Graph-regularized Multi-class Support Vector Machines for Face and Action Recognition

Alexandros Iosifidis and Moncef Gabbouj
Department of Signal Processing, Tampere University of Technology, Tampere, Finland
Email: {alexandros.iosifidis,moncef.gabbouj}@tut.fi

***Abstract*—In this paper, we formulate a variant of the Support Vector Machine classifier that exploits graph-based discrimination criteria within a multi-class optimization process. We employ two kNN graphs in order to describe intra-class and between-class data relationships. These graph structures are combined in order to form a regularizer which is used in order to regularize the multi-class SVM optimization problem. The derived multi-class classifier is compared with the standard SVM classifier and SVM formulations exploiting geometric class information on six publicly available databases designed for human action recognition in the wild and facial image classification problems, where its effectiveness is shown.**

I. Introduction

Support Vector Machine (SVM) [1] is a binary classifier that determines a decision function discriminating two classes with maximal margin. It has been employed in many classification problems due to its good generalization power and its ability to determine a global optimal solution, since it is formulated as a quadratic convex optimization problem. Moreover, non-linear decision functions can be detemrined by exploiting the well-known kernel trick. In order to determine multiple decision functions, binary SVMs are usually combined following the One-Versus-Rest (OVR) or the One-Versus-One (OVO) schemes. This means that for a K-class classification problem, multiple (K for OVR and $\frac{K(K-1)}{2}$ for OVO schemes) binary SVMs are independently trained, each of which determines a binary decision function during the training phase. In the test phase, samples are introduced to all binary SVMs and the classifiers' outputs are combined in order to obtain the final classification result [2]. In order to exploit the inter-relationships that may appear between multiple classes, SVM variants that solve a joint optimization problem determining K decision functions (one per class in an OVR manner) have been proposed [3].

In standard binary and multi-class SVM methods, the obtained decision functions are determined to be the ones providing the maximal margin between the classes to be discriminated. Such an approach, while being very effective, disregards geometric properties of the classes forming the classification problem to be solved. It has been shown that the exploitation of class geometric properties within the maximum margin-based classification framework can lead to increased classification performance. Specifically, it has been shown that the exploitation of the intra-class variance within the binary SVM formulation can lead in enhanced performance [4], [5], [6]. In addition, it has been shown that the exploitation of intrinsic graph structures defined under the Graph Embedding framework [10] further enhances the performance of the resulting classifier [7]. For the multi-class SVM formulation, the intra-class variance has also been exploited in [8], [9].

All the above approaches combine intrinsic class geometric properties (i.e. properties to be minimized) with the maximum margin property of SVMs in order to enhance its generalization performance. As has been explained in [7], [9], this process is equivalent to a two-step process where the training data are first mapped to a new feature space in which the adopted intrinsic class geometric property is minimized. In that space, the standard SVM classifier is subsequently applied. A question that arises from observing this two-step process is whether the exploitation of discrimination criteria formulated using both intrinsic (to be minimized) and penalty (to be maximized) geometric properties of the classes forming the classification problem would further increase the generalization ability of the classifier.

In this paper, we first formulate an optimization problem for SVM-based multi-class classification that exploits a regularizer combining both intrinsic and penalty geometric properties of the classes forming the classification problem at hand. We design this regularizer to be in line with the Graph Embedding framework [10], which has been widely exploited in Discriminant Analysis-based subspace learning. That is, we define an intrinsic graph expressing properties of the data that are subject to minimization and a penalty graph expressing properties of the data that are subject to maximization. We combine these two graph structures and incorporate them in the multi-class SVM formulation of [3]. We apply the proposed method in human action recognition and facial image classification problems, where we compare its performance with both standard SVM and SVM formulations exploiting intrinsic class geometric properties.

II. Related Work

Let us denote by $\{\mathbf{x}_i, l_i\}$, $i = 1, \ldots, N$ a set of D-dimensional vectors $\mathbf{x}_i$ and the corresponding class labels $l_i \in \{1, \ldots, K\}$. We would like to train a multi-class classification scheme that is able to classify a test vector $\mathbf{x}_t \in \mathbb{R}^D$ to one of the K classes.

A. Binary SVM classifier

In order to exploit binary SVMs for multi-class classification we define binary labels $y_i \in \{-1, 1\}$ denoting whether the

978-0-9928-6265-7/16/$31.00 ©2016 IEEE

vectors $\mathbf{x}_i$ belong to the positive or negative class of the binary classification problem at hand. In SVM, the optimal decision function is obtained by solving the following optimization problem:

$$\underset{\mathbf{w},b}{min}\ \frac{1}{2}\mathbf{w}^T\mathbf{w} + c\sum_{i=1}^{N}\xi_i, \tag{1}$$

$$s.t.:\ \ y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i,\ \ \xi_i \geq 0, \quad i = 1, \ldots, N, \tag{2}$$

where $\mathbf{w} \in \mathbb{R}^D$ defines the decision hyperplane, b is the offset of the hyperplane from the origin, ξ_i, $i = 1, \ldots, N$ are the so-called slack variables and $c > 0$ is a regularization parameter. The solution of the above-described optimization problem is a quadratic optimization problem of the form:

$$\underset{\boldsymbol{\alpha}}{max}\ \mathbf{1}^T\boldsymbol{\alpha} - (\boldsymbol{\alpha} \circ \mathbf{y})^T\mathbf{K}(\boldsymbol{\alpha} \circ \mathbf{y}) \tag{3}$$

subject to $0 \leq \alpha_i \leq c$, $i = 1, \ldots, N$. $\boldsymbol{\alpha} \in \mathbb{R}^N$ is a vector containing the Lagrange multipliers α_i, $i = 1, \ldots, N$, $\mathbf{y} \in \mathbb{R}^N$ is a vector containing the binary labels y_i, $i = 1, \ldots, N$ and $\circ$ denotes the Hadamard (element-wise) product operator.. $\mathbf{K} \in \mathbb{R}^{N \times N}$ is the so-called kernel matrix. For linear decision functions $[\mathbf{K}]_{ij} = \mathbf{x}_i^T\mathbf{x}_j$, while for nonlinear decision functions nonlinear kernel functions (like the RBF) are employed.

B. Multi-class SVM classifier

In order to define K hyperplanes described by the vectors $\mathbf{w}_k$, $k = 1, \ldots, K$ the following optimization problem was proposed in [3]:

$$\underset{\mathbf{w}_k, b_k}{min}\ \sum_{k=1}^{K}\frac{1}{2}\mathbf{w}_k^T\mathbf{w}_k + c\sum_{i=1}^{N}\sum_{k \neq l_i}\xi_i^k, \tag{4}$$

$$\mathbf{w}_{l_i}^T\mathbf{x}_i + b_{l_i} \geq \mathbf{w}_k^T\mathbf{x}_i + b_k + 2 - \xi_i^k, \xi_i^k \geq 0, i = 1, \ldots, N, k \neq l_i, \tag{5}$$

which is equivalent to the following quadratic problem:

$$\underset{\boldsymbol{\alpha}_k}{max}\sum_{k,i,j=1}^{K,N,N}\left(\alpha_i^k\alpha_j^{l_i} - \frac{1}{2}\alpha_i^k\alpha_j^k - \frac{1}{2}\alpha_i\alpha_j c_j^{l_i}\right)\mathbf{x}_i^T\mathbf{x}_j + 2\sum_{k,i=1}^{K,N}\alpha_i^k, \tag{6}$$

$$s.t.:\ \ \sum_{i=1}^{N}\alpha_i^k = \sum_{i=1}^{N}c_i^k\alpha_i, \quad k = 1, \ldots, K, \tag{7}$$

$$0 \leq \alpha_i^k \leq c,\ \ \alpha_i^{l_i} = 0,\ \ i = 1, \ldots, N,\ \ k \neq l_i. \tag{8}$$

In the above, $\alpha_i^k, i = 1, \ldots, N, k = 1, \ldots, K$ are the Lagrange multipliers and c_i^k, α_i are variables defined as:

$$\alpha_i = \sum_{k=1}^{K}\alpha_i^k, \quad c_i^k = 1,\ if\ l_i = k\ and\ c_i^k = 0,\ if\ l_i \neq k. \tag{9}$$

The kernel trick can also be exploited in order to define nonlinear decision functions for multi-class classification.

C. Graph Embedding

The Graph Embedding [10] assumes that the training data $\mathbf{x}_i$, $i = 1, \ldots, N$ form an undirected weighted graph $\mathcal{G} = \{\mathbf{X}, \mathbf{V}\}$, where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$ and $\mathbf{V} \in \mathbb{R}^{N \times N}$ is a similarity matrix whose elements denote the relationships between the graph vertices $\mathbf{x}_i$. Furthermore, a penalty graph $\mathcal{G}^p = \{\mathbf{X}, \mathbf{V}^p\}$ can be defined, whose weight matrix $\mathbf{V}^p \in \mathbb{R}^{N \times N}$ penalizes specific relationships between the graph vertices $\mathbf{x}_i$. For example, the graph weights used in Marginal Discriminant Analysis [10] are defined by:

$$V_{ij} = \begin{cases} 1, & l_i = l_j\ and\ \mathbf{x}_j \in \mathcal{N}_i, \\ 1, & l_i = l_j\ and\ \mathbf{x}_i \in \mathcal{N}_j, \\ 0, & otherwise, \end{cases} \tag{10}$$

$$V_{ij}^p = \begin{cases} 1, & l_i \neq l_j\ and\ \mathbf{x}_j \in \mathcal{N}_i, \\ 1, & l_i \neq l_j\ and\ \mathbf{x}_i \in \mathcal{N}_j, \\ 0, & otherwise. \end{cases} \tag{11}$$

$\mathcal{N}_i$ denotes the neighborhood of sample $\mathbf{x}_i$.

Data $\mathbf{x}_i \in \mathbb{R}^D$ are projected to a low-dimensional feature space $\mathbb{R}^d$, $d < D$, by applying a linear transformation optimizing the following criterion:

$$\mathbf{W}^* = \underset{tr(\mathbf{W}^T\mathbf{X}\mathbf{L}_p\mathbf{X}^T\mathbf{W}) = c}{argmin}\ tr\left(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W}\right), \tag{12}$$

where $tr(\cdot)$ is the trace operator and $\mathbf{L} \in \mathbb{R}^{N \times N}$ is the so-called graph Laplacian matrix defined as $\mathbf{L} = \mathbf{D} - \mathbf{V}$, where $\mathbf{D}$ is the diagonal degree matrix having elements $D_{ii} = \sum_{j=1}^{N}V_{ij}$. $\mathbf{L}_p \in \mathbb{R}^{N \times N}$ is the graph Laplacian matrix of $\mathcal{G}^p$, that is $\mathbf{L}_p = \mathbf{D}^p - \mathbf{V}^p$.

The columns of the transformation matrix $\mathbf{W}$ are formed by the eigenvectors of the matrix $\mathbf{S} = \left(\mathbf{X}\mathbf{L}_p\mathbf{X}^T\right)^{-1}\left(\mathbf{X}\mathbf{L}\mathbf{X}^T\right)$ corresponding to the d minimal eigenvalues λ_i.

D. SVM classifiers exploiting intrinsic graphs

In order to exploit intrinsic graph structures in binary SVM-based classification, the following optimization problem was proposed in [7]:

$$\underset{\mathbf{w},b}{min}\ \frac{1}{2}\mathbf{w}^T\tilde{\mathbf{S}}_i\mathbf{w} + c\sum_{i=1}^{N}\xi_i, \tag{13}$$

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i,\ \ \xi_i \geq 0, \quad i = 1, \ldots, N, \tag{14}$$

where $\tilde{\mathbf{S}}_i \in \mathbb{R}^{D \times D}$ is a matrix describing the properties of the training data that are subject to minimization and is defined as $\tilde{\mathbf{S}}_i = \mathbf{I} + \lambda\mathbf{X}\mathbf{L}\mathbf{X}^T$. L is the Laplacian matrix of the intrinsic graph defined under the Graph Embedding framework [10] and $\lambda > 0$ is a regularization parameter. It should be noted here that the optimization problem in (13) is a generalization of the optimization problems proposed in [4], [5], where only the within-class scatter of the training data $\mathbf{S}_w = \sum_{k=1}^{K}\sum_{i,l_i=k}(\mathbf{x}_i - \mathbf{m}_k)(\mathbf{x}_i - \mathbf{m}_k)^T$ is considered.

For multi-class classification, the following optimization problem was proposed in [8], [9]:

$$\underset{\mathbf{w}_k, b_k}{min}\ \sum_{k=1}^{K}\frac{1}{2}\mathbf{w}_k^T\mathbf{S}_w\mathbf{w}_k + c\sum_{i=1}^{N}\sum_{k \neq l_i}\xi_i^k, \tag{15}$$

$$\mathbf{w}_{l_i}^T\mathbf{x}_i+b_{l_i} \geq \mathbf{w}_k^T\mathbf{x}_i+b_k+2-\xi_i^k, \xi_i^k \geq 0, i=1,\dots,N, k \neq l_i. \quad (16)$$

For the extension of the multi-class optimization problem (15) to non-linear decision functions, two-step processes were proposed. Specifically, in [8], the training data $\mathbf{x}_i$ are projected to $\tilde{\mathbf{x}}_i$ by using $\tilde{\mathbf{x}}_i = \mathbf{S}_w^{-\frac{1}{2}}\mathbf{x}_i$ and the nonlinear version of standard multi-class SVM (4) is subsequently solved by using $\tilde{\mathbf{x}}_i$. In [9], the training data $\mathbf{x}_i$ are non-linearly mapped to the feature space determined by applying kernel PCA, and subsequently the linear optimization problem (15) is solved in that space.

III. Proposed Method

In order to exploit intra-class and between-class information encoded in kNN graphs (10) and (11) within a multi-class SVM formulation, we solve the following optimization problem:

$$\underset{\mathbf{w}_k,b_k}{min} \; \sum_{k=1}^{K} \frac{1}{2}\mathbf{w}_k^T\mathbf{w}_k + c\sum_{i=1}^{N}\sum_{k\neq l_i}\xi_i^k + \sum_{k=1}^{K}\frac{\lambda}{2}\mathbf{w}_k^T\mathbf{S}\mathbf{w}_k \quad (17)$$

subject to the constraints:

$$\mathbf{w}_{l_i}^T\mathbf{x}_i+b_{l_i} \geq \mathbf{w}_k^T\mathbf{x}_i+b_k+2-\xi_i^k, \xi_i^k \geq 0, i=1,\dots,N, k \neq l_i. \quad (18)$$

In (17), $\mathbf{S}$ is a matrix expressing a combination of intrinsic and penalty training data relationships, as described in Section II-C, i.e. $\mathbf{S} = \left(\mathbf{X}\mathbf{L}_p\mathbf{X}^T\right)^{-1}\left(\mathbf{X}\mathbf{L}\mathbf{X}^T\right)$. The equivalent dual optimization problem to (17) subject to the constraints in (18), is the following:

$$\begin{aligned}\mathcal{D} \;&=\; \frac{1}{2}\sum_{k=1}^{K}\mathbf{w}_k^T(\mathbf{I}+\lambda\mathbf{S})\mathbf{w}_k + c\sum_{i=1}^{N}\sum_{k\neq l_i}\xi_i^k - \sum_{k=1}^{K}\sum_{i=1}^{N}\beta_i^k\xi_i^k \\ &-\; \sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_i^k\left[(\mathbf{w}_{l_i}-\mathbf{w}_k)^T\mathbf{x}_i + b_{l_i} - b_k - 2 + \xi_i^k\right] (19)\end{aligned}$$

with the constraints:

$$\alpha_i^k \geq 0,\;\; \beta_i^k \geq 0,\;\; \xi_i^k \geq 0, \quad i=1,\dots,N,\;\; k \neq l_i. \quad (20)$$

By determining the saddle points of $\mathcal{D}$ with respect to $\mathbf{w}_k$, b_k and ξ_i^k, we obtain:

$$\begin{aligned}\nabla\mathcal{D}|_{\mathbf{w}_k^*} \;&=\; 0 \Rightarrow \mathbf{w}_k = (\mathbf{I}+\lambda\mathbf{S})^{-1}\sum_{i=1}^{N}(\alpha_i c_i^k - \alpha_i^k)\mathbf{x}_i, (21) \\ \nabla\mathcal{D}|_{b_k^*} \;&=\; 0 \Rightarrow \sum_{i=1}^{N}\alpha_i^k - \sum_{i=1}^{N}\alpha_i c_i^k = 0, \quad (22) \\ \nabla\mathcal{D}|_{x_i^k{}^*} \;&=\; 0 \Rightarrow c = \alpha_i^k + \beta_i^k, \quad (23)\end{aligned}$$

with the constraints $0 \leq \alpha_i^k \leq c$. Substituting (21), (22) and (23) in (19), we obtain:

$$\mathcal{D} = \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N} q_{ij}^k\mathbf{x}_i^T(\mathbf{I}+\lambda\mathbf{S})^{-1}\mathbf{x}_j + 2\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_i^k, \quad (24)$$

$$q_{ij}^k = \left(\alpha_i^k\alpha_j^{l_i} - \frac{1}{2}\alpha_i^k\alpha_j^k - \frac{1}{2}\alpha_i\alpha_j c_j^{l_i}\right), \quad (25)$$

with the constraints:

$$\sum_{i=1}^{N}\alpha_i^k = \sum_{i=1}^{N}c_i^k\alpha_i, \quad k=1,\dots,K, \quad (26)$$

$$0 \leq \alpha_i^k \leq c, \;\; \alpha_i^{l_i} = 0, \quad i=1,\dots,N, \;\; k \neq l_i, \quad (27)$$

which is a quadratic optimization problem in terms of $\boldsymbol{\alpha}$.

In order to obtain non-linear decision functions, we assume that there is a non-linear function $\phi(\cdot) : \mathbf{x}_i \in \mathbb{R}^D \rightarrow \phi(\mathbf{x}_i) \in \mathcal{F}$ mapping the data from the input space to the kernel space. Let us denote by $\boldsymbol{\Phi} = [\phi(\mathbf{x}_1, \dots, \phi(\mathbf{x}_N)] \in \mathbb{R}^{|\mathcal{F}|\times N}$ a matrix containing the training data representations in $\mathcal{F}$. The kernel matrix can be defined as $\mathbf{K} = \boldsymbol{\Phi}^T\boldsymbol{\Phi}$. Based on the Representer Theorem, we can express the decision functions as linear combinations of $\boldsymbol{\Phi}$, i.e.:

$$\mathbf{w}_k = \sum_{i=1}^{N}\gamma_i^k\phi(\mathbf{x}_i) = \boldsymbol{\Phi}\boldsymbol{\gamma}_k, \;\; k=1,\dots,K, \quad (28)$$

where $\boldsymbol{\gamma}_k \in \mathbb{R}^N$ is a vector containing the reconstruction weights of $\mathbf{w}_k$ with respect to $\phi(\mathbf{x}_i)$, $i=1,\dots,N$. The matrix $\mathbf{S} \in \mathbb{R}^{|\mathcal{F}|\times|\mathcal{F}|}$ can be expressed as follows:

$$\begin{aligned}\mathbf{S} \;&=\; \tilde{\mathbf{S}}_p^{-1}\mathbf{S}_i = (\boldsymbol{\Phi}\mathbf{L}_p\boldsymbol{\Phi}^T + r\mathbf{I})^{-1}(\boldsymbol{\Phi}\mathbf{L}\boldsymbol{\Phi}^T) \\ &=\; \frac{1}{r}\boldsymbol{\Phi}\mathbf{L}\boldsymbol{\Phi}^T - \frac{1}{r^2}\boldsymbol{\Phi}(\mathbf{L}_p^{-1} + \frac{1}{r}\mathbf{K})^{-1}\mathbf{K}\mathbf{L}\boldsymbol{\Phi}^T \quad (29)\end{aligned}$$

The equivalent to (17) subject to the constraints in (18) dual optimization problem can now be expressed as follows:

$$\begin{aligned}\mathcal{D} \;&=\; \frac{1}{2}\sum_{k=1}^{K}\boldsymbol{\gamma}_k^T\boldsymbol{\Theta}\boldsymbol{\gamma}_k + c\sum_{i=1}^{N}\sum_{k\neq l_i}\xi_i^k - \sum_{k=1}^{K}\sum_{i=1}^{N}\beta_i^k\xi_i^k \\ &-\; \sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_i^k\left[(\boldsymbol{\gamma}_{l_i}-\boldsymbol{\gamma}_k)^T\mathbf{k}_i + b_{l_i} - b_k - 2 + \xi_i^k\right], (30)\end{aligned}$$

with the constraints:

$$\alpha_i^k \geq 0,\;\; \beta_i^k \geq 0,\;\; \xi_i^k \geq 0, \quad i=1,\dots,N,\;\; k \neq l_i. \quad (31)$$

In (30), $\boldsymbol{\Theta} = \mathbf{K} + \frac{\lambda}{r}\mathbf{K}\mathbf{L}\mathbf{K} - \frac{\lambda}{r^2}\mathbf{K}(\mathbf{L}_p^{-1} + \frac{1}{r}\mathbf{K})^{-1}\mathbf{K}\mathbf{L}\mathbf{K}$ and $\mathbf{k}_i \in \mathbb{R}^N$ is a vector containing the values $\mathbf{k}_{i,j} = \phi(\mathbf{x}_j)^T\phi(\mathbf{x}_i)$.

By determining the saddle points of $\mathcal{D}$ with respect to $\boldsymbol{\gamma}_k$, b_k and ξ_i^k, we obtain:

$$\begin{aligned}\nabla\mathcal{D}|_{\boldsymbol{\gamma}_k^*} \;&=\; 0 \Rightarrow \boldsymbol{\gamma}_k = \boldsymbol{\Theta}^{-1}\sum_{i=1}^{N}(\alpha_i c_i^k - \alpha_i^k)\mathbf{k}_i, \quad (32) \\ \nabla\mathcal{D}|_{b_k^*} \;&=\; 0 \Rightarrow \sum_{i=1}^{N}\alpha_i^k - \sum_{i=1}^{N}\alpha_i c_i^k = 0, \quad (33) \\ \nabla\mathcal{D}|_{x_i^k{}^*} \;&=\; 0 \Rightarrow c = \alpha_i^k + \beta_i^k, \quad (34)\end{aligned}$$

with the constraints $0 \leq \alpha_i^k \leq c$.

Substituting (32), (33) and (34) in (30), we obtain:

$$\mathcal{D} = \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N} q_{ij}^k\mathbf{k}_i^T(\boldsymbol{\Theta}^{-1})^T\mathbf{k}_j + 2\sum_{k=1}^{K}\sum_{i=1}^{N}\alpha_i^k, \quad (35)$$

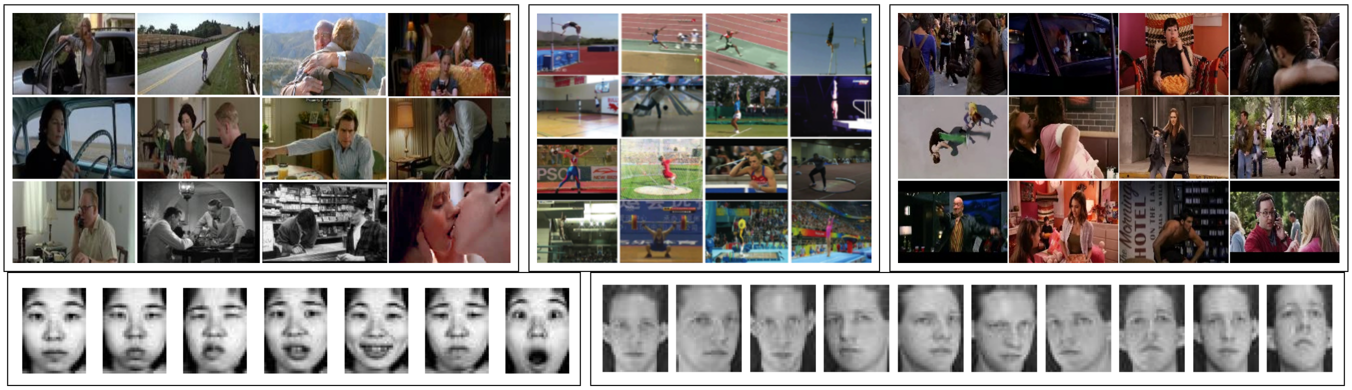

Fig. 1. (Top) Video frames from Hollywood2, Olympic sports and Hollywood 3D action datasets. Bottom Facial images from the JAFFE and ORLS datasets

$$q_{ij}^k = \left(\alpha_i^k \alpha_j^{l_i} - \frac{1}{2}\alpha_i^k \alpha_j^k - \frac{1}{2}\alpha_i \alpha_j c_j^{l_i} \right), \qquad (36)$$

with the constraints:

$$\sum_{i=1}^{N} \alpha_i^k = \sum_{i=1}^{N} c_i^k \alpha_i, \quad k = 1, \dots, K, \qquad (37)$$

$$0 \le \alpha_i^k \le c, \ \alpha_i^{l_i} = 0, \quad i = 1, \dots, N, \ k \neq l_i, \qquad (38)$$

which is a quadratic optimization problem in terms of $\boldsymbol{\alpha}$.

By comparing (35) and (6), we can observe that, as in the linear case, the two optimization problems are similar. Specifically, the solution of (35) is equivalent to the solution of (6), by exploiting the modified kernel matrix $\tilde{\mathbf{K}} = \mathbf{K}(\mathbf{\Theta}^{-1})^T\mathbf{K} = \left([\mathbf{I} + \frac{\lambda}{r}\mathbf{K}\mathbf{L} - \frac{\lambda}{r^2}\mathbf{K}(\mathbf{L}_p^{-1} + \frac{1}{r}\mathbf{K})^{-1}\mathbf{K}\mathbf{L}]^{-1}\right)^T \mathbf{K}$.

IV. Experiments

We evaluated the performance of the proposed classifier in human action recognition and facial image classification problems. In human action recognition, we employed three benchmark datasets, i.e. the Hollywood2 [11], the Olympic sports [12] and the recently introduced Hollywood 3D [13] datasets. For facial image classification we employed the JAFFE [14] and ORL [15] datasets. Example action video frames and facial images from all datasets are illustrated in Figure 1. In all our experiments we compare the performance of the proposed method with that of SVM, MCVSVM [5], [9] and Graph Embedded SVM using (10) [7].

In our first set of experiments, we applied the competing methods on human action recognition. We used the state-of-the-art video representation proposed in [16] that describes a video by using HOG, HOF, MBHx, MBHy and (normalized) Trajectory descriptors evaluated on the trajectories of densely sampled interest points. After descriptor calculation, the video is represented by $V = 5$ Bag-of-Words (BoW)-based representations, i.e. one BoW-based representation per descriptor type. We follow [16] and use 4000 codewords for each BoW representation. Classification is performed by employing the RBF-χ^2 kernel which has been found to outperform other choices for BoW-based representations [18]. Different descriptor types are combined by following a multi-channel approach [17]. In all our experiments we have used 5-NN graphs. In all action recognition datasets, performance is evaluated by computing the average precision (AP) for each action class and reporting the mean AP over all classes (mAP). This is due to the fact that a video may depict more than one actions.

TABLE I
Performance (mAP) in human action recognition.

	Hollywood2	Olympic sports	Hollywood 3D
SVM	61.41%	82.77%	29.45%
Method [9]	65.98%	84.94%	30.31%
Method [5]	65.74%	84.86%	30.29%
Method [7]	66.06%	84.96%	31.11%
Proposed method	**67.48**%	**87.82**%	**33.13**%

The performance achieved by the competing methods is illustrated in Table I. The proposed method exploiting both intrinsic and penalty kNN graphs achieves higher performance when compared to standard SVM and SVM methods exploiting only intrinsic class geometric information. In Table II, we also compare the performance obtained by applying the proposed classifiers on the BoW-based video representation exploiting the Improved Dense Trajectory-based video description with that of some recently proposed action recognition methods. As can be seen, this video description and classifier combination provides very good performance, which is comparable with other (state-of-the-art) approaches. It is worth noting here that most of the methods listed in Table II employ standard (binary) SVM classification. Thus, we expect that the application of the proposed classifiers would enhance their performance.

In our second set of experiments we have applied multi-class classification in two facial image datasets, i.e. the JAFFE and ORL datasets. We have applied the five-fold cross validation by taking into account the labels of the data. That is, each fold is formed by 20% of the facial images of each class. In each cross-validation round, one fold is used as test set, while the remaining folds form the training set. Five cross-validation rounds are conducted, one per each test fold index. The classification accuracy of each classifier is subsequently measured for that experiment. We perform five experiments

TABLE II
COMPARISON OF OUR RESULTS WITH SOME STATE-OF-THE-ART METHODS ON THE HOLLYWOOD2, OLYMPIC SPORTS AND HOLLYWOOD 3D DATASETS.

	Hollywood2	Olympic sports	Hollywood 3D
Method [19]	-	-	28.7%
Method [20]	-	-	29.28%
Method [21]	-	-	30.52%
Method [21]	-	-	30.52%
Method [22]	-	-	**36.9**%
Method [23]	45.8%	-	-
Method [24]	59.5%	80.6%	-
Method [25]	62.5%	85.49%	-
Method [26]	-	85.5%	-
Method [27]	63.3%	**89**%	-
Method [28]	61.69%	**88.89**%	-
Method [29]	62.5%	**89.74**%	31.79%
Method [16] - BoWs	62.2%	83.3%	-
Method [16] - FVs	64.3%	**91.1**%	-
Proposed method	**67.5**%	**87.82**%	**33.13**%

TABLE III
PERFORMANCE (CR) IN FACIAL IMAGE CLASSIFICATION PROBLEMS.

	JAFFE	ORL
SVM	82.38%	92.25%
Method [9]	84.29%	94.5%
Method [5]	82.38%	94.5%
Method [7]	84.29%	94.25%
Proposed method	**87.62**%	**98.25**%

for each database and calculate the mean classification rate in order to measure the performance of the various classifiers. We apply non-linear classification using the RBF kernel function. The performance of the competing methods is illustrated in Table III. Compared to the standard SVM and SVM methods exploiting only geometric data relationships described in intrinsic graphs, the proposed method exploiting combined information appearing in both intrinsic and penalty graphs provides higher performance.

V. CONCLUSIONS

In this paper, we described a graph-regularized multi-class SVM classifier exploiting kNN graphs encoding intra-class and between-class data relationships. We have provided direct solutions for the optimization problem solved in both linear and non-linear cases and compared the performance of the proposed classifier with related classification methods in human action recognition and facial image classification problems, where it outperformed relating classification approaches.

REFERENCES

[1] V. Vapnik, *Statistical Learning Theory*, Wiley-Interscience, 2006.

[2] A.M. Martinez and A.C. Kak, *An overview of ensemble methods for binary clasifiers in multi-class problems: Experimental study on One-vs-One and One-vs-All scemes*, Pattern Recognition, vol. 44, pp. 1761-1776, 2011.

[3] J. Weston and C. Watkins, *Multi-class Support Vector Machines*, European Symposium of Artificial Neural Networks, 1999.

[4] A. Tefas, C. Kotropoulos and I. Pitas, *Using Support Vector Machines to enhance the performance of Elastic Graph Matching for frontal face authentication*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 7, pp. 735-746, 2001.

[5] S. Zafeiriou, A. Tefas and I. Pitas, *Minimum Class Variance Support Vector Machines*, IEEE Transactions on Image Processing, vol. 16, no. 10, pp. 2551-2564, 2007.

[6] G. Orfanidis and A. Tefas, *Exploiting subclass information in Support Vector Machines*, IEEE International Conference on Pattern Recognition, 2012.

[7] G. Arvanitidis and A. Tefas, *Exploiting Graph Embedding in Support Vector Machines*, IEEE International Workshop on Machine Learning for Signal Processing, 2012.

[8] I. Kotsia and I. Pitas, *Facial expression recognition in image sequences using geometric deformation features and Support Vector Machines*, IEEE Transactions on Image Processing, vol. 16, no. 1, pp. 172-187, 2007.

[9] I. Kotsia, S. Zafeiriou and I. Pitas, *A Novel Class of Multiclass Classifiers based on the Minimization of Within-Class-Variance*, IEEE Transactions on Neural Networks, vol. 20, no. 1, pp. 14-34, 2009.

[10] S. Yan, D. Xu, B. Zhang and H.J. Zhang, *Graph Embedding and Extensions: A General Framework for Dimensionality Reduction*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 1, pp. 40-51, 2007.

[11] M. Marszalek, I. Laptev and C. Schmid, *Actions in context*, Computer Vision and Pattern Recognition, 2009.

[12] J.C. Niebles, C.W. Chend and L. Fei-Fei, *Modeling Temporal Structure of Decomposable Mition Segemnts for Activity Classification*, European Conference on Computer Vision, 2010.

[13] S. Hadfield and R. Bowden, *Hollywood 3D: Recognizing Actions in 3D Natural Scenes*, Computer Vision and Pattern Recognition, 2013.

[14] M. Lyons, S. Akamatsu, M. Kamachi and J. Gyoba, *Coding facial expressions with Gabor wavelets*, IEEE Int. Conf. Automatic Face and Gest. Recogn., 1998.

[15] F. Samaria and A. Harter, *Parameterisation of a stochastic model for human face identification*, IEEE Workshop Appl. Comp. Vision, 1994.

[16] H. Wang and C. Schmid, *Action Recognition with Improved Trajectories*, Computer Vision and Pattern Recognition, 2013.

[17] 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.

[18] I. Laptev, M. Marszalek, C. Schmid and B. Rozenfeld, *Learning realistic human actions from movies*, Computer Vision and Pattern Recognition, 2008.

[19] I. Mademlis, A. Iosifidis, A. Tefas and I. Pitas, *Stereoscopic Video Description for Human Action Recognition*, IEEE Symp. Series Comput. Intell., 2014.

[20] A. Iosifidis, A. Tefas and I. Pitas, *Multi-view Regularized Extreme Learning Machine for Human Action Recognition*, Art. Intell.: Methods and Applications, 2014.

[21] A. Iosifidis, A. Tefas and I. Pitas, *Human action recognition in stereoscopic videos based on bag of features and disparity pyramids*, European Signal Processing Conference, 2014.

[22] S. Hadfield, K. Lebeda and R. Bowden, *Natural action recognition using invariant 3D motion encoding*, European Conference on Computer Vision, 2014.

[23] A. Iosifidis, A. Tefas and I. Pitas, *Discriminant Bag of Words based Representation for Human Action Recognition*, Pattern Recognition Letters, vol. 49, pp. 185-192, 2014.

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

[25] A. Iosifidis, A. Tefas and I. Pitas, *Distance-based Human Action Recognition using optimized class representations*, Neurocomputing, vol. 161, pp. 47-55, 2015.

[26] A. Gaidon, Z. Harchaoui and C. Schmid, *Activity representation with motion hierarchies*, International Journal of Computer Vision, vol. 107, no. 3, pp. 219-238, 2014.

[27] D. Oneata, J. Verbeek and C. Schmid, *Action and event recognition with fisher vectors on a compact feature set*, International Conference on Computer Vision, 2013.

[28] A. Iosifidis, A. Tefas and I. Pitas, *Class-specific Reference Discriminant Analysis with application in Human Behaviour Analysis*, IEEE Transactions on Human-Machine Systems, vol. 45, no. 3, 315-326, 2015.

[29] A. Iosifidis, A. Tefas and I. Pitas, *Graph Embedded Extreme Learning Machine*, IEEE Transactions on Cybernetics, vol. 46, no. 1, 311-324, 2016.