Exploiting local and global geometric data relationships in Support Vector Data Description

Vasileios Mygdalis[†], Anastasios Tefas[†] and Ioannis Pitas[†⋆]
[†] Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, 54124, Greece
[⋆]Department of Electical and Electronic Engineering, University of Bristol, UK

***Abstract*—In this paper, we describe a one-class classification method based on Support Vector Data Description, which exploits multiple graph structures in its optimization process. We derive in a generic solution which can be employed for supervised one-class classification tasks. The devised method can produce linear or non-linear decision functions, depending on the adopted kernel function. In our experiments, we simultaneously adopted two graphs that describe local and global geometric training data relationships, respectively. We evaluated the proposed classifier in publicly available datasets, where its performance compared favorably against closely related methods.**

I. Introduction

Support Vector Data Description (SVDD) is a support vector based one-class classifier, initially proposed in [1]. The training phase of this classifier involves determining the minimum bounding hypersphere which encloses the target class. Test patterns that fall inside this hypersphere belong to the target class, or considered as outliers, otherwise. The SVDD optimization problem attempts to minimize the radius of the hypersphere with respect to its center. Finally, the hyperpshere center is expressed as a linear combination of the determined support vectors. Applications of SVDD have been found in several fields, including hyperspectral image classification [2], [3], mechanical failure detection [4], biomedical data classification [5], face recognition [6], video summarization [7] and other outlier detection tasks [8], [9].

Despite being simple and effective, SVDD has drawbacks related to its training speed and classification accuracy. In terms of traing speed, a method that determines redundand training vectors has been proposed in [10], exploiting the fact that in support vector based classification, removing the non support vectors from the training set does not affect the classification model. Additionally, a method to determine the appropriate gaussian kernel parameter (i.e., the sigma) in a fast manner, have been proposed in [2]. Furthermore, methods to improve the SVDD classification accuracy have also been proposed. In [11], the authors perform the whitening transform in the training data. That is, they minimize the class variance within the SVDD optimization process. In [7], an additional clustering step is performed in the training data, in order to determine subclasses within the training class. The subclass information is thereby employed in the SVDD optimization process. In essence, both [7] and [11] employ additional optimization criteria for the SVDD, such that low-variance directions are overemphasized. In essence, instead of hypersphere, the solutions of [7] and [11] resemble hyperellipses that tightly enclose the target class.

Recently, a method that extends the SVDD in the context of semi-supervised learning have been proposed in [12]. To this end, relationships between labeled and unlabeled training patterns expressed in Nearest Neighbourhood (kNN) graph structures are employed in the SVDD optimization process. The corresponding semi-supervised one-class classification method based on One-Class Support Vector Machines have been proposed in [13]. Besides local information described in kNN graphs for the semi-supervised classification case, generic graph-type regularization methods have also been proposed for supervised classification [14], [15], [16], [17]. Depending on the adopted graph type (either kNN or fully connected graph), local or global data relationships are expressed in the optimization process of a multiclass classifier, leading to more accurate solutions. In order to obtain optimal solutions, it is important to select the appropriate graph type which describes the underlying data distribution accurately. As have been shown in multiclass classification problems [15], [16], optimal classification models are obtained when advanced graph settings are employed (i.e., both positive and penalty graphs, describing opposite data relationships), in the classifier optimization process. To the best of our knowledge, in the one-class classification case, employing advanced graph settings in the SVDD optimization process had never been attempted.

In this paper, we describe a one-class classification method based on SVDD, which can exploit multiple graph structures in its optimization process. Diverse information encoded in kNN or fully connected graphs can be employed simultaneously, so that the solution emphasizes on unified diverse criteria. The derived solution can be employed for supervised one-class classification tasks. The devised method produces linear or non-linear decision functions, depending on the adopted kernel function. From our derivations, the solution of the proposed method resembles the standard SVDD solution, in a regularized modified space. In order to evaluate the performance of the proposed classifier, we have employed publicly available datasets, related to human action recognition, as well as face recognition applications.

The rest of the paper is structured as follows. In Section II, we briefly describe the standard SVDD classifier. In Section III, we analytically describe the proposed classification method, which exploits local and global geometric data relationships in the SVDD optimization process. Next, in Section IV, we provide an extension of the proposed method, in order to exploit non-linear decision functions. Experimental results are presented in Section V, and finally, conclusions are drawn in Section VI.

II. Support Vector Data Description

Let the vectors $\boldsymbol{x}_i \in \mathbb{R}^D$, $i = 1, \ldots, N$ form the target class, from which we wish to generate a one-class classification model, by employing the SVDD method. SVDD aims at modeling this target class through a hypersphere, with center $\mathbf{a} \in \mathbb{R}^D$ and radius R, which encloses the training vectors onto a bounded, spherically shaped, area. The optimal hypersphere can be found by solving the following optimization problem:

$$\underset{R,\xi_i,\boldsymbol{a}}{\textit{minimize:}} \quad R^2 + c\sum_{i=1}^{N}\xi_i \tag{1}$$

$$\textit{subject to :} \quad \|\boldsymbol{x}_i - \boldsymbol{a}\|^2 \leq R^2 + \xi_i, \tag{2}$$

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

where ξ_i, $i = 1, \ldots, N$ are the slack variables and $c > 0$ is a free parameter that allows some training error (i.e., soft margin formulation), in order to increase the generalization performance. This optimization problem can be solved through Lagrange optimization, generating the following hypersphere center:

$$\boldsymbol{a} = \sum_{i=1}^{N}\gamma_i\boldsymbol{x}_i, \tag{4}$$

where γ_i is the Lagrange multiplier corresponding to each constraint (2). The hypersphere radius R can be calculated by using any support vector $\boldsymbol{x}_k$ whose coefficient satisfies $\lambda_k > 0$ [1], as follows:

$$R^2 = (\boldsymbol{x}_k \cdot \boldsymbol{x}_k) - \sum_{i=1}^{N}\gamma_i(\boldsymbol{x}_i, \boldsymbol{x}_k) - \sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i\gamma_j(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{5}$$

In order to obtain non-linear solutions, any kernel function $\kappa(\cdot,\cdot)$, which expresses data similarity in a space of increased dimensionality can be employed, leading to the following optimization problem:

$$\underset{\gamma_i}{\textit{maximize:}} \sum_{i=1}^{N}\gamma_i \;\kappa(\boldsymbol{x}_i\boldsymbol{x}_i) - \sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i\gamma_j\kappa(\boldsymbol{x}_i\boldsymbol{x}_j) \tag{6}$$

$$\textit{subject to :} 0 \leq \gamma_i \leq c, \quad \sum_{i=1}^{N}\gamma_i = 1, \tag{7}$$

where the contraints in (7) ensure that the corresponding constraints in (3) are also satisfied.

For a given test sample $\boldsymbol{x} \in \mathbb{R}^D$, we decide that it belongs to the target class, if it satisfies the following inequality:

$$\kappa(\boldsymbol{x}, \boldsymbol{x}) - 2\sum_{i=1}^{N}\gamma_i\kappa(\boldsymbol{x}, \boldsymbol{x}_i) + \sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i\gamma_j\kappa(\boldsymbol{x}_i, \mathbf{x}_j) \leqslant R^2. \tag{8}$$

III. SVDD exploiting local and global geometric data information

Let $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$ be a datamatrix consisting of vectors $\boldsymbol{x}_i \in \mathbb{R}^D$, corresponding to the training data samples. The geometry of this dataset can be expressed with an undirected weighted graph, i.e., $\mathcal{G} = \{\boldsymbol{X}, \boldsymbol{W}\}$, where the graph vertex set is formed with the training data and $\boldsymbol{W}$ is the graph weight matrix.

Global geometric data relationships between the training data can be expressed using a fully-connected graph. As have been shown in [18], a fully connected graph with equal weights, i.e.,:

$$W_{ij}^G = 1/N, i, j = 1, \ldots, N, \tag{9}$$

can be employed to express the total scatter of the class, as follows:

$$\boldsymbol{C} = \frac{1}{N}\sum_{i}^{N}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^T = \frac{1}{N}\boldsymbol{X}(\boldsymbol{I} - \boldsymbol{W}^G)\boldsymbol{X}^T, \tag{10}$$

where $\bar{\boldsymbol{x}}$ is the mean vector and $\boldsymbol{I}$ is a $N \times N$ identity matrix. If we observe (10) from a graph embedding point of view, the identity matrix can be considered as the corresponding degree matrix $\mathbf{D}^G = \boldsymbol{I}$, where $D_{ii} = \sum_{i\neq j}^{N} W_{ij}^G, j = 1, \ldots, N$.

Additionally, global geometric data relationships can also be expressed with a fully connected weighted graph, having graph weights initiated with a heat kernel function, i.e.:

$$W_{ij}^G = exp\left(-\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2}{2\sigma^2}\right), \tag{11}$$

where σ is a free parameter that scales the Euclidean distances between the graph vertices $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

In order to express *local geometric data relationships*, we can construct kNN graphs. That is, we define a neighborhood $\mathcal{N}_i$ for each vertex, containing the k most similar vectors to $\boldsymbol{x}_i$. Then, the corresponding weight matrix is given by:

$$W_{ij}^L = \begin{cases} W_{ij}^G, & \text{if } \boldsymbol{x}_j \in \mathcal{N}_i \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Inspired by dimensionality reduction applications [18], if would like to create a representation that includes both local and global geometric data information for our dataset, we should minimize the following objective function:

$$\begin{aligned}\mathcal{F} =& \mu\sum_{ij}||\boldsymbol{x}_i - \boldsymbol{x}_j||^2 W_{ij}^G + (1-\mu)\sum_{ij}||\boldsymbol{x}_i - \boldsymbol{x}_j||^2 W_{ij}^L = \\ =& 2\boldsymbol{X}(\mu\boldsymbol{L}_G + (1-\mu)\boldsymbol{L}_L)\boldsymbol{X}^T = 2\boldsymbol{M}, \end{aligned} \tag{13}$$

where $\boldsymbol{L}_G \in \mathbb{R}^{N\times N}$ is the Laplacian matrix of the global geometric data relationships graph, $\boldsymbol{L}_L \in \mathbb{R}^{N\times N}$ the Laplacian matrix describing the local geometric data relationships and $\mu \in (0, 1)$ is a weighting parameter, in order to balance the adopted information contained in the two graphs. In order to extend the proposed method to exploit more graphs, i.e., M graphs, we should work as follows. First, in order to control the membership of each graph in the process, we define a parameter vector $\boldsymbol{\mu} \in \mathbb{R}^M$, such that $\sum_{i=1}^{M}\mu_i = 1$.

Next, we calculate the weighted sum of each graph Laplacian matrices. Finally, the optimal parameters μ_i can be determined by employing multiple kernel learning techniques [19].

In the supervised classification case, we wish to find the optimal hypersphere in a regularized space, where the training data are enclosed and geometric data relationships are expressed at the same time. To this end, we employ the matrix $\boldsymbol{M} \in \mathbb{R}^{D \times D}$, which encodes the local and global geometric data relationships, in the proposed optimization problem:

$$\underset{R,\xi_i,\boldsymbol{a}}{\textit{minimize:}} \quad R^2 + c\sum_{i=1}^{N} \xi_i \tag{14}$$

$$\textit{subject to :} \quad (\boldsymbol{x}_i - \boldsymbol{a})^T \boldsymbol{M}^{-1} (\boldsymbol{x}_i - \boldsymbol{a}) \leq R^2 + \xi_i, \tag{15}$$

$$\xi_i \geq 0, \qquad i = 1, \ldots, N, \tag{16}$$

where $\boldsymbol{a}$ is the hypersphere center, R is the hypersphere radius, ξ_i are the slack variables and c is a trade-off parameter between training error and generalization performance. By employing a vector $\boldsymbol{u} = \boldsymbol{M}^{\frac{1}{2}}\boldsymbol{a}$, the optimization problem can be solved by determining the saddle points of the Lagrangian:

$$\mathcal{L} = R^2 + c\sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \beta_i \xi_i - \\ - \sum_{i=1}^{N} \gamma_i \left(R^2 + \xi_i - \|\boldsymbol{M}^{-\frac{1}{2}}\boldsymbol{x}_i - \boldsymbol{u}\|^2 \right), \tag{17}$$

which lead to the following optimality conditions:

$$\frac{\vartheta \mathcal{L}}{\vartheta R} = 0 \Rightarrow \sum_{i=1}^{N} \lambda_i = 1., \tag{18}$$

$$\frac{\vartheta \mathcal{L}}{\vartheta \xi_i} = 0 \Rightarrow \beta_i = c - \gamma_i, \tag{19}$$

$$\frac{\vartheta \mathcal{L}}{\vartheta \boldsymbol{u}} = 0 \Rightarrow \boldsymbol{u} = \sum_{i=1}^{N} \gamma_i \boldsymbol{M}^{-\frac{1}{2}} \boldsymbol{x}_i. \tag{20}$$

The condition (19) can always be met if we demand $0 \leq \gamma_i \leq c$, thus the Lagrange multipliers β_i can be removed. From (20), the hypersphere center $\boldsymbol{a}$ can be found as follows:

$$\boldsymbol{a} = \boldsymbol{M}^{-1}\boldsymbol{X}\boldsymbol{\gamma}, \tag{21}$$

where $\boldsymbol{\gamma} \in \mathbb{R}^N$ is a vector containing the Lagrange multipliers. Every training pattern $\boldsymbol{x}_i$ which satisfies (15) (i.e., $\xi_i = 0$), falls inside the hypersphere and, thus, its corresponding Lagrange multiplier is equal to zero.

In any other case, $\gamma_i > 0$ and $\boldsymbol{x}_i$ is a support vector. The optimal radius can be recovered from any support vector $\boldsymbol{x}_k$ as follows:

$$R^2 = \|\boldsymbol{x}_k - \boldsymbol{a}\|^2 = \|\boldsymbol{x}_k - \boldsymbol{M}^{-1}\boldsymbol{X}\boldsymbol{\gamma}\|^2. \tag{22}$$

Having calculated the optimal center and radius, in order to make a decision whether a test pattern $\boldsymbol{x} \in \mathbb{R}^D$ falls inside the hypersphere, we calculate the following decision value:

$$f(\boldsymbol{x}) = R^2 - \|\boldsymbol{x} - \boldsymbol{a}\|^2, \tag{23}$$

where the test pattern is classified to the target class when $f(\boldsymbol{x}) \geq 0$, or considered an outlier otherwise.

By expressing the radius and the center in terms of support vectors, using the equations (21) and (22), we obtain the following solution:

$$f(\boldsymbol{x}) = \|\boldsymbol{x}_k - \boldsymbol{M}^{-1}\boldsymbol{X}\boldsymbol{\gamma}\|^2 - \|\boldsymbol{x} - \boldsymbol{M}^{-1}\boldsymbol{X}\boldsymbol{\gamma}\|^2. \tag{24}$$

Next, in order to obtain the support vector coefficient vector $\boldsymbol{\gamma}$, we reformulate the Lagrangian defined in (17), exploiting (18), (19) and (20), as follows:

$$\mathcal{L} = \sum_{i=1}^{N} \gamma_i \boldsymbol{x}_i \boldsymbol{M}^{-1} \boldsymbol{x}_i - \sum_{i=1}^{N}\sum_{j=1}^{N} \gamma_i \gamma_j \boldsymbol{x}_i \boldsymbol{M}^{-1} \boldsymbol{x}_j \tag{25}$$

Finally, the solution is obtained by solving the following optimization problem:

$$\underset{\gamma_i}{\textit{maximize:}} \sum_{i=1}^{N} \gamma_i \boldsymbol{x}_i \boldsymbol{M}^{-1} \boldsymbol{x}_i - \sum_{i=1}^{N}\sum_{j=1}^{N} \gamma_i \gamma_j \boldsymbol{x}_i \boldsymbol{M}^{-1} \boldsymbol{x}_j \tag{26}$$

$$\textit{subject to :} 0 \leq \gamma_i \leq c, \quad \sum_{i=1}^{N} \gamma_i = 1. \tag{27}$$

Here, it should be noted that the parameter c can take any positive value. However, setting a value $c = 0$, eliminates the chance of convergence, since the constraints in (27) will never be met. Moreover, setting any value $c \geq 1$, leads to the same solution for $c = 1$, since the support vectors coefficients should satisfy $\sum_{i=1}^{N} \gamma_i = 1$. Thus, the parameter c should be limited to values of $(0, 1]$.

IV. Kernel SVDD exploiting local and global geometric data information

In the previous section, we have described the linear SVDD case, exploiting local and global geometric data information in its optimization process. In this section, we describe the kernel extension of the proposed method.

In order to obtain non linear solutions for the SVDD, it is required that we employ a mapping function $\phi(\cdot) : \mathbb{R}^D \mapsto \mathcal{H}$, such that we obtain data representations in the feature space $\mathcal{H}$. The dimensionality of $\mathcal{H}$ depends on the adopted kernel function choice, e.g., it can be even infinite when the RBF kernel function is used.

By employing the RBF kernel function, the matrix $\boldsymbol{M}$ that contains the combination of local and global geometric data relationships, becomes of arbitrary dimensionality, i.e.:

$$\boldsymbol{M} = \boldsymbol{\Phi}(\mu \boldsymbol{L}^G + (1 - \mu)\boldsymbol{L}^L)\boldsymbol{\Phi}^T, \tag{28}$$

where $\boldsymbol{\Phi}$ is a matrix that contains the training data representation in the feature space $\mathcal{H}$, such that:

$$\boldsymbol{\Phi} = [\phi(\boldsymbol{x}_1), \ldots, \phi(\boldsymbol{x}_N)]. \tag{29}$$

The matrix $\boldsymbol{M}$ may be singular in such feature spaces. Thus, a regularized version of $\boldsymbol{M}$ should be employed, such that:

$$\tilde{\boldsymbol{M}} = \boldsymbol{M} + \rho \boldsymbol{I}, \tag{30}$$

where ρ is a regularization parameter allowing the matrix $\boldsymbol{M}$ to be invertible, and $\boldsymbol{I}$ is an identity matrix of appropriate dimensions. By exploiting the Woodbury identity, the inverse of $\tilde{\boldsymbol{M}}$ is given by:

$$\tilde{\boldsymbol{M}}^{-1} = \frac{1}{\rho}\boldsymbol{I} - \frac{1}{\rho^2}\boldsymbol{\Phi}\left([\mu \boldsymbol{L}_G + (1-\mu)\boldsymbol{L}_L]^{-1} + \frac{1}{\rho}\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T. \tag{31}$$

The calculation of the term $\boldsymbol{\Phi}^T\boldsymbol{\Phi} = \boldsymbol{K}$ is possible and $\boldsymbol{K} \in \mathbb{R}^{N\times N}$ is the kernel matrix expressing data similarity in $\mathcal{H}$. In what follows, we employ a matrix $\boldsymbol{V}$ for notation simplicity, where:

$$\boldsymbol{V} = \frac{1}{\rho^2}\left([\mu \boldsymbol{L}_G + (1-\mu)\boldsymbol{L}_L]^{-1} + \frac{1}{\rho}\boldsymbol{K}\right). \tag{32}$$

By replacing (31) in (26), we obtain an optimization problem, where we wish to maximize:

$$\begin{aligned}
&\sum_{i=1}^{N}\gamma_i\phi(\boldsymbol{x}_i)\left(\frac{1}{\rho}\boldsymbol{I} - \boldsymbol{\Phi}\boldsymbol{V}\boldsymbol{\Phi}^T\right)\phi(\boldsymbol{x}_i) - \\
&-\sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i\gamma_j\phi(\boldsymbol{x}_i)\left(\frac{1}{\rho}\boldsymbol{I} - \boldsymbol{\Phi}\boldsymbol{V}\boldsymbol{\Phi}^T\right)\phi(\boldsymbol{x}_j) = \\
&= \sum_{i=1}^{N}\gamma_i\left(\frac{1}{\rho}\kappa(\boldsymbol{x}_i,\boldsymbol{x}_i) - \boldsymbol{\kappa}_i^T\boldsymbol{V}\boldsymbol{\kappa}_i\right) - \\
&-\boldsymbol{\gamma}^T\left(\frac{1}{\rho}\boldsymbol{K} - \boldsymbol{K}\boldsymbol{V}\boldsymbol{K}\right)\boldsymbol{\gamma},
\end{aligned} \tag{33}$$

subject to the contraints in (27). The function $\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)$ is the kernel function, expressing data similarity between training samples $\boldsymbol{x}_i, \boldsymbol{x}_j$, and $\boldsymbol{\kappa_i}$ is the i-th column of the kernel matrix $\boldsymbol{K}$.

By observing (33), it is of the same form as the standard kernel SVDD [1], solved with a modified kernel:

$$\tilde{\boldsymbol{K}} = \frac{1}{\rho}\boldsymbol{K} - \boldsymbol{K}\boldsymbol{V}\boldsymbol{K}. \tag{34}$$

Thus, the solution of the proposed method lies in a different space (i.e., $\mathcal{H}' \neq \mathcal{H}$), where global and local geometric data relationships have been expressed. Finally, in order to determine whether a test sample $\boldsymbol{x} \in \mathbb{R}^D$, falls inside the hypersphere, we can employ the standard SVDD solution (8), using the modified kernel presented in (34).

V. Experimental Evaluation

In this section, we describe the conducted experiments in order to evaluate the performance of the proposed SVDD classifier, exploiting local and global geometric data relationships in the optimization process. In what follows, the performance of the proposed method is reported with the acronym SVDD-GL, where global geometric data relationships were expressed through (9). Since the kernel version consistently outperformed the linear alternative, we only report the obtained performance of the kernel version. For comparison reasons, we have also trained standard SVDD classifier [1], along with the SVDD exploiting the whitening transform [11] (SVDD-W). Here we should note that SVDD-W, is a variant of the proposed method when employing $\mu = 1$ and $\boldsymbol{W}^G$ is calculated through (9). Additionally, we have employed the proposed method for a value of $\mu = 1$, exploiting only global information in its optimization process (SVDD-G), by calculating $\boldsymbol{W}^G$ through (11). Finally, we trained a variant of the proposed algorithm, exploiting only local geometric data information (SVDD-L) in its optimization process. That is, we have employed the proposed method, fixing the value of the parameter $\mu = 0$. This variant can be considered as a modified version of the semi-supervised SVDD method, proposed in [12].

For each dataset, we have determined the optimal set of parameters for each method, by employing a cross validation approach. We have set the values of $c = (0.01, 0.1, 0.2, 0.4, 0.5, 0.6, 0.8)$, values of $\rho = 10^{(-3,\dots,3)}$ and finally, we have set the values of $\mu = (0.2, 0.4, 0.6, 0.8)$. We have employed the same values for all competing methods.

TABLE I
Dataset information

Dataset Name	Problem	Dimensions	Dataset Size (most populated class)
Olympic Sports	HAR	500	640 (54)
Hollywood2	HAR	4000	823 (135)
Hollywood3D	HAR	4000	643 (51)
PubFig+LFW	FR	1536	8720 (231)

We conducted experiments in publicly available datasets. Information for each employed dataset is summarized in Table I. Since all employed datasets are involve multiclass classification problems, we have tranformed them into imbalanced binary problems. That is, we have trained the classifiers in each class separately, and then tested on the test portion of the datasets, containing all classes. In imbalanced classification problems, a commonly employed metric is the g-mean metric [20], which combines both precision and recall measurements as follows:

$$g = \sqrt{Precision \times Recall}. \tag{35}$$

In our first set of experiments, we employed the competing methods in human action recognition (HAR). We have employed the Olympic Sports [21], the Hollywood2 [22] and Hollywood3D [23] datasets. In order to obtain vectorial video representation for the HAR datasets, we have employed the dense trajectories video description [24]. This video description calculates five descriptor types, namely the Histogram of Oriented Gradients, Histogram of Optical Flow, Motion Boundary Histogram along direction x, Motion Boundary Histogram along direction y and the normalized trajectory coordinates, on the trajectories of densely-sampled video frame interest points. Afterwards, we have employed the Bag-of-Words model [25] in order to obtain five histograms for each video, one for each descriptor type. Finally, the descriptor types are combined using a late fusion approach.

In our second set of experiments, we have employed the competing algorithms in face recognition (FR). We have employed the PubFig+LFW dataset [26]. We have employed the feature vectors (Histogram of Oriented Gradients, Local Binary Patterns, and Gabor wavelet features, reduced to 2048 dimensions with PCA), which were extracted from $13,002$ facial images representing 83 individuals from PubFig83, divided into 2/3 training (8720 faces) and 1/3 testing set ($4,282$ faces), as well as $12,066$ images representing over $5,000$ faces which were used as a distractor set from LFW. We have employed the first 1536 dimensions from the 2048, as suggested in [26]. For each of the 83 individuals, we have employed the training images for each class and tested on the respective test set of this class, as well the 500 first images of the distractor set.

TABLE II
AVERAGE G-MEANS RATES

Dataset Name	SVDD	SVDD-W	SVDD-L	SVDD-G	**SVDD-GL**
Olympic Sports	61.14	63.14	64.97	65.33	**73.18**
Hollywood2	58.29	59.54	59.37	60.12	**60.74**
Hollywood3D	56.79	58.18	58.52	58.54	**62.59**
PubFig+LFW	76.55	77.46	77.15	77.20	**77.95**

Finally, we report the average obtained g-mean metric for each dataset in Table II. As can be seen, the proposed method outperformed the competition, in every case. From the conducted experiments, we can conclude that employing multiple graph types in the SVDD optimization process, improves the generalization performance.

VI. Conclusion

In this paper, we have described a one-class classification method, exploiting multiple graph structures in the SVDD optimization process. Our experiments denote that kNN and fully connected graphs can increase the generalization performance of SVDD when used simultaneously. This can be explained by the fact that kNN graphs provide local geometric information, and fully connected graphs provide global geometric information about the training class. Future work could include extending the present work in other classifiers. Moreover, the proposed method could exploit multiple graph types, thus evolving methods to automatically determine the appropriate combination of multiple graph types could be developed.

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) and grant agreement number 287674 (3DTVS).

References

[1] D. M. Tax and R. P. Duin, "Support vector data description," *Machine learning*, vol. 54, no. 1, pp. 45–66, 2004.

[2] S. Khazai, A. Safari, B. Mojaradi, and S. Homayouni, "Improving the svdd approach to hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 4, pp. 594–598, 2012.

[3] J. Muñoz-Marí, L. Bruzzone, and G. Camps-Valls, "A support vector domain description approach to supervised classification of remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 8, pp. 2683–2692, 2007.

[4] Z. Jiang, X. Feng, X. Feng, and L. Li, "A study of svdd-based algorithm to the fault diagnosis of mechanical equipment system," *Physics Procedia*, vol. 33, pp. 1068–1073, 2012.

[5] A. Yousef and N. M. Charkari, "A novel method based on physicochemical properties of amino acids and one class classification algorithm for disease gene identification," *Journal of biomedical informatics*, vol. 56, pp. 300–306, 2015.

[6] S.-W. Lee, J. Park, and S.-W. Lee, "Low resolution face recognition based on support vector data description," *Pattern Recognition*, vol. 39, no. 9, pp. 1809–1812, 2006.

[7] V. Mygdalis, A. Iosifidis, A. Tefas, and I. Pitas, "Video summarization based on subclass support vector data description," *IEEE Symposium Series on Computational Intelligence (SSCI), IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES)*, pp. 183–187, 2014.

[8] S.-M. Guo, L.-C. Chen, and J. S. H. Tsai, "A boundary method for outlier detection based on support vector domain description," *Pattern Recognition*, vol. 42, no. 1, pp. 77–83, 2009.

[9] B. Liu, Y. Xiao, L. Cao, Z. Hao, and F. Deng, "Svdd-based outlier detection on uncertain data," *Knowledge and information systems*, vol. 34, no. 3, pp. 597–618, 2013.

[10] Y. Xiao, B. Liu, Z. Hao, and L. Cao, "A k-farthest-neighbor-based approach for support vector data description," *Applied intelligence*, vol. 41, no. 1, pp. 196–211, 2014.

[11] S. Zafeiriou and N. Laskaris, "On the improvement of support vector techniques for clustering by means of whitening transform," *IEEE Signal Processing Letters*, vol. 15, pp. 198–201, 2008.

[12] P. Duong, V. Nguyen, M. Dinh, T. Le, D. Tran, and W. Ma, "Graph-based semi-supervised support vector data description for novelty detection," *International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6, 2015.

[13] J. Muñoz-Marí, F. Bovolo, L. Gómez-Chova, L. Bruzzone, and G. Camp-Valls, "Semisupervised one-class support vector machines for classification of remote sensing data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 8, pp. 3188–3197, 2010.

[14] G. Arvanitidis and A. Tefas, "Exploiting graph embedding in support vector machines," *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2012.

[15] A. Iosifidis, A. Tefas, and I. Pitas, "Graph embedded extreme learning machine," *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 311–324, 2016.

[16] A. Iosifidis and M. Gabbouj, "Multi-class support vector machine classifiers using intrinsic and penalty graphs," *Pattern Recognition*, vol. 55, pp. 231–246, 2016.

[17] A. Maronidis, A. Tefas, and I. Pitas, "Subclass graph embedding and a marginal fisher analysis paradigm," *Pattern Recognition*, vol. 48, no. 12, pp. 4024–4035, 2015.

[18] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: a general framework for dimensionality reduction," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 1, pp. 40–51, 2007.

[19] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *The Journal of Machine Learning Research*, vol. 12, pp. 2211–2268, 2011.

[20] M. Kubat, R. C. Holte, and S. Matwin, "Machine learning for the detection of oil spills in satellite radar images," *Machine learning*, vol. 30, no. 2-3, pp. 195–215, 1998.

[21] J. C. Niebles, C.-W. Chen, and L. Fei-Fei, "Modeling temporal structure of decomposable motion segments for activity classification," *European Conference on Computer Vision (ECCV)*, pp. 392–405, 2010.

[22] M. Marszalek, I. Laptev, and C. Schmid, "Actions in context," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2929–2936, 2009.

[23] S. Hadfield and R. Bowden, "Hollywood 3d: Recognizing actions in 3d natural scenes," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3398–3405, 2013.

[24] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International journal of computer vision*, vol. 103, no. 1, pp. 60–79, 2013.

[25] A. Iosifidis, A. Tefas, and I. Pitas, "Discriminant bag of words based representation for human action recognition," *Pattern Recognition Letters*, 2014.

[26] B. Becker and E. Ortiz, "Evaluating open-universe face identification on the web," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 904–911, 2013.