature joint occurrence probability and $\chi 2$ random field theory from magnetic resonance (mr) images, Pattern Recognition Letters 31 (9) (2010) 781–790.

[28] Y. Zhan, J. Yin, Robust local tangent space alignment via iterative weighted PCA, Neurocomputing 74 (11) (2011) 1985–1993.

[29] J. Jin, J. An, Robust discriminant analysis and its application to identify protein coding regions of rice genes, Mathematical Biosciences 232 (2) (2011) 96–100.

[30] V. Vapnik, The Nature of Statistical Learning Theory, Springer Verlag, 2000.

[31] G. McLachlan, J. Wiley, Discriminant Analysis and Statistical Pattern Recognition, Wiley Online Library, 1992.

[32] R. Fletcher, Practical Methods of Optimization, John Wiley & Sons Inc, 1987.

[33] M. Girolami, Mercer kernel-based clustering in feature space, IEEE Transactions on Neural Networks 13 (3) (2002) 780–784.

[34] M. Hubert, P. Rousseeuw, S. Verboven, A fast method for robust principal components with applications to chemometrics, Chemometrics and Intelligent Laboratory Systems 60 (1–2) (2002) 101–111.

[35] C. Hsu, C. Lin, A comparison of methods for multiclass support vector machines, IEEE Transactions on Neural Networks 13 (2) (2002) 415–425.

[36] T. Kanade, J. Cohn, Y.-L. Tian, Comprehensive database for facial expression analysis, in: Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), 2000, pp. 46–53.
[37] L. Yin, X. Wei, Y. Sun, J. Wang, M. Rosato, A 3D facial expression database for facial behavior research, in: 7th International Conference on Automatic Face and Gesture Recognition, 2006, FGR 2006, IEEE, 2006, pp. 211–216.
[38] M. Lyons, S. Akamatsu, M. Kamachi, J. Gyoba, Coding facial expressions with gabor wavelets, in: Third IEEE International Conference on Automatic Face and Gesture Recognition, IEEE, 2002, pp. 200–205.
[39] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, software available at ⟨http://www.csie.ntu.edu.tw/∼cjlin/libsvm⟩, 2001.
[40] A. Frank, A. Asuncion, UCI machine learning repository URL ⟨http://archive.ics.uci.edu/ml⟩, 2010.
[41] D. Bolis, A. Maronidis, A. Tefas, I. Pitas, Improving the Robustness of Subspace Learning Techniques for Facial Expression Recognition, in: Artificial Neural Networks–ICANN 2010, Springer, 2010, pp. 470–479.
[42] B. Lindgren, Statistical Theory, Chapman & Hall/CRC, 1993.
[43] J. Yang, A. Frangi, J. Yang, D. Zhang, Z. Jin, KPCA plus LDA: a complete kernel Fisher discriminant framework for feature extraction and recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence (2005) 230–244.
[44] B. Draper, W. Yambor, J. Beveridge, Analyzing PCA-based face recognition algorithms: eigenvector selection and distance measures, empirical evaluation methods in computer vision.
[45] B. Draper, K. Baek, M. Bartlett, J. Beveridge, Recognizing faces with PCA and ICA, Computer Vision and Image Understanding 91 (1–2) (2003) 115–137.

Vretos Nicholas graduated from the Department of Informatics of The University Pierre et Marie Curie in Paris (Paris VI) in 2002. He is currently a researcher and developer at the Department of Informatics, in the Artificial Intelligence Information Analysis (AIIA) laboratory, at the Aristotle University of Thessaloniki. He has published more than 15 Journals, conference papers and Book Chapters. His research interests include digital signal processing, face detection/recognition, object tracking, image and video semantic content analysis, 3D Face Recognition, 3D Facial Expressions Recognition and Video Fingerprinting.

Anastasios Tefas received the B.Sc. in informatics in 1997 and the Ph.D. degree in informatics in 2002, both from the Aristotle University of Thessaloniki, Greece. Since 2008, he has been a Lecturer at the Department of Informatics, Aristotle University of Thessaloniki. From 2006 to 2008, he was an Assistant Professor at the Department of Information Management, Technological Institute of Kavala. From 2003 to 2004, he was a temporary lecturer in the Department of Informatics, University of Thessaloniki. From 1997 to 2002, he was a researcher and teaching assistant in the Department of Informatics, University of Thessaloniki. Dr. Tefas participated in 10 research projects financed by national and European funds. He has co-authored 22 journal papers, 77 papers in international conferences and contributed 7 chapters to edited books in his area of expertise. Over 1200 citations have been recorded to his publications and his H-index is 19 according to Google scholar. His current research interests include computational intelligence, pattern recognition, statistical machine learning, digital signal and image processing and computer vision.

Ioannis Pitas received the Diploma of Electrical Engineering in 1980 and the Ph.D. 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 or Visiting Assistant Professor at several Universities. He has published over 607 papers and contributed in 27 books in his areas of interest and edited or co-authored another 7. 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 or co- Editor of four international journals and General or Technical Chair of three international conferences. His current interests are in the areas of digital image and video processing and analysis, multidimensional signal processing, watermarking and computer vision.