Class-specific Reference Discriminant Analysis with application in Human Behaviour Analysis

Alexandros Iosifidis, Anastasios Tefas, and Ioannis Pitas

Abstract—**In this paper, a novel nonlinear subspace learning technique for class-specific data representation is proposed. A novel data representation is obtained by applying nonlinear class-specific data projection to a discriminant feature space, where the data belonging to the class under consideration are enforced to be close to their class representation, while the data belonging to the remaining classes are enforced to be as far as possible from it. A class is represented by an optimized class vector, enhancing class discrimination in the resulting feature space. An iterative optimization scheme is proposed to this end, where both the optimal nonlinear data projection and the optimal class representation are determined in each optimization step. The proposed approach is tested on three problems relating to human behaviour analysis: face recognition, facial expression recognition and human action recognition. Experimental results denote the effectiveness of the proposed approach, since the proposed Class-specific Reference Discriminant Analysis outperforms Kernel Discriminant Analysis, Kernel Spectral Regression and Class-specific Kernel Discriminant Analysis, as well as Support Vector Machine-based classification, in most cases.**

Index Terms—**Class-Specific Kernel Discriminant Analysis, Class-Specific Kernel Spectral Regression, Optimized Class Representation, Human-Computer Interaction.**

I. Introduction

Human behaviour analysis based on computer vision techniques is an active research field, due to its importance in a wide range of applications, including human-computer interaction [1] and assisted living [2]. It has attracted the attention of the research community for more than two decades and it is among the most popular research areas in computer vision and pattern recognition. Three of the basic problems in this area are the identification of persons based on their facial characteristics, the recognition of peoples' emotional state based on their facial expressions and the recognition of human activities.

Subspace learning techniques have been successfully employed in all of the three aforementioned problems, e.g., in [3], [4], [5], [6]. In such techniques, samples (e.g., facial images, or action videos) are represented by feature vectors and the objective is the determination of an optimal data projection that optimizes some criterion defined over training feature vectors that enhances the discrimination of various classes. Then, the input (high-dimensional in most cases) feature space is mapped to a low-dimensional feature space of increased discrimination power, where classification is usually performed based on simple similarity criteria, like the minimal Euclidean distance from the class mean vectors.

Criteria that have been employed for optimal subspace determination can be divided in two categories: generative and discriminative ones. Criteria belonging to the first category try to determine subspaces that best express the population of the available samples, without exploiting the labeling information that may be available for the training data. For example, Principal Component Analysis (PCA) [7] aims at the determination of an optimal subspace of maximal data dispersion, Independent Component Analysis (ICA) [8], [9] aims at the determination of statistically independent data projections, while Non-negative Matrix Factorization (NMF) [10] aims at the determination of data projections preserving the non-negative nature of samples (e.g., facial images). Such techniques have been widely adopted for their ability to reveal properties of interest appearing in the data. However, since they do not take into account any labeling information, their discriminative ability is restricted. Criteria exploiting labeling information of training data usually outperform the ones belonging to the first category in classification problems, since the objective in this case is the discrimination of various classes [11]. Linear Discriminant Analysis (LDA) and its variants [7], [3], [12] are probably the most widely adopted discriminant subspace learning techniques, due to their effectiveness in a wide range of classification problems. LDA aims at the determination of an optimal subspace in which samples belonging to different classes are as far from one another and that the within class dispersion from their mean is as small as possible. For the cases where linear projections are not appropriate for class discrimination, kernel extensions have also be proposed [3], [12], [4].

Standard Discriminant Learning techniques, like LDA [13], [7], Kernel Discriminant Analysis (KDA) [3], (kernel) Spectral Regression (KSR) [12] and Class-specific (kernel) Discriminant Analysis (CSKDA) [4], represent classes by adopting the corresponding class mean vectors. Thus, they inherently set the assumption that the classes forming the classification problem follow unimodal normal distributions having the same covariance structure [7]. However, these are two strong assumptions that are difficult to be met in real classification problems. When these assumptions are not met, the adoption of optimized class representations, other than the class mean vectors, leads to the determination of a discriminant subspace of increased class discrimination power [14], [15], [16]. In this paper, we follow this line of work and propose an optimization scheme for the determination of such an optimized class representation for class-specific nonlinear data projection that leads to the determination of a discriminant subspace having increased class discrimination power.

We propose a novel class-specific discrimination criterion

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

which is used to optimize both the data projections and the class representation for the determination of a low-dimensional feature space of increased discrimination power. We apply the proposed criterion in three problems relating to human behaviour analysis: the recognition of human face, facial expression and activity. Since kernel methods have been found to outperform linear ones in these classification problems [3], [17], [18], we formulate our class-specific criterion to exploit data representations in arbitrary-dimensional Hilbert spaces for nonlinear data projection and classification [19], [20], [21], [22]. We propose two iterative optimization schemes to this end: the first one, referred to as direct optimization scheme hereafter, optimizes the original criterion with respect to both the data projection matrix and the class representation, referred to as class reference vector hereafter. The second one solves an approximation of the original criterion for the determination of the data projection matrix. For the latter case, we introduce the proposed class-specific criterion in the Spectral Regression framework [12] in order to obtain a faster optimization method, compared to the direct case. We compare the performance of the proposed Class-specific Reference Discriminant Analysis (CSRDA) algorithm with that of other Discriminant Analysis-based classification schemes, i.e., KDA, KSR and CSKDA, as well as with the performance of the Kernel Support Vector Machine (KSVM) classifier, which is one of standard choices in nonlinear classification problems. Experimental results on nine publicly available datasets (ORL [23], AR [11] and Extended YALE-B [24] for face recognition, COHN-Kanade [25], JAFFE [26] and BU [27] for facial expression recognition and Hollywood2 [28], Olympic Sports [29] and ASLAN [30] for human action recognition) demonstrate the effectiveness of the proposed approach.

The contributions of the paper are:

- A novel class-specific discrimination criterion is proposed, which aims at the determination of both the optimal data projection and the optimal class representation, that will be subsequently used for classification.
- An optimization scheme that solves an approximation of the original criterion for data projections determination, leading to faster optimization. Experimental results also denote that this variant is able to outperform the original criterion in most cases.
- Evaluation of the proposed method using nine publicly available databases, where its effectiveness is proven by comparing its performance with that of related classification methods.

II. Class-specific classification

Let us denote by $\mathcal{U}$ an annotated database containing N samples (facial images or action videos), each belonging to one or multiple classes forming a class set $\mathcal{C} = \{1, \dots, C\}$. In the case of facial images each sample is accompanied by a label denoting the depicted person/expression. In the case of human actions, each video may depict one or multiple actions (e.g., an action video may depict a person running in a scene where several bypassers are walking). Let us also assume that each of the N samples in $\mathcal{U}$ has been pre-processed, in order to be represented by a vector $\mathbf{x}_i \in \mathbb{R}^D$, $i = 1, \dots, N$. In the case of facial images, $\mathbf{x}_i$ is obtained by vectorizing the facial image i, while in the case of human actions, $\mathbf{x}_i$ is a vector representation (in our experiments we have employed the Bag of Words (BoW)-based one, as in Section V) of action video i. The presense/absence of each (ID/expression/action) class j in each sample i can be stored in C binary label vectors $\mathbf{c}_j \in \mathbb{R}^N$, $j = 1, \dots, C$ whose elements are set equal to $c_{ji} = 1$ in the case where sample i belongs to class j and to $c_{ji} = 0$, otherwise. Let us denote by N_{j0}, N_{j1} the number of zero and non-zero elements in $\mathbf{c}_j$, respectively. By using $\mathbf{x}_i$ and $\mathbf{c}_j$, C binary classifiers can be learned, each discriminating class j from the remaining ones, in an one-versus-rest manner.

We approach this problem by learning C discriminant subspaces $\mathbb{R}^{d_j}, j = 1, \dots, C$, where d_j denotes the dimensionality of the resulting feature space for the class-specific classification problem discriminating class j from the remaining ones. The dimensionality d_j of the projection spaces obtained for different classes may vary. That is, in the case where the class under consideration is well distinguished from the remaining ones, class discrimination may require the use of few dimensions, while in the case where a class is confused with some others, a higher number of projection dimensions may be required to achieve adequate class discrimination. This is a limitation of traditional multi-class discriminant subspace learning methods [7], [3], [12], since by applying such methods only one subspace is learned which is shared among the classes and its dimensionality is limited. For example, in LDA the dimensionality of the resulting discriminant subspace is at most equal to $C - 1$. Class-specific methods are able to overcome this limitation, since the dimensionality of the resulting spaces can be proportional to the number of the training data belonging to the class under consideration, i.e., at most equal to N_{j1}. For example, the maximal discriminant space dimensionality obtained by applying Class-specific LDA [31] is $N_{j1} - 1$, i.e., equal to the rank of the within-class scatter matrix employed in its optimization process.

In order to exploit kernel techniques for nonlinear data projection and classification, the input space $\mathbb{R}^D$ is mapped to an arbitrary-dimensional feature space $\mathcal{F}$ (usually having the properties of Hilbert spaces [32], [19], [20], [21], [22]) by employing a function $\phi(\cdot) : \mathbf{x}_i \in \mathbb{R}^D \rightarrow \phi(\mathbf{x}_i) \in \mathcal{F}$ determining a nonlinear mapping from the input space $\mathbb{R}^D$ to the arbitrary-dimensional space $\mathcal{F}$. In this space, we would like to determine a data projection matrix $\mathbf{W}$ that will be used to map a given sample $\mathbf{x}_i$ to a low-dimensional feature space $\mathbb{R}^{d_j}$ of increased discrimination power:

$$\mathbf{z}_i = \mathbf{W}^T \phi(\mathbf{x}_i), \quad \mathbf{z}_i \in \mathbb{R}^{d_j}. \tag{1}$$

In practice, since the multiplication in (1) can not be directly computed, the so-called *kernel trick* [19], [20] is adopted. That is, the multiplication in (1) is inherently computed by using dot-products in $\mathcal{F}$. After the determination of the low-dimensional space $\mathbb{R}^{d_j}$, a distance measure, usually the Euclidean one, can be used in order to determine if a vector representing a test sample belongs to class j or not.

III. Class-Specific Reference Discriminant Analysis

Let us denote by $\phi(\boldsymbol{\mu}_j) \in \mathcal{F}$ a so-called reference vector that will be used in order to represent class j. Usually, it is chosen to be the class mean vector in $\mathcal{F}$, i.e., $\phi(\boldsymbol{\mu}_j) = \frac{1}{N_{j1}} \sum_{i,c_{ji}=1} \phi(\mathbf{x}_i)$ [4]. In the proposed approach, we do not restrict $\phi(\boldsymbol{\mu}_j)$ to be the class mean vector in $\mathcal{F}$. $\phi(\boldsymbol{\mu}_j)$ can be any vector that enhances the discrimination of class j from the remaining ones in the discriminant space $\mathbb{R}^{d_j}$. An optimization process for the determination of such an optimized class vector is described in subsection III-C. As has been previously described, we would like to learn a data projection matrix $\mathbf{W}$ which maps $\mathcal{F}$ to a low-dimensional discriminant space $\mathbb{R}^{d_j}$ where the samples belonging to class j are as close as possible to the image of $\phi(\boldsymbol{\mu}_j)$ in $\mathbb{R}^{d_j}$, i.e., $\mathbf{z}_j = \mathbf{W}^T \phi(\boldsymbol{\mu}_j)$, while the samples belonging to the remaining action classes are as far as possible from it. That is, we would like to learn a projection matrix $\mathbf{W} \in \mathbb{R}^{|\mathcal{F}| \times d_j}$ minimizing:

$$D_j = \sum_{i,c_{ji}=1} \|\mathbf{W}^T \phi(\mathbf{x}_i) - \mathbf{W}^T \phi(\boldsymbol{\mu}_j)\|_2^2 \qquad (2)$$

and maximizing:

$$D_0 = \sum_{i,c_{ji}=0} \|\mathbf{W}^T \phi(\mathbf{x}_i) - \mathbf{W}^T \phi(\boldsymbol{\mu}_j)\|_2^2. \qquad (3)$$

$\mathbf{W}$ can be determined by solving the optimization problem (4), where $tr(\cdot)$ denotes the trace operator:

$$\begin{aligned} \mathcal{J}(\mathbf{W}) &= \frac{D_0}{D_j} = \frac{\sum_{i,c_{ji}=0} \|\mathbf{W}^T \phi(\mathbf{x}_i) - \mathbf{W}^T \phi(\boldsymbol{\mu}_j)\|_2^2}{\sum_{i,c_{ji}=1} \|\mathbf{W}^T \phi(\mathbf{x}_i) - \mathbf{W}^T \phi(\boldsymbol{\mu}_j)\|_2^2} \\ &= \frac{tr\left(\mathbf{W}^T \mathbf{S}_0 \mathbf{W}\right)}{tr\left(\mathbf{W}^T \mathbf{S}_j \mathbf{W}\right)}, \qquad (4) \end{aligned}$$

where $\mathbf{S}_j$, $\mathbf{S}_0$ are defined by:

$$\mathbf{S}_j = \sum_{i,c_{ji}=1} \Big(\phi(\mathbf{x}_i) - \phi(\boldsymbol{\mu}_j)\Big)\Big(\phi(\mathbf{x}_i) - \phi(\boldsymbol{\mu}_j)\Big)^T \qquad (5)$$

$$\mathbf{S}_0 = \sum_{i,c_{ji}=0} \Big(\phi(\mathbf{x}_i) - \phi(\boldsymbol{\mu}_j)\Big)\Big(\phi(\mathbf{x}_i) - \phi(\boldsymbol{\mu}_j)\Big)^T \qquad (6)$$

The direct maximization of (4) is intractable since $\mathbf{S}_j$, $\mathbf{S}_0$ express the intra-class and out-of-class variances of the training samples with respect to $\boldsymbol{\mu}_j$ in $\mathcal{F}$, respectively ($\mathbf{S}_j$, $\mathbf{S}_0$ are matrices of arbitrary dimensions). Next, we describe an optimization process that can be used to maximize (4) for the determination of the optimal data projection.

A. Direct Optimization of (4)

Let us denote by $\mathbf{\Phi}$, $\mathbf{\Phi}_j$ and $\mathbf{\Phi}_0$ three matrices containing the representations in $\mathcal{F}$ of the entire training set, the training data belonging to class j and to the remaining classes, respectively, i.e., $\mathbf{\Phi} = \{\phi(\mathbf{x}_i)\}_{i=1}^N$, $\mathbf{\Phi}_j = \{\phi(\mathbf{x}_i)\}_{i,c_{ji}=1}$ and $\mathbf{\Phi}_0 = \{\phi(\mathbf{x}_i)\}_{i,c_{ji}=0}$. The so-called kernel matrix, $\mathbf{K} \in \mathbb{R}^{N \times N}$, is given by $\mathbf{K} = \mathbf{\Phi}^T \mathbf{\Phi}$. In the following, we assume that the data set is centered in $\mathcal{F}$[1].

In order to directly optimize $\mathcal{J}$ in (4), we express the matrix $\mathbf{W}$ as a linear combination of the training data (represented in $\mathcal{F}$) [20], [19], [32], i.e.,:

$$\mathbf{W} = \sum_{i=1}^N \phi(\mathbf{x}_i) \boldsymbol{\alpha}_i^T = \mathbf{\Phi} \mathbf{A}. \qquad (7)$$

$\mathbf{A} \in \mathbb{R}^{N \times d_j}$ is a matrix containing the reconstruction weights of $\mathbf{W}$, with respect to the training data in $\mathcal{F}$. $\phi(\boldsymbol{\mu}_j)$ can also be expressed as a linear combination of the columns of $\mathbf{\Phi}_j$, i.e., $\phi(\boldsymbol{\mu}_j) = \mathbf{\Phi}_j \mathbf{b}_j$, where $\mathbf{b}_j \in \mathbb{R}^{N_{j1}}$ is a vector containing the reconstruction weights of $\phi(\boldsymbol{\mu}_j)$ with respect to $\mathbf{\Phi}_j$. An optimization process for the determination of $\mathbf{b}_j$ is described in subsection III-C.

As shown in Appendix I, the optimization problem in (4) can be transformed to the following equivalent optimization problem:

$$\mathcal{J}(\mathbf{A}) = \frac{tr\left(\mathbf{A}^T \mathbf{M}_0 \mathbf{A}\right)}{tr\left(\mathbf{A}^T \mathbf{M}_j \mathbf{A}\right)}, \qquad (8)$$

where the objective is the determination of the optimal reconstruction weights $\mathbf{A}$ maximizing $\mathcal{J}$. Since the matrices $\mathbf{Q}_0 = \mathbf{A}^T \mathbf{M}_0 \mathbf{A}$ and $\mathbf{Q}_j = \mathbf{A}^T \mathbf{M}_j \mathbf{A}$ are real ($\mathbf{Q}_0, \mathbf{Q}_j \in \mathbb{R}^{N \times N}$) and positive semi-definite, the optimization problem in (8) corresponds to the standard *trace ratio* optimization problem, which has been used by a number of dimensionality reduction algorithms [7], [3], [33], [34]. However, the trace ratio problem does not have a direct closed-form globally optimal solution [35], [36], [37], [13]. Therefore, it is conventionally approximated by solving the *ratio trace* problem, i.e., in our case $\tilde{\mathcal{J}}(\mathbf{A}) = Tr\left((\mathbf{A}^T \mathbf{M}_j \mathbf{A})^{-1} (\mathbf{A}^T \mathbf{M}_0 \mathbf{A})\right)$, which is equivalent to the optimization problem $\mathbf{M}_j \mathbf{v} = \lambda \mathbf{M}_0 \mathbf{v}$, $\lambda \neq 0$, and can be solved by performing eigenanalysis to the matrix $\mathbf{M} = \mathbf{M}_0^{-1} \mathbf{M}_j$ in the case where $\mathbf{M}_0$ is invertible, or $\mathbf{M} = \mathbf{M}_j^{-1} \mathbf{M}_0$ in the case where $\mathbf{M}_j$ is invertible.

Although the trace ratio problem does not have a closed form solution, [37] and [13] show that the original trace ratio problem can be converted to an equivalent *trace difference* problem having the form:

$$\tilde{\mathcal{J}}(\mathbf{A}, \lambda) = Tr\Big(\mathbf{A}^T \left(\mathbf{M}_0 - \lambda \mathbf{M}_j\right) \mathbf{A}\Big), \qquad (9)$$

where $\lambda \geq 0$ is the trace ratio value, i.e., $\lambda = \frac{tr\left(\mathbf{A}^T \mathbf{M}_0 \mathbf{A}\right)}{tr\left(\mathbf{A}^T \mathbf{M}_j \mathbf{A}\right)}$. The best trace ratio value λ^* can be calculated by applying an iterative optimization scheme based on the Newton-Raphson method. For more details on the λ^* calculation, please refer to [37], [13]. We have employed the method in [37], since it was found to be efficient. After obtaining λ^*, the optimal reconstruction weights matrix $\mathbf{A}^*$ is obtained by:

$$\mathbf{A}^* = \underset{\mathbf{A}^T \mathbf{A} = \mathbf{I}}{argmax}\ Tr\Big(\mathbf{A}^T \left(\mathbf{M}_0 - \lambda^* \mathbf{M}_j\right) \mathbf{A}\Big). \qquad (10)$$

That is, $\mathbf{A}^*$ is formed by the eigenvectors of $\mathbf{M} = \mathbf{M}_0 - \lambda^* \mathbf{M}_j$ corresponding to the d_j maximal eigenvalues.

After the determination of $\mathbf{A}$, a test sample $\mathbf{x}_t$ can be mapped to the discriminant space $\mathbb{R}^{d_j}$ by applying:

$$\mathbf{z}_t = \mathbf{A}^T \mathbf{k}_t, \qquad (11)$$

where $\mathbf{k}_t \in \mathbb{R}^N$ is a vector having its elements equal to $\mathbf{k}_{t,i} = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_t)$.

[1] This can always be done by using $\tilde{\phi}(\mathbf{x}_i) = \phi(\mathbf{x}_i) - \phi(\mathbf{m})$, where $\mathbf{m} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$, leading to a centered version of the kernel matrix given by $\tilde{\mathbf{K}} = \frac{1}{N} \mathbf{K} \mathbf{1} - \frac{1}{N} \mathbf{1} \mathbf{K} + \frac{1}{N^2} \mathbf{1} \mathbf{K} \mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^{N \times N}$ is a matrix of ones.

B. Spectral Regression-based Optimization of (4)

As described in subsection III-A, the criterion $\mathcal{J}$ in (4) can be directly optimized by solving the trace difference problem in (9), which is performed by applying an iterative optimization scheme requiring the solution of an eigenanalysis problem at each optimization step. The time complexity of such an approach is equal to $O(KN^3)$, where K is the number of iterations required for convergence. Since this process is based on random initialization of $\mathbf{A}$ and (Newton-Raphson based) gradient ascent, the number of iterations required for convergence is usually high (approximately equal to $K = 10$ in our experiments). Thus, the above-described optimization process is time consuming, especially for large classification problems. We describe a method for the determination of the reconstruction weights matrix $\mathbf{A}$ based on the ratio trace problem by following a Spectral Regression approach [12].

Let us denote by $\mathbf{v}$ an eigenvector of the problem $\mathbf{S}_0\mathbf{v} = \lambda\mathbf{S}_j\mathbf{v}$ with eigenvalue λ. $\mathbf{v}$ can be expressed as a linear combination of the training data in $\mathcal{F}$, i.e., $\mathbf{v} = \sum_{i=1}^N \alpha_i\phi(\mathbf{x}_i)$. By setting $\mathbf{Ka} = \mathbf{q}$, this eigenanalysis problem can be transformed to the following equivalent problem:

$$\mathbf{P}_0\mathbf{q} = \lambda\mathbf{P}_j\mathbf{q}. \tag{12}$$

The derivation of (12) is described in Appendix II. Thus, the reconstruction weights matrix $\mathbf{A}$ can be performed by applying a two step procedure:

- Solution of the eigenproblem $\mathbf{P}_0\mathbf{q} = \lambda\mathbf{P}_j\mathbf{q}$, which is tractable since $\mathbf{P}_0, \mathbf{P}_j \in \mathbb{R}^{N\times N}$. The solution of this problem leads to the determination of a matrix $\mathbf{Q} = [\mathbf{q}_1, \dots, \mathbf{q}_{d_j}]$, where $\mathbf{q}_i$ is the eigenvector corresponding to the i-th largest eigenvalue.
- Determination of the matrix $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_{d_j}]$, where $\mathbf{Ka}_i = \mathbf{q}_i$. In the case where $\mathbf{K}$ is non-singular, the vectors $\mathbf{a}_i$ are given by $\mathbf{a}_i = \mathbf{K}^{-1}\mathbf{q}_i$. When this is not true, the vectors $\mathbf{a}_i$ can be obtained by solving the following set of linear equations:

$$\left(\mathbf{K} + \delta\mathbf{I}\right)\mathbf{a}_i = \mathbf{q}_i. \tag{13}$$

  where $\delta \geq 0$ is a regularization parameter. Thus, $\mathbf{a}_i$ is given by $\mathbf{a}_i = \left(\mathbf{K} + \delta\mathbf{I}\right)^{-1}\mathbf{q}_i$.

The above-described optimization process requires the solution of one eigenanalysis problem (12) and the inversion of a $N\times N$ matrix, leading to a time complexity equal to $O(2N^3)$.

After the calculation of $\mathbf{A}$, a test sample $\mathbf{x}_t$ can be mapped to the discriminant space $\mathbb{R}^{d_j}$ by using (11).

C. Reference Class Vector Determination

As described, class j is represented by the reference vector $\phi(\boldsymbol{\mu}_j)$, which is not restricted to be the class mean vector in $\mathcal{F}$. Here, we describe a process for the determination of an optimized class representation $\phi(\boldsymbol{\mu}_j)$ that maximizes discrimination of class j in the discriminant space $\mathbb{R}^{d_j}$.

By observing that $\mathbf{S}_j$, $\mathbf{S}_0$ are functions of $\phi(\boldsymbol{\mu}_j)$, as detailed in (4), and by using $\phi(\boldsymbol{\mu}_j) = \boldsymbol{\Phi}_j\mathbf{b}_j$, $\phi(\boldsymbol{\mu}_j)$ can be determined by maximizing $\mathcal{J}$ with respect to $\mathbf{b}_j$, i.e.,:

$$\mathbf{b}_j^* = \arg\max_{\mathbf{b}_j} \mathcal{J}(\mathbf{W}, \mathbf{b}_j). \tag{14}$$

By using $\mathbf{b}_j$, the equivalent to $\mathcal{J}$ trace difference optimization problem can be written in the form:

$$\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{b}_j) = tr\left(\mathbf{W}^T\Big(\mathbf{S}_0(\mathbf{b}_j) - \lambda\mathbf{S}_j(\mathbf{b}_j)\Big)\mathbf{W}\right). \tag{15}$$

By solving for $\nabla_{\mathbf{b}_i}\left(\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{b}_j)\right) = 0$, we obtain:

$$\mathbf{b}_j^* = \frac{\gamma}{N_{j1}}\mathbf{1}_{N_{j1}}. \tag{16}$$

The derivation of γ is described in Appendix III.

D. Optimization with respect to both $\mathbf{A}$ and $\mathbf{b}_j$

Since $\mathcal{J}$ is a function of both the reconstruction weights matrix $\mathbf{A}$ and the class representation $\mathbf{b}_j$, we would like to determine a combination $\{\mathbf{A}, \mathbf{b}_j\}$ maximizing $\mathcal{J}$. Taking into account that $\mathbf{A}$ is a function of $\mathbf{b}_j$ and that $\mathbf{b}_j$ is a function of $\mathbf{A}$, a direct maximization of $\mathcal{J}$ with respect to both $\mathbf{A}$ and $\mathbf{b}_j$ is difficult. In order to maximize $\mathcal{J}$ with respect to both $\mathbf{A}$ and $\mathbf{b}_j$, we employ an iterative optimization scheme. In the following, we introduce an index t denoting the iteration of this iterative optimization scheme.

Let us denote by $\mathbf{b}_{j,t}$ the reference class vector that is calculated at the t-th iteration of the proposed optimization scheme. By using $\mathbf{b}_{j,t}$, the data projection matrix $\mathbf{A}_t$ can be calculated by following one of the processes described in subsections III-A and III-B. After the determination of $\mathbf{A}_t$, $\mathbf{b}_{j,t+1}$ can be calculated by using (16). The above described process is initialized by using the class mean vector, i.e., $\mathbf{b}_{j,0} = \frac{1}{N_{j1}}\mathbf{1}_{N_{j1}}$ and is terminated when $(\mathcal{J}(t+1) - \mathcal{J}(t))/\mathcal{J}(t) < \epsilon$, where ϵ is a small positive value (equal to $\epsilon = 10^{-6}$ in our experiments).

By following the above-described optimization process, the proposed CSRDA is able to determine a combination of $\{\mathbf{A}, \mathbf{b}_j\}$ maximizing $\mathcal{J}$ with respect to both the data projection matrix $\mathbf{A}$ and the reference class vector $\mathbf{b}_j$. Since the criterion $\mathcal{J}$ is used in order to measure the discrimination of class j with respect to the remaining classes, the proposed CSRDA method enhances class discrimination, when compared to CSKDA. Assuming that both training and test data come from the same distribution, we expect that the higher the class discrimination is, the better the generalization performance of the method will be.

E. Classification process (Test phase)

After the determination of the discriminant space $\mathbb{R}^{d_j}$, both the training data $\mathbf{x}_i$, $i = 1, \dots, N$ and the reference class vector $\phi(\boldsymbol{\mu}_j)$ are mapped to that space by using (11) and $\mathbf{z}_i$, $i = 1, \dots, N$, $\mathbf{z}_j$ are obtained. Subsequently, we can train a classifier by choosing one of the following two options:

- Calculation of the similarity values s_i between the training data $\mathbf{z}_i$, $i = 1, \dots, N$ and the reference class vector $\mathbf{z}_j$ in $\mathbb{R}^{d_j}$, i.e.,:

$$s_i = \|\mathbf{z}_i - \mathbf{z}_j\|_2^{-1}. \tag{17}$$

  Based on these similarity values, a threshold value T_j is determined. T_j can be used to determine if a test sample, represented by a discriminant vector $\mathbf{z}_t \in \mathbb{R}^{d_j}$, belongs to class j, or not.

- Calculation of distance vectors $\mathbf{d}_i \in \mathbb{R}^{d_j}$ having elements equal to:

$$d_{ik} = |\mathbf{z}_{ik} - \mathbf{z}_{jk}|, \;\; k = 1, \ldots, d_j, \tag{18}$$

where $\mathbf{z}_{ik}$, $\mathbf{z}_{jk}$ are the k-th elements of $\mathbf{z}_i$ and $\mathbf{z}_j$, respectively. $|\cdot|$ denotes the absolute value operator. By using $\mathbf{d}_i$, classification can be performed based on a linear classifier, e.g., linear SVM.

In section V we report performance using the second option, which outperforms the first one in most cases.

In case of multi-class classification, we train C linear SVM classifiers in an one-versus-rest manner using the above described process. A test sample is introduced to all C classifiers and is assigned to the class providing the maximal probability, similar to [38], [17].

IV. Evaluation Methods

We evaluate the performance of the proposed CSRDA algorithm using nine publicly available datasets: ORL, AR and Extended YALE-B (face recognition), COHN-KANADE, BU and JAFFE (facial expression recognition) and Hollywood2, Olympic Sports and ASLAN (human action recognition). We compare the performance of the proposed CSRDA-based classification schemes with that of related multi-class classification schemes, i.e., KSR [12] and KDA [3] based data projection followed with nearest class centroid classification. In addition, we compare the performance of the proposed CSRDA-based classification schemes with that of CSKDA [4] and with that of the KSVM classifier [38], which is one of the standard choices in kernel-based classification. Finally, we also provide the performance of the Nearest Class Mean (NM), Nearest Neighbor (1NN) classification schemes.

For the experiments involving facial image classification we have employed the RBF kernel function:

$$\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = exp\Big(- \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2} \Big), \tag{19}$$

where the value of σ is set equal to the mean Euclidean distance between the (vectorized) training facial images $\mathbf{x}_i$.

In human action recognition, we use the methods proposed in [30], [39] as baseline approaches: on the ASLAN dataset we employ a set of 12 histogram similarity values expressing the similarity of pairs of videos represented by using the BoW model for Histogram of Oriented Gradients (HOG), Histogram of Optical Flow (HOF) and Histogram of oriented gradient and optical Flow (HNF) descriptors evaluated on Space-Time Interest Point (STIP) video locations [40]. This video pair similarity representation is employed for classification using a linear SVM classifier. We also employ linear kernel for the proposed CSRDA and the remaining Discriminant Analysis algorithms on this dadtaset. For the remaining datasets, we employ the BoW-based video representation by using HOG, HOF, Motion Boundary Histograms (MBHx, MBHy) and (normalized) Trajectory descriptors evaluated on the trajectories of densely sampled interest points [39]. Classification is performed by employing a kernel SVM classifier and the RBF-χ^2 kernel [41], where different descriptors are combined in a multi-channel approach [42]:

$$\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = exp\Big(- \sum_k \frac{1}{4A^k} D\left(\mathbf{x}_i^k, \mathbf{x}_j^k\right) \Big). \tag{20}$$

$D\left(\mathbf{x}_i^k, \mathbf{x}_j^k\right)$ is the χ^2 distance between the BoW-based video representation of $\mathbf{x}_i$ and $\mathbf{x}_j$, with respect to the k-th channel. A^k is the mean value of the χ^2 distances between the training data for the k-th channel. We also employ the χ^2 kernel for the proposed CSRDA and the remaining Discriminant Analysis algorithms.

As described in Section II, the dimensionality of the discriminant space determined by the proposed CSRDA algorithm can vary and, in general, can be different over the various classes forming the classification problem. Intuitively, this can also be explained by the fact that some classes may be harder to be discriminated from the remaining ones. In such cases, higher subspace dimensionality may enhance the discrimination of the class under consideration. However, due to the high number of possible combinations of discriminant space dimensionalities in the above described datasets, we considered only the case where all the class-specific discriminant spaces are of the same dimensionality, i.e., $d_j = d$, $j = 1, \ldots, C$. The value of d is determined by applying five-fold cross-validation using the values $d = \{1, 10, 100, 1000\}$ for the ASLAN dataset and $d = 1, \ldots, 25$ for the remaining datasets.

A. Face recognition datasets

1) The ORL dataset [23]: consists of 400 facial images depicting 40 persons (10 images each). The images were captured at different times and with different conditions, in terms of lighting, facial expressions (smiling/not smiling) and facial details (open/closed eyes, with/without glasses). Facial images were taken in frontal position with a tolerance for face rotation and tilting up to 20 degrees. Example images of the dataset are illustrated in Figure 1.

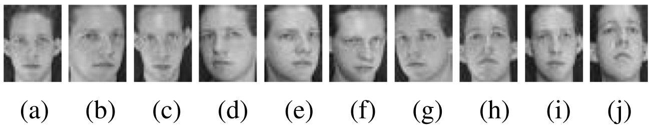
(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. 1. *Facial images depicting a person of the ORL dataset.*

2) The AR dataset [11]: consists of over 4000 facial images depicting 70 male and 56 female faces. In our experiments we have used the preprocessed (cropped) facial images provided by the database, depicting 100 persons (50 males and 50 females) having a frontal facial pose, performing several expressions (anger, smiling and screaming), in different illumination conditions (left and/or right light) and with some occlusions (sun glasses and scarf). Each person was recorded in two sessions, separated by two weeks. Example images of the dataset are illustrated in Figure 2.

3) The Extended YALE-B dataset [24]: consists of facial images depicting 38 persons in 9 poses, under 64 illumination conditions. In our experiments we have used the frontal

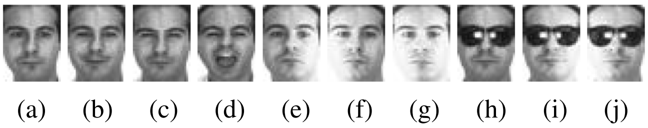

Fig. 2. *Facial images depicting a person of the AR dataset.*

cropped images provided by the database. Example images of the dataset are illustrated in Figure 3.

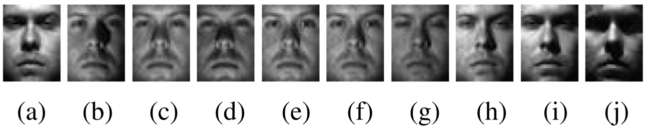

Fig. 3. *Facial images depicting a person of the Extended YALE-B dataset.*

B. Facial expression recognition datasets

1) The COHN-KANADE dataset [25]: consists of facial images depicting 210 persons of age between 18 and 50 (69% female, 31% male, 81% Euro-American, 13% Afro-American and 6% other groups). We have randomly selected 35 images for each facial expression, i.e., anger, disgust, fear, happiness, sadness, surprise and neutral. Example images of the dataset are illustrated in Figure 4.

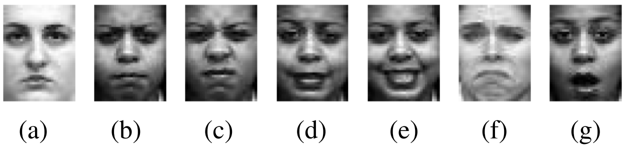

Fig. 4. *Facial images from the COHN-KANADE dataset: a) neutral, b) anger, c) disgust, d) fear, e) happy, f) sad and g) surprise.*

2) The BU dataset [27]: consists of facial images depicting over 100 persons (60% feamale and 40% male) with a variety of ethnic/racial background, including White, Black, East-Asian, Middle-east Asian, Hispanic Latino and others. All expressions, except the neutral one, are expressed at four intensity levels. In our experiments, we have employed the images depicting the most expressive intensity of each facial expression. Example images of the dataset are illustrated in Figure 5.

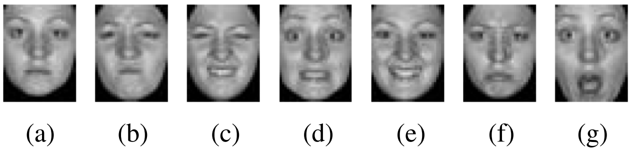

Fig. 5. *Facial images depicting a person of the BU dataset: a) neutral, b) anger, c) disgust, d) fear, e) happy, f) sad and g) surprise.*

3) The JAFFE dataset [26]: consists of 210 facial images depicting 10 Japanese female persons. Each of the persons is depicted in 3 images for each expression. Example images of the dataset are illustrated in Figure 6.

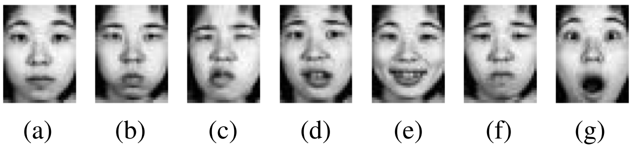

Fig. 6. *Facial images depicting a person of the JAFFE dataset: a) neutral, b) anger, c) disgust, d) fear, e) happy, f) sad and g) surprise.*

C. Action recognition datasets

1) The ASLAN dataset [30]: consists of thousands of videos collected from the web, in over 400 complex action classes. A "same/not-same" benchmark is provided, which addresses the action recognition problem as a video pair similarity problem. Specifically, the goal is to answer the following binary question: "Does a pair of videos depict the same action?". Example video frames from this dataset are illustrated in Figure 7. We used the standard partitioning provided by the database. The database consists of ten splits of video pairs, each containing 300 pairs of same actions and 300 pairs of not-same actions. The splits contain mutually exclusive action classes. This means that, action classes appearing in one split do not appear in any other split. Performance is evaluated by applying the ten-fold cross-validation procedure. In each fold, nine of the splits are used to train the algorithms and performance is measured on the remaining one. An experiment consists of ten folds, one for each test split. Performance is calculated by using the mean accuracy and the standard error from the mean (SE) over all folds.

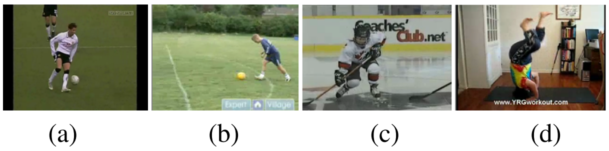

Fig. 7. *Video frames from the ASLAN dataset: a), b) "same" actions and c), d) "not-same" actions.*

2) The Hollywood2 dataset [28]: consists of 1707 sequences depicting 12 actions. It has been collected from 69 different Hollywood movies. Example video frames from this dataset are illustrated in Figure 8. We used the standard training-test split provided by the database (823 sequences are used for training and performance is measured in the remaining 884 sequences). Training and test sequences come from different movies. The performance is evaluated by computing the average precision (AP) for each action class and reporting the mean AP over all classes (mAP), as suggested in [28]. This is due to the fact that some sequences of the dataset depict multiple actions.

3) The Olympic Sports dataset [29]: consists of 783 sequences depicting athletes practicing 16 sports, which have been collected from YouTube and annotated using Amazon Mechanical Turk. Example video frames from the dataset are illustrated in Figure 9. The dataset has rich scene context information, which is helpful for recognizing sport actions. We used the standard training-test split provided by the database (649 sequences are used for training and performance is measured in the remaining 134 sequences). The performance

Fig. 8. *Video frames from the Hollywood2 dataset: a) get out of car, b) run, c) hug, d) sit up, e) drive car, f) eat, g), stand up h) sit down,i) answering phone, j) hand shaking, k) fight and l) kissing.*

is evaluated by computing the mean Average Precision (mAP) over all classes, as suggested in [29].

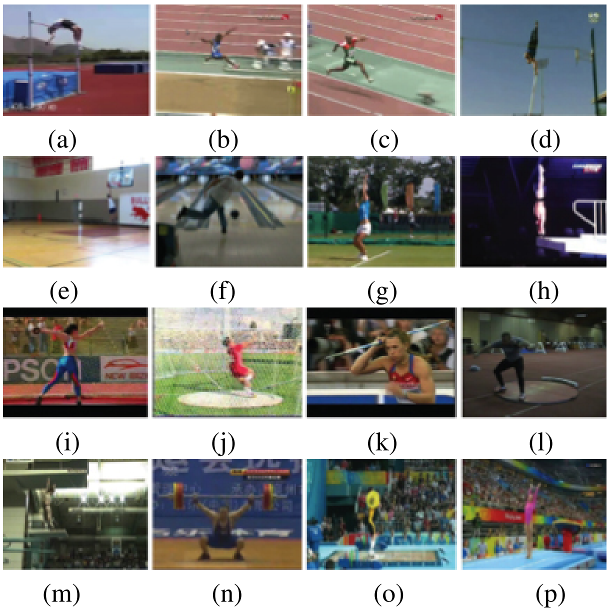

Fig. 9. *Video frames from the Olympic Sports dataset: a) high jump, b) long jump, c) triple jump, d) pole vault, e) basketball lay-up, f) bowling, g) tennis serve, h) platform, i) discus throw, j) hammer throw, k) javelin throw, l) shot-put, m) vault, n)snatch , o) clean-jerk and p) springboard.*

V. Results

A. Face Recognition Results

In our first set of experiments, we have applied the competing algorithms on the face recognition datasets. Since there is not a widely adopted experimental protocol for these datasets, we randomly partition the datasets in training and test sets as follows: we randomly select a subset of the facial images depicting each of the persons in each dataset in order to form the training set and we keep the remaining facial images for evaluation. We create five such dataset partitions, each corresponding to a different training set cardinality. Experimental results obtained by applying the competing algorithms are illustrated in Table I. The CSRDA-based classification scheme employing the Spectral Regression approach, usually outperforms the one employing the direct optimization process, i.e., CSRDA(d), in most cases. In addition, the proposed class-specific classification schemes usually outperform multi-class ones. The proposed approach, by optimizing both the data projection matrix and the class representation, also enhances class discrimination when compared to the CSKDA approach, leading to enhanced classification performance in all cases.

B. Facial Expression Recognition Results

In our second set of experiments, we have applied the competing algorithms on the facial expression recognition datasets. Since there is not a widely adopted experimental protocol for these datasets too, we apply the five-fold cross-validation procedure [43] by employing the facial expression labels. That is, we randomly split the facial images depicting the same expression in five sets and we use five splits of all the expressions for training and the remaining splits for evaluation. This process is performed five times, one for each evaluation split. Experimental results obtained by applying the competing algorithms are illustrated in Table II. The proposed CSRDA-based classification scheme outperforms the remaining choices in all the cases.

C. Action Recognition Results

Table III illustrates the performance of each classification scheme on the ASLAN dataset. The SVM classifier clearly outperforms the Discriminant Analysis-based multi-class classification schemes in all the cases. By comparing the performance of KDA, KSR and CSKDA, the adoption of a class-specific approach in CSKDA does not lead to increased class discrimination, when compared to the multi-class approach. This can be explained by the fact that the classification problem of the ASLAN dataset involves only two classes and, thus, in this case, the class-specific and two-class approaches are similar. Since in these experiments we perform a linear projection, we have also applied the Representative Class Vector Linear Discriminant Analysis (RCVLDA) [14] algorithm. The adoption of optimized class representations enhances class discrimination. Specifically, both RCVLDA and the proposed CSRDA algorithms achieve increased classification performance, when compared to the KDA and CSKDA algorithms, respectively. The proposed CSRDA-based classification scheme outperforms all the remaining Discriminant Analysis-based classification schemes. In addition, it also outperforms the SVM classifier in all the cases. Overall, it provides the best performance, equal to 61.03%, by exploiting all the 36 histogram-pair similarity values (concatenation of the similarity vectors calculated for the HOG, HOF and HNF descriptors).

Table IV illustrates the performance obtained by applying all the competing classification schemes on the Olympic Sports and Hollywood2 datasets. The proposed CSRDA-based classification scheme employing the Spectral Regression-based optimization process, outperforms all the remaining algorithms. The precision values obtained for each action class of the

TABLE I

PERFORMANCE FOR DIFFERENT TRAINING-TEST PARTITIONS ON THE FACE RECOGNITION DATASETS.

AR	NM	1NN	KSVM	KSR	KDA	CSKDA(d)	**CSRDA(d)**	CSKDA	**CSRDA**
10%	24.04%	23.91%	20.22%	35.74%	**37.26**%	24.26%	27.13%	27.13%	29.39%
20%	22.05%	27.95%	27.86%	42.48%	44.24%	42.67%	43.29%	44.1%	**44.71**%
30%	24.28%	40.44%	46.56%	66.67%	67.22%	66.28%	67%	68.56%	**69.17**%
40%	20.25%	45.38%	44.13%	65.25%	66.31%	65.44%	66.75%	66.81%	**68**%
50%	23.69%	60.31%	61.62%	85.69%	86.77%	85.85%	87.69%	87.77%	**88.54**%
ORL	NM	1NN	KSVM	KSR	KDA	CSKDA(d)	**CSRDA(d)**	CSKDA	**CSRDA**
10%	**71.67**%	**71.67**%	44.44%	57.22%	56.67%	58.89%	61.67%	51.67%	54.17%
20%	80.63%	82.19%	49.06%	86.56%	**87.81**%	74.69%	75.94%	70.94%	75.31%
30%	81.79%	85.36%	76.07%	89.64%	90.36%	76.07%	77.14%	90.71%	**91.43**%
40%	81.67%	89.17%	84.17%	92.22%	93.75%	81.25%	81.67%	93.75%	**94.58**%
50%	85%	91.5%	88%	93.5%	93%	86.5%	87%	93%	**96**%
Yale	NM	1NN	KSVM	KSR	KDA	CSKDA(d)	**CSRDA(d)**	CSKDA	**CSRDA**
10%	28.99%	43.28%	53.9%	69.69%	70.01%	68.01%	69.69%	70.51%	**71.69**%
20%	21.57%	56.86%	75.13%	85.35%	81.84%	84.98%	85.91%	86.02%	**87.56**%
30%	18.89%	61.93%	81.11%	87.89%	79.06%	85.67%	86.37%	88.77%	**89.94**%
40%	15.58%	69.94%	89.4%	93.56%	88.99%	88.99%	91%	93.63%	**94.18**%
50%	13.49%	75.82%	93.34%	96.38%	94.65%	95.56%	96.13%	96.79%	**97.62**%

TABLE II

PERFORMANCE ON THE FACIAL EXPRESSION RECOGNITION DATASETS.

	NM	1NN	KSVM	KSR	KDA	CSKDA(d)	**CSRDA(d)**	CSKDA	**CSRDA**
BU	63.57%	60.43%	67.43%	67.57%	64.43%	65.29%	66.14%	67%	**67.71**%
JAFFE	48.55%	49.05%	52.86%	53.81%	51.43%	50.95%	51.9%	56.67%	**57.14**%
KANADE	49.56%	50.2%	62.45%	65.71%	62.45%	64.49%	65.31%	64.9%	**66.12**%

TABLE III

PERFORMANCE (ACCURACY ± SE) ON THE ASLAN DATASET.

	HOG	HOF	HNF	ALL
NM	52.67 ± 0.78%	53.03 ± 0.81%	52.63 ± 0.79%	53.18 ± 0.59%
1NN	51.85 ± 0.56%	51 ± 0.94%	51.55 ± 0.81%	51.6 ± 1.05%
SVM	57.78 ± 0.82%	56.68 ± 0.56%	59.47 ± 0.66%	60.88 ± 0.77%
KDA	50.33 ± 0.38%	50.28 ± 0.27%	49.82 ± 0.31%	51.20 ± 0.43%
KSR	55.42 ± 0.48%	51.82 ± 0.45%	54.5 ± 0.99%	54.9 ± 0.71%
RCVLDA	59.70 ± 0.91%	56.93 ± 0.63%	59.17 ± 0.72%	60.95 ± 0.81%
CSKDA(d)	55.42 ± 0.48%	51.47 ± 0.44%	54.5 ± 0.99%	54.9 ± 0.71%
CSRDA(d)	59.65 ± 0.69%	57.1 ± 0.57%	59.38 ± 0.6%	60.45 ± 0.82%
CSKDA	55.4 ± 0.47%	51.81 ± 0.45%	54.48 ± 0.99%	54.9 ± 0.71%
CSRDA	**60.08± 0.68**%	**57.33± 0.57**%	**59.98± 0.83**%	**61.03± 0.54**%

Olympic Sports and the Hollywood2 datasets are provided in Tables V, VI, respectively. In these Tables we also provide the precision values obtained by applying SVM and CSKDA algorithms.

TABLE IV

PERFORMANCE (MAP) ON THE OLYMPIC SPORTS AND HOLLYWOOD2 DATASETS.

	Olympic Sports	Hollywood2
NM	62.38%	46.59%
1NN	54.98%	38.54%
KSVM	86.56%	61.51%
KSR	88.35%	61.34%
KDA	88.64%	61.04%
CSKDA(d)	85.86%	59.93%
CSRDA(d)	86.93%	60.65%
CSKDA	87.65%	60.5%
CSRDA	**88.89**%	**61.69**%

TABLE V

PRECISION VALUES ON THE OLYMPIC SPORTS DATASET.

	KSVM	CSKDA	**CSRDA**
Basketball lay-up	96.69%	**98.35**%	**98.35**%
Bowling	82.92%	89.42%	**90.71**%
Clean and Jerk	**88.68**%	86.35%	86.35%
Discus Throw	**92.33**%	91.51%	**92.33**%
Diving 3m	**100**%	**100**%	**100**%
Diving 10m	**100**%	**100**%	**100**%
Hammer Throw	93.01%	93.01%	**96.36**%
High Jump	60.82%	**71.2**%	**71.2**%
Javelin Throw	**100**%	**100**%	**100**%
Long Jump	84.85%	**88.31**%	**88.31**%
Pole Vault	**92.21**%	86.15%	86.15%
Shot Put	78.34%	79.96%	**81.91**%
Snatch	**77.64**%	73.64%	77.08%
Triple Jump	50.35%	**66.78**%	66.32%
Tennis Serve	93.39%	**100**%	**100**%
Vault	77.74%	77.68%	**87.17**%
Mean	85.56%	87.65%	**88.89**%

In Table VII, we compare the performance of the adopted action recognition method with that of some other methods evaluating their performance on Olympic Sports and Hollywood2 datasets. The proposed CSRDA algorithms, when combined with the improved trajectory-based video represen-

TABLE VI

Precision values on the Hollywood2 dataset.

	KSVM	CSKDA	**CSRDA**
Answer Phone	**40.98**%	28.12%	39.41%
Drive Car	89.26%	90.4%	**90.72**%
Eat	**67.19**%	66.7%	65.82%
Fight	78.98%	79.41%	**80.4**%
Get Out of Car	**60.95**%	58.49%	59.62%
Hand Shake	**42.47**%	40.83%	40.56%
Hug Person	**47.07**%	45.52%	46.01%
Kiss	62.87%	62.33%	**64.13**%
Run	83.13%	83.26%	**83.37**%
Sit Down	69.73%	**70.89**%	70.33%
Sit up	22.34%	24.82%	**25.16**%
Stand up	73.2%	**75.23**%	74.42%
Mean	61.51%	60.5%	**61.69**%

tation achieves satisfactory performance in both datasets.

TABLE VII

Comparison of our results with some methods on the Olympic Sports and Hollywood2 datasets.

	Olympic Sports	Hollywood2
Brendel et al. [44]	77.33%	-
Vig et al. [45]	-	61.9%
Gaidon et al. [46]	82.7%	-
Mathe et al. [47]	-	61%
Jiang et al. [48]	80.6%	59.5%
Jain et al. [49]	83.2%	**62.5**%
Improved Trajectories+CSRDA(d)	**86.93**%	60.65%
Improved Trajectories+CSRDA	**88.89**%	**61.69**%

D. Statistical Significance Analysis

We performed the Friedman test suggested in [50] to test whether k classifiers applied to M classification problems perform equally well. In the analysis we include the four related classifiers (i.e., KSR, KDA, CSKDA and the proposed CSRDA based classification schemes) and compare their performance in $M = 9$ classification problems (i.e. three face recognition, three facial expression and three action recognition problems discussed in subsections V-A, V-B and V-C, respectively). Since the results obtained for different dataset partitions on the face recognition datasets are correlated, we rank the algorithms based on their performance on each case and calculate the mean rank in order to determine the order of each algorithm. The same process is applied for the different histogram-pair similarity value sets on the ASLAN dataset. The mean rank of each algorithm and the overall mean rank are illustrated in Table VIII. There was statistical evidence of a difference between the four classifiers ($F_{3,24} = 9.34$, $p = 0.001$).

In order to test whether the proposed CSRDA based classification scheme performs equally well with each of the remaining three classifiers, we performed the Nemenyi test suggested in [50] for pairwise comparisons when the classifiers are applied to M classification problems. We compare performance of the proposed CSRDA based classification scheme with that of the KSR, KDA and CSKDA based classifiers in $M = 9$ classification problems (i.e. three face recognition, three facial expression and three action recognition problems discussed in subsections V-A, V-B and V-C, respectively). The differences between the ranks of CSKDA, KDA and KSR with the rank of the proposed CSRDA algorithm are illustrated in Table IX. The proposed CSRDA based classification scheme performs significantly better than the CSKDA and KDA based classification schemes, while it performs better than the KSR based classification scheme ($CD = 1.56$, $p = 0.001$).

TABLE VIII

Mean ranks for Friedman test.

KSR	KDA	CSKDA	CSRDA	Overall
2.64	3.28	2.93	1.13	2.49

TABLE IX

Mean rank differences for Nemenyi test.

CSKDA	KDA	KSR
1.8	2.15	1.51

VI. Conclusions

In this paper, we proposed a novel class-specific discriminant subspace learning algorithm for the determination of a discriminant space based on an optimized class representation. Two optimization methods have been proposed to this end, where the first solves the original criterion for optimal data projections calculation, while the second solves an approximation of the original criterion leading to faster optimization. The adopted optimized class representation is also determined by applying an optimization process aiming at enhancing class discrimination in the resulting discriminant space. These two optimization steps are iteratively repeated, in order to determine both the optimal data projections and the optimal class representation, which will be subsequently used in order to decide whether a test sample belongs to the class under consideration, or not. The proposed approach has been applied to three problems relating to human behaviour analysis, i.e., the recognition of human face, facial expression and activity, where it has achieved good performance. Comparative results with other related classification schemes denote the effectiveness of the proposed method.

Acknowledgment

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement number 316564 (IMPART).

Appendix I
Derivation of $\mathcal{J}(\mathbf{A})$ (8)

We expand $\mathbf{W}^T\mathbf{S}_j\mathbf{W}$:

$$\begin{aligned}
\mathbf{W}^T\mathbf{S}_j\mathbf{W} &= \sum_{i=1}^{N_{j1}}\left(\mathbf{W}^T\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T\mathbf{W}\right)\\
&- \mathbf{W}^T\mathbf{\Phi}_j\mathbf{1}_{N_{j1}}\mathbf{b}_j^T\mathbf{\Phi}_j^T\mathbf{W}\\
&- \mathbf{W}^T\mathbf{\Phi}_j\mathbf{b}_j\mathbf{1}_{N_{j1}}^T\mathbf{\Phi}_j^T\mathbf{W}\\
&+ \frac{1}{N_{j1}}\mathbf{W}^T\mathbf{\Phi}\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{\Phi}^T\mathbf{W}\\
&= \mathbf{A}^T\mathbf{K}_j\left(\mathbf{I}-\mathbf{1}_{N_{j1}}\mathbf{b}_j^T-\mathbf{b}_j\mathbf{1}_{N_{j1}}^T\right.\\
&+ \left. N_{j1}\mathbf{b}_j\mathbf{b}_j^T\right)\mathbf{K}_j^T\mathbf{A} = \mathbf{A}^T\mathbf{M}_j\mathbf{A}, \qquad (21)
\end{aligned}$$

where $\mathbf{1}_{N_{j1}} \in \mathbb{R}^{N_{j1}}$ is a vector of ones and:

$$\mathbf{M}_j = \mathbf{K}_j\left(\mathbf{I}-\mathbf{1}_{N_{j1}}\mathbf{b}_j^T-\mathbf{b}_j\mathbf{1}_{N_{j1}}^T+N_{j1}\mathbf{b}_j\mathbf{b}_j^T\right)\mathbf{K}_j^T. \qquad (22)$$

We also expand $\mathbf{W}^T\mathbf{S}_0\mathbf{W}$:

$$\begin{aligned}
\mathbf{W}^T\mathbf{S}_0\mathbf{W} &= \sum_{i=1}^{N_{j0}}\left(\mathbf{W}^T\phi(\mathbf{z}_i)\phi(\mathbf{z}_i)^T\mathbf{W}\right)\\
&- \mathbf{W}^T\mathbf{\Phi}_0\mathbf{1}_{N_{j0}}\mathbf{b}_j^T\mathbf{\Phi}_j^T\mathbf{W}\\
&- \mathbf{W}^T\mathbf{\Phi}_j\mathbf{b}_j\mathbf{1}_{N_{j0}}^T\mathbf{\Phi}_0^T\mathbf{W}\\
&+ \frac{1}{N_{j1}}\mathbf{W}^T\mathbf{\Phi}\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{\Phi}^T\mathbf{W}\\
&= \mathbf{A}^T\left[\sum_{i=1}^{N_{j0}}\left(\mathbf{K}_0\mathbf{K}_0^T\right)+\mathbf{K}_j\left(\mathbf{1}_{N_{j1}}\mathbf{b}_j^T\right.\right.\\
&+ \left.\left.\mathbf{b}_j\mathbf{1}_{N_{j1}}^T+N_{j0}\mathbf{b}_j\mathbf{b}_j^T\right)\mathbf{K}_j^T\right]\mathbf{A}\\
&= \mathbf{A}^T\mathbf{M}_0\mathbf{A}, \qquad (23)
\end{aligned}$$

where $\mathbf{1}_{N_{j0}} \in \mathbb{R}^{N_{j0}}$ is a vector of ones and:

$$\mathbf{M}_0 = \sum_{i=1}^{N_{j0}}\left(\mathbf{K}_0\mathbf{K}_0^T\right)+\mathbf{K}_j\left(\mathbf{1}_{N_{j1}}\mathbf{b}_j^T+\mathbf{b}_j\mathbf{1}_{N_{j1}}^T+N_{j0}\mathbf{b}_j\mathbf{b}_j^T\right)\mathbf{K}_j^T. \qquad (24)$$

By using (21), (23) we obtain:

$$\mathcal{J}_{\mathbf{A}} = \frac{tr\left(\mathbf{A}^T\mathbf{M}_0\mathbf{A}\right)}{tr\left(\mathbf{A}^T\mathbf{M}_j\mathbf{A}\right)}. \qquad (25)$$

Appendix II
Derivation of $\mathbf{P}_0\mathbf{z} = \lambda\mathbf{P}_j\mathbf{z}$

Let us denote by $\mathbf{1}_j \in \mathbb{R}^N$ a vector having its elements equal to $\mathbf{1}_{ji} = 1$ if $\mathbf{c}_{ji} = 1$ and $\mathbf{1}_{ji} = 0$ otherwise. Let us also denote by $\mathbf{1}_0 \in \mathbb{R}^N$ a vector having its elements equal to $\mathbf{1}_{0i} = 1$ if $\mathbf{c}_{ji} = 0$ and $\mathbf{1}_{0i} = 0$ otherwise. By using these vectors, we define the matrices $\mathbf{J}_1 = \mathbf{1}_j\mathbf{1}_j^T$ and $\mathbf{J}_2 = \mathbf{1}_0\mathbf{1}_0^T$.

We expand $\mathbf{S}_0\mathbf{v}$:

$$\begin{aligned}
\mathbf{S}_0\mathbf{v} &= \sum_{k=1}^{N}\alpha_k\sum_{i=1}^{N_{j0}}\left(\phi(\mathbf{z}_i)\phi(\mathbf{z}_i)^T\phi(\mathbf{x}_k)\right.\\
&- \phi(\mathbf{z}_i)\mathbf{b}_j^T\mathbf{\Phi}_j^T\phi(\mathbf{x}_k)-\mathbf{\Phi}_j\mathbf{b}_j\phi(\mathbf{z}_i)\phi(\mathbf{x}_k)\\
&+ \left.\mathbf{\Phi}_j\mathbf{b}_j\mathbf{b}_j^T\mathbf{\Phi}_j^T\phi(\mathbf{x}_k)\right). \qquad (26)
\end{aligned}$$

By using (26), we obtain:

$$\begin{aligned}
\mathbf{\Phi}^T\mathbf{S}_0\mathbf{v} &= \sum_{k=1}^{N}\alpha_k\sum_{i=1}^{N_{j0}}\left(\mathbf{\Phi}^T\phi(\mathbf{z}_i)\phi(\mathbf{z}_i)^T\phi(\mathbf{x}_k)\right.\\
&- \mathbf{\Phi}^T\phi(\mathbf{z}_i)\mathbf{b}_j^T\mathbf{\Phi}_j^T\phi(\mathbf{x}_k)\\
&- \mathbf{\Phi}^T\mathbf{\Phi}_j\mathbf{b}_j\phi(\mathbf{z}_i)\phi(\mathbf{x}_k)\\
&+ \left.\mathbf{\Phi}^T\mathbf{\Phi}_j\mathbf{b}_j\mathbf{b}_j^T\mathbf{\Phi}_j^T\phi(\mathbf{x}_k)\right)\\
&= \mathbf{K}\mathbf{P}_0\mathbf{K}\alpha, \qquad (27)
\end{aligned}$$

where $\mathbf{P}_0 = \mathbf{J}_2 - \mathbf{1}_0\mathbf{d}_j^T - \mathbf{d}_j\mathbf{1}_0^T + \mathbf{d}_j\mathbf{d}_j^T$. $\mathbf{d}_j \in \mathbb{R}^N$ is a vector having its elements equal to $\mathbf{d}_{ji} = \gamma$ if $\mathbf{c}_{ji} = 1$ and $\mathbf{d}_{ji} = 0$, otherwise.

We expand $\mathbf{S}_j\mathbf{v}$:

$$\begin{aligned}
\mathbf{S}_j\mathbf{v} &= \sum_{k=1}^{N}\alpha_k\sum_{i=1}^{N_{j0}}\left(\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_k)\right.\\
&- \phi(\mathbf{x}_i)\mathbf{b}_j^T\mathbf{\Phi}_j^T\phi(\mathbf{x}_k)-\mathbf{\Phi}_j\mathbf{b}_j\phi(\mathbf{x}_i)\phi(\mathbf{x}_k)\\
&+ \left.\mathbf{\Phi}_j\mathbf{b}_j\mathbf{b}_j^T\mathbf{\Phi}_j^T\phi(\mathbf{x}_k)\right). \qquad (28)
\end{aligned}$$

By using (28), we obtain:

$$\begin{aligned}
\mathbf{\Phi}^T\mathbf{S}_j\mathbf{v} &= \sum_{k=1}^{N}\alpha_k\sum_{i=1}^{N_{j0}}\left(\mathbf{\Phi}^T\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_k)\right.\\
&- \mathbf{\Phi}^T\phi(\mathbf{x}_i)\mathbf{b}_j^T\mathbf{\Phi}_j^T\phi(\mathbf{x}_k)\\
&- \mathbf{\Phi}^T\mathbf{\Phi}_j\mathbf{b}_j\phi(\mathbf{x}_i)\phi(\mathbf{x}_k)\\
&+ \left.\mathbf{\Phi}^T\mathbf{\Phi}_j\mathbf{b}_j\mathbf{b}_j^T\mathbf{\Phi}_j^T\phi(\mathbf{x}_k)\right)\\
&= \mathbf{K}\mathbf{P}_j\mathbf{K}\alpha, \qquad (29)
\end{aligned}$$

where $\mathbf{P}_j = \mathbf{J}_1 - \mathbf{1}_j\mathbf{d}_j^T - \mathbf{d}_j\mathbf{1}_j^T + \mathbf{d}_j\mathbf{d}_j^T$.

By using $\mathbf{K}\alpha = \mathbf{z}$, and (27), (29) the result follows.

Appendix III
Derivation of $\gamma(\mathbf{W})$

The derivatives of (21), (23) with respect to $\mathbf{b}_j$ are given by:

$$\frac{\vartheta tr\left(\mathbf{A}^T\mathbf{M}_j\mathbf{A}\right)}{\vartheta\mathbf{b}_j} = 2N_{j1}\mathbf{K}_j^T\mathbf{A}\mathbf{A}^T\mathbf{K}_j\left(\mathbf{b}_j-\mathbf{1}_{N_{j1}}\right) \qquad (30)$$

and

$$\frac{\vartheta tr\left(\mathbf{A}^T\mathbf{M}_0\mathbf{A}\right)}{\vartheta\mathbf{b}_j} = 2\mathbf{K}_j^T\mathbf{A}\mathbf{A}^T\mathbf{K}_j\left(N_{j0}\mathbf{b}_j-N_{j1}\mathbf{1}_{N_{j1}}\right). \qquad (31)$$

By using (30) and (31) we obtain:

$$\left(N - N_{j1} - \lambda N_{j1}\right)\mathbf{b}_j + \left(N_{j1} + \lambda N_{j1}\right)\mathbf{1}_{N_{j1}} = 0. \tag{32}$$

From (32) it is straightforward that:

$$\mathbf{b}_j = \frac{\gamma}{N_{j1}}\mathbf{1}_{N_{j1}}. \tag{33}$$

In order to calculate γ, we replace (33) in $\mathbf{A}^T\mathbf{M}_j\mathbf{A}$ and $\mathbf{A}^T\mathbf{M}_0\mathbf{A}$ and we obtain:

$$\begin{aligned}\mathbf{A}^T\mathbf{M}_j\mathbf{A} &= \mathbf{A}^T\Bigg[\mathbf{K}_j\mathbf{I}\mathbf{K}_j^T - \frac{2\gamma}{N_{j1}}\mathbf{K}_j\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{K}_j^T \\ &+ \frac{\gamma^2}{N_{j1}}\mathbf{K}_j\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{K}_j^T\Bigg]\mathbf{A}, \end{aligned} \tag{34}$$

$$\begin{aligned}\mathbf{A}^T\mathbf{M}_0\mathbf{A} &= \mathbf{A}^T\Bigg[\mathbf{K}_0\mathbf{K}_0^T + \frac{2\gamma}{N_{j1}}\mathbf{K}_j\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{K}_j^T \\ &+ \frac{\gamma^2 N_{j0}}{N_{j1}^2}\mathbf{K}_j\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{K}_j^T\Bigg]\mathbf{A} \end{aligned} \tag{35}$$

and

$$tr\left(\mathbf{A}^T\mathbf{M}_j\mathbf{A}\right) = \frac{\gamma^2 - 2\gamma}{N_{j1}}tr\left(\mathbf{A}^T\mathbf{K}_j\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{K}_j^T\mathbf{A}\right) + c_1, \tag{36}$$

$$\begin{aligned}tr\left(\mathbf{A}^T\mathbf{M}_0\mathbf{A}\right) &= \frac{N_{j0}\gamma^2 - 2N_{j1}\gamma}{N_{j1}^2}tr\left(\mathbf{A}^T\mathbf{K}_j\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{K}_j^T\mathbf{A}\right) \\ &+ c_2. \end{aligned} \tag{37}$$

By substituting $b = tr\left(\mathbf{A}^T\mathbf{K}_0\mathbf{K}_0^T\mathbf{A}\right)$, $e = tr\left(\mathbf{A}^T\mathbf{K}_1\mathbf{1}_{N_{j1}}\mathbf{1}_{N_{j1}}^T\mathbf{K}_1^T\mathbf{A}\right)$ and $f = tr\left(\mathbf{A}^T\mathbf{K}_1\mathbf{K}_1^T\mathbf{A}\right)$ we obtain:

$$tr\left(\mathbf{A}^T\mathbf{M}_j\mathbf{A}\right) = N_{j0}\gamma^2 e + 2N_{j1}\gamma + b, \tag{38}$$

$$tr\left(\mathbf{A}^T\mathbf{M}_0\mathbf{A}\right) = N_{j1}\gamma^2 e + 2N_{j1}\gamma + f \tag{39}$$

and

$$\gamma = \frac{h + \left(h^2 + 4q\left(b + f\right)N_{j1}e\right)^{1/2}}{2qe}, \tag{40}$$

where $h = fN_{j0} - bN_{j1}$ and $q = N_{j1}^2 + N_{j1}N_{j0}$.

By analyzing b, e and f we have:

$$\begin{aligned} b &= tr\left(\mathbf{A}^T\mathbf{K}_0\mathbf{K}_0^T\mathbf{A}\right) = \sum_{k=1}^{N_{j0}}\phi(\mathbf{z}_k)^T\mathbf{W}\mathbf{W}^T\phi(\mathbf{z}_k) \\ &= \sum_{i=1}^{N_{j0}}\|\mathbf{z}_{k0}\|^2. \end{aligned} \tag{41}$$

That is, b is equal to the sum of the (squared) Euclidean norm of the training data corresponding to $\mathbf{c}_{ji} = 0$ in the obtained discriminant space $\mathbb{R}^{d_j}$.

$$\begin{aligned} e &= tr\left(\mathbf{A}^T\mathbf{\Phi}^T\phi(\boldsymbol{\mu}_j)\phi(\boldsymbol{\mu}_j)^T\mathbf{\Phi}\mathbf{A}\right) \\ &= tr\left(\mathbf{W}^T\phi(\boldsymbol{\mu}_j)\phi(\boldsymbol{\mu}_j)^T\mathbf{W}\right) = tr\left(\mathbf{z}_j\mathbf{z}_j^T\right), \end{aligned} \tag{42}$$

where $\mathbf{z}_j$ is the reference class vector in the obtained discriminant space $\mathbb{R}^{d_j}$.

$$f = tr\left(\mathbf{A}^T\mathbf{K}_1\mathbf{K}_1^T\mathbf{A}\right) = tr\left(\mathbf{W}^T\mathbf{\Phi}_j\mathbf{\Phi}_j^T\mathbf{W}\right) = tr\left(\mathbf{Z}_j\mathbf{Z}_j^T\right). \tag{43}$$

References

[1] P. Barr, J. Noble, and R. Biddle, "Video game values: Human-computer interaction and games," *Interacting with Computers*, vol. 19, no. 2, pp. 180–195, 2007.

[2] A. Tefas and I. Pitas, "Human centered interfaces for assisted living," *International Conference on Man-Machine Interactions*, 2011.

[3] L. Juwei, K. Plataniotis, and A. Venetsanopoulos, "Face recognition using kernel direct discriminant analysis algorithms," *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 117–126, 2003.

[4] G. Goudelis, S. Zafeiriou, A. Tefas, and I. Pitas, "Class-specific kernel-discriminant analysis for face verification," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 570–587, 2007.

[5] A. Maronidis, D. Bolis, A. Tefas, and I. Pitas, "Improving subspace learning for facial expression recognition using person dependent and geometrically enriched training sets," *Neural Networks*, vol. 24, no. 8, pp. 814–823, 2011.

[6] A. Iosifidis, A. Tefas, and I. Pitas, "Multi-view human movement recognition based on fuzzy distances and Linear Discriminant Analysis," *Computer Vision and Image Understanding*, vol. 116, no. 3, pp. 347–360, 2012.

[7] R. Duda, P. Hart, and D. Stork, *Pattern Classification, 2nd ed.* Wiley-Interscience, 2000.

[8] L. Chengjun and H. Wechsler, "Independent component analysis of gabor features for face recognition," *IEEE Transactions on Neural Networks*, vol. 14, no. 4, pp. 919–928, 2003.

[9] M. Bartlett, J. Movellan, and T. Sejnowski, "Face recognition by independent component analysis," *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1450–1464, 2002.

[10] D. Lee and H. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[11] A. Martinez and A. Kak, "PCA versus LDA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228–233, 2001.

[12] D. Cai, X. He, and J. Han, "Speed up kernel discriminant analysis," *International Journal on Very Large Data Bases*, vol. 20, no. 1, pp. 21–33, 2011.

[13] Y. Jia, F. Nie, and C. Zhang, "Trace Ratio problem revisited," *IEEE Transactions on Neural Networks*, vol. 20, no. 4, pp. 729–735, 2009.

[14] A. Iosifidis, A. Tefas, and I. Pitas, "On the optimal class representation in Linear Discriminant Analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 9, pp. 1491–1497, 2013.

[15] E. Gopi and P. Palanisamy, "Formulating particle swarm optimization based membership Linear Discriminant Analysis," *Swarm and Evolutionalry Computation*, vol. 12, pp. 65–73, 2013.

[16] A. Iosifidis, A. Tefas, and I. Pitas, "Kernel Reference Discriminant Analysis," *Pattern Recognition Letters*, vol. 49, pp. 85–91, 2014.

[17] H. Wang, M. Ullah, A. Klaser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," *British Machine Vision Conference*, 2009.

[18] A. Iosifidis, A. Tefas, and I. Pitas, "minimum Class Variance Extreme Learning Machine for human action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 11, pp. 1968–1979, 2013.

[19] K. 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.

[20] B. Scholkopf, S. Mika, C. Burges, P. Knirsch, K. Muller, G. Ratsch, and A. Smola, "Input space vs. feature space in kernel-based methods," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1000–1017, 1999.

[21] W. Zheng, L. Zhao, and Z. Cairong, "Foley-Sammon optimal discriminant vectors using kernel approach," *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 1–9, 2005.

[22] Q. Liu, X. Tang, H. Lu, and S. Ma, "Face recognition using kernel scatter-difference-based discriminant analysis," *IEEE Transactions on Neural Networks*, vol. 17, no. 4, pp. 1081–1085, 2006.

[23] F. Samaria and A. Harter, "Parameterisation of a stochastic model for human face identification," *IEEE Workshop on Applications of Computer Vision*, 1994.

[24] K. Lee, J. Ho, and D. Kriegman, “Acquiriing linear subspaces for face recognition under variable lighting,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 684–698, 2005.

[25] T. Kanade, Y. Tian, and J. Cohn, “Comprehensive database for facial expression analysis,” *IEEE International Conference on Automatic Face and Gesture Recognition*, 2000.

[26] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, “Coding facial expressions with gabor wavelets,” *IEEE International Conference on Automatic Face and Gesture Recognition*, 1998.

[27] L. Yin, X. Wei, Y. Sun, J. Wang, and M. Rosato, “A 3D facial expression database for facial behavior research,” *IEEE International Conference on Automatic Face and Gesture Recognition*, 2006.

[28] M. Marszalek, I. Laptev, and C. Schmid, “Actions in context,” *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[29] J. Niebles, C. Chend, and L. Fei-Fei, “Modeling temporal structure of decomposable mition segemnts for activity classification,” *European Conference on Computer Vision*, 2010.

[30] O. Gross, T. Hassner, and L. Wolf, “The action similarity labeling challenge,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 615–621, 2013.

[31] S. Zafeiriou, A. Tefas, and I. Pitas, “Learning discriminant person-specific facial models using expandable graphs,” *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 1, pp. 55–68, 2007.

[32] A. Argyriou, C. Micchelli, and M. Pontil, “When is there a representer theorem? Vector versus matrix regularizers,” *Journal of Machine Learning Research*, vol. 10, pp. 2507–2529, 2009.

[33] J. Tenenbaum, V. Silva, and J. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290, pp. 2319–2323, 2000.

[34] S. Boweils and I. Saul, “Nonlinear dimensionality reduction by Locally Linear Embedding,” *Science*, vol. 290, pp. 2323–2326, 2000.

[35] Y. Guo, S. Li, J. Yang, T. Shu, and L. Wu, “A generalized Foley-Sammon transform based on generalized Fisher discriminant criterion and its application to face recognition,” *Pattern Recognition Letters*, vol. 24, no. 3, pp. 147–158, 2003.

[36] S. Yan and X. Tang, “Trace quotient problem revisited,” *European Conference on Computer Vision*, 2006.

[37] H. Wang, S. Yan, D. Xu, X. Tang, and T. Huang, “Trace Ratio vs. Ratio Trace for dimensionality reduction,” *Computer Vision and Pattern Recognition*, 2007.

[38] R. Fan, P. Chen, and C. J. Lin, “Working set selection using the second order information for training svm,” *Journal of Machine Learning Research*, vol. 4, no. 6, pp. 1889–1918, 2005.

[39] H. Wang and C. Schmid, “Action recognition with improved trajectories,” *International Conference on Computer Vision*, 2013.

[40] I. Laptev, “On space-time interest points,” *International Journal of Computer Vision*, vol. 64, no. 2, pp. 107–123, 2005.

[41] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning realistic human actions from movies,” *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[42] J. Zhang, M. Marszalek, M. Lazebnik, and C. Schmid, “Local features and kernels for classification of texture and object categories: A comprehensive study,” *International Journal of Computer Vision*, vol. 73, no. 2, pp. 213–238, 2007.

[43] P. Devijver and J. Kittler, *Pattern Recognition: A Statistical Approach*. Prentice-Hall, 1982.

[44] W. Brendel and S. Todorovic, “Learning spatiotemporal graphs of human activities,” *International Conference on Computer Vision*, 2011.

[45] E. Vig, M. Dorr, and D. Cox, “Space-variant descriptor sampling for action recognition based on saliency and eye movements,” *European Conference on Computer Vision*, 2012.

[46] A. Gaidon, Z. Harchaoui, and C. Schmid, “Recognizing activities with cluster-tries of tracklets,” *British Machine Vision Conference*, 2012.

[47] S. Mathe and C. Sminchisescu, “Dynamic eye movement datasets and learnt saliency models for visual action recognition,” *European Conference on Computer Vision*, 2012.

[48] Y. Jiang, Q. Dai, X. Xue, W. Liu, and C. Ngo, “Trajectory-based modeling of human actions with motion reference points,” *European Conference on Computer Vision*, 2012.

[49] M. Jain, H. Jegou, and P. Bouthemy, “Better exploiting motion for better action recognition,” *Computer Vision and Pattern Recognition*, 2013.

[50] J. Demsar, “Statistical comparisons of classifiers over multiple data sets,” *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

Alexandros Iosifidis received a Diploma in Electrical & Computer Engineering in 2008 and a Master of Engineering in the area of Mechatronics in 2010 from the Democritus University of Thrace, Greece. He also received a Ph.D. in Informatics in 2014 from the Aristotle University of Thessaloniki, Greece. He is a postdoctoral researcher at the Artificial Intelligence and Information Analysis laboratory of the Department of Informatics in Aristotle University of Thessaloniki. His research interests include image processing, computer vision and pattern recognition.

Anastasios Tefas (S97-M04) 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 2013 he has been an Assistant Professor at the Department of Informatics, Aristotle University of Thessaloniki. From 2008 to 2012, he was a Lecturer at the same University. His current research interests include computational intelligence, pattern recognition, statistical machine learning, digital signal and image processing and computer vision.

Ioannis Pitas (SM94-F07) received the Diploma and PhD degree in Electrical Engineering, both from the Aristotle University of Thessaloniki, Greece. Since 1994, he has been a Professor at the Department of Informatics of the same University. His current interests are in the areas of image/video processing, intelligent digital media, machine learning, human centered interfaces, affective computing, computer vision, $3D$ imaging and biomedical imaging.