IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 21, 2019, accepted July 16, 2019, date of publication July 24, 2019, date of current version August 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2930824

Multi-View Learning With Robust Generalized Eigenvalue Proximal SVM

PENG HUANG[1], QIAOLIN YE[1], (Member, IEEE), YAN LI[1], GUOWEI YANG[2,3], AND YINGAN LIU[1]

[1]College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China
[2]Jiangsu Key Laboratory of Auditing Information Engineering, School of Information Engineering, Nanjing Audit University, Nanjing 211815, China
[3]School of Electronic Information, Qingdao University, Qingdao 266071, China

Corresponding author: Yingan Liu (lyahp123@njfu.edu.cn)

This work was supported in part by the National Science Foundation of China under Grant 61871444 and Grant 61772277, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20171453, and in part by the Qinglan and Six Talent Peaks Project of Jiangsu Province.

ABSTRACT Multi-view learning mechanism, which enhances learning performance by training multi-model data sets, is a popular filed in recent years. Multi-view generalized eigenvalue proximal support vector machine (MvGSVM), as a most recently proposed classifier, has been shown to be successful in multi-model classification, which incorporates multi-view learning into classical GEPSVM. However, this method is still based on squared L2-norm distance measure, thus its robustness is not guaranteed in the presence of outliers. To address this problem, we propose a robust multi-view GEPSVM based on Lp-norm minimization and Ls-norm maximization. But, the introduction of the Lp-norm and Ls-norm makes the problem different from the generalized eigenvalue problem. So an efficient iterative algorithm is designed to solve this problem, and we also give the proof of convergence of the algorithm. The performances in extensive experiments demonstrate the effectiveness and robustness of the algorithm.

INDEX TERMS Robustness, multi-view learning, classification, generalized eigenvalue proximal support vector machine (GEPSVM).

I. INTRODUCTION

Support vector machine (SVM) [1]–[3], as a supervised learning tool [4], has performed powerfully in pattern recognition and data mining over the past decades. The main idea of traditional SVM is to construct two parallel hyperplanes between $+1$ class and -1 class datasets in a real space, and seek an optimal hyperplane by maximizing the margin between the two parallel hyperplanes [5]. SVM has been popularly applied in a great deal of practical problems [6]–[10] with its characteristic based on structural and empirical risk minimization, such as image classification, handwriting recognition, disease diagnosis, bioinformatics and so on.

However, there are two main constraints for original SVM that limit its exposure to a wider range of applications: the complex Quadratic Programming Problems (QPPs) [11] and Exclusive Or problems (XOR). In order to solve the two problems, Mangasarian and Wild proposed a simple and fast classifier for binary classification problem, termed as generalized eigenvalue proximal support vector machine (GEPSVM) [12]. The main idea of GEPSVM is to generate two hyperplanes and each hyperplane is close to one class and as far as possible from another class [13]. Different from the standard SVM in obtaining the optimal plane by solving a quadratic programming problem (QPP), GEPSVM attempts to achieve two nonparallel hyperplanes by solving a pair of generalized eigenvalue problems. Hence, one superiority of GEPSVM is that it can deal with XOR problem effectively, which has been testified in many researches on the cross datasets [12], [16], [36]. With the powerful scalability of GEPSVM, many improved methods have been developed in the past period of time [14]–[22]. Jayadeva *et al.* [14] proposed an extended fuzzy multi-category algorithm based on GEPSVM for multiclass problem. Ye [16] put forward a new algorithm to resolve the singular problem by singular value decomposition, which is called improved proximal support vector machine via generalized eigenvalues (IGEPSVM). Shao [17] presented a developed version of GEPSVM via replacing generalized eigenvalue decomposition with standard eigenvalue decomposition, which is

The associate editor coordinating the review of this manuscript and approving it for publication was Tallha Akram.

more efficacious and faster than GEPSVM. Chen *et al.* trained a novel manifold proximal support vector machine (MPSVM) [18] for semi-supervised classification by introducing a manifold regularization (MR) term. After that, Liang *et al.* [20] proposed a novel method called manifold regularized proximal support vector machine via generalized eigenvalue (MRGEPSVM). By reformulating differential search algorithm (DSA) to find near optimal values of the GEPSVM parameters, Marghny *et al.* proposed DSA-GEPSVM [19]. In addition, inspired of the optimization objective for GEPSVM, Jayadera *et al.* raised twin support vector machine (TWSVM) [13], which tries to obtain two nonparallel hyperplanes by solving two small-scale QPPs instead of generalized eigenvalue problems, is an important branch of SVM. After that, there have been many improved versions of TWSVM in recently, like WLMSVM [23], IMBSVM [24] and so on.

Multiview learning (MVL) [25], [26] is an ascendant learning direction in machine learning that utilizes different feature sets of the same object to perform calculations, and via capitalizing on the complementarity and consistency among distinct views to enhance classifiers performance [28], [29]. These diverse views are usually derived from common multi-model data with different feature extraction methods [25] in practical applications. For instance, one picture can be described by color and texture feature sets while a people can be identified by fingerprints or gaits. Furthermore, it is still valid for the artificially generated multi-view sets of the objects which have no extra modal features [27]. However, the main challenge of multi-view learning is to seek out an impactful model to combine multiple views. To solve the problem, Farquhar *et al.* [31] proposed a new method which combines the two distinct stages of kernel canonical correlation analysis (KCCA) [30] and SVM [1] into a single optimization in 2005, called SVM-2K, calculating with two views of the data. Sindhwani *et al.* [32] devised a co-regularization framework with multiple views. In recent years, Xie and Sun extended laplacian twin support vector machine (LapTWSVM) [33] to multi-view laplacian support vector machines (MvLapTWSVM) [34] with a new framework which combines two views in the constraint, by introducing two similar one-dimensional projections to find two variant TWSVM from distinct feature spaces alternative to the objective function. Later, by introducing a multi-view co-regularization to associate two views, Sun [35] gave an improved version based on GEPSVM, termed as multi-view learning with GEPSVM (MvGSVM), which converts a complex optimization problem to a generalized eigenvalue problem. Tang *et al.* propose a new multi-view privileged SVM model, multi-view privileged SVM model (PSVM-2V) [36]. After that, they built a new multi-view learning model based on nonparallel support vector machine, termed as MVNPSVM [37], which combines the large margin mechanism and the consensus principle. Zhang *et al.* stacked the correlation restricted boltzmann machine (RBM) to create the correlation deep belief network (DBN), and then proposed the multimodal correlation DBN for learning multi-view data representations (CRBM) [38]. He *et al.* [39] develop a model which can perform better in both multi-view and transfer learning settings (MTDM). Besides, there are many applications for multi-angle learning in other fields of machine learning, such as multi-view dimensionality reduction [40], multi-view ensemble learning [41], and multi-view clustering [42].

Notwithstanding, it is worth noting that almost methods mentioned above may suffer from the sensitivity of outliers or noises due to the squared L2-norm operation. In order to expand robust methods that can reduce the influence of outliers and noises, many works introduce L1-norm distance metric into related algorithms in recent years, as a result, extensive studies have shown that L1-norm operation is an effective way against noises [43]–[51]. For instance, Yan and Yan [45] reconstructed the ratio term of GEPSVM with L1-norm metric to seek two nonparallel planes by managing a pair of QPPs. L1-norm projection twin SVM (L1-TWSVM) is proved that has a more stable performance, in which Yan *et al.* [46] constructed an unconstrained convex programming problem and generate multiple projection axes for each class by using recursive algorithms. Besides, L1-norm is also actively applied in feature extraction and dimensionality reduction (DR), such as principal component analysis based on L1-norm maximization (PCA-L1) [52]–[54], linear discriminant analysis based on L1-norm maximization (L1-LDA) [55], [56], etc. Yet, considering the fact that although L1-norm distance can obtain better robustness than L2-norm distance, it also lacks satisfactory robustness especially large outliers and noises exist. In [64], inspired by [58]–[60], Sun *et al.* reformulated the optimization based on L1-GEPSVM via using an Lp-norm regularization to replace L2-norm regularization, named robust nonparallel proximal SVM with Lp-norm regularization (LpNPSVM), which solves a strongly convex programming problem in each iteration. Qi *et al.* [62] extended the generalized multiple kernel learning (GMKL) [63] to a more robust generalized MKL method, by joining L1-norm and Lp-norm. Actually, L1-norm and L2-norm are both the special form of Lp-norm. However, the related Lp-norm researches still employ non-robust squared L2-norm in the objectives. Previous our work [57] has shown that Lp-norm distance metric is a better choice for improving robustness, who redesigned a flexible linear discriminant analysis via simultaneous Ls-norm distance maximization and Lp-norm distance minimization (FLDA-Lsp), which solves a new more effective iterative algorithm for the solution to the objective as an excellent contribution, and the result of this research shows the convergence and superiority of Lp-norm distance metric.

Under the inspiration of the works of multi-view learning [25], [33], [37] and robust Lp-norm distance measure [57], we propose a robust multi-view GEPSVM algorithm, Lp,s-MvGEPSVM, which is based on the Lp-norm distance minimization and Ls-norm maximization.

Of course, the differences between the proposed method and our previous work, in addition to the multi-view learning, there is a dimension reduction algorithm in [57] but ours is a classification method. In specific, the main contributions of this paper is shown as follows:

(1) Considering the importance of the distance from hyperplanes to the corresponding $+1$ class and -1 class may be different, Lp,s-MvGEPSVM introduces Lp-norm and Ls-norm distance measure. Hence the adverse impacts resulting from outliers and noises could be alleviated by setting relatively minor p and s.

(2) Despite previous improvement, we also do the same managing for the multi-view co-regularization term to ensure the difference between the two views is as small as possible. Obviously, MvGSVM is a special form of our algorithm if we set $p = 2$ and $s = 2$ in (30), which illustrates the new proposed method is flexible. And in other words, the optimization of the more complex objective is one highlight of this paper.

(3) Since the objective function is non-absolutely convex and contains Lp-norm minimization term and Ls-norm maximization term, we put forward a new impactful method to solve this problem.

(4) The convergence of our iteration algorithm is guaranteed and the experiment results on the distinct datasets exhibit the effectiveness and robustness of our algorithm. Moreover, our ideas can be integrated into other methods that will be discussed in future work.

The remainder content of this paper is organized as follows. Section II gives a brief retrospect about GEPSVM and MvGSVM. Section III proposes Lp,s-MvGEPSVM with its geometric interpretation and section IV gives theoretical proof in details. All results of our experimentation are displayed in Section V. Finally, Section VI summarizes the whole work.

II. RELATED WORK

In this paper, we consider a binary classification problem and all vectors are column vectors. For a given real space of dimension d, suppose dataset $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_n\}$ and the corresponding label vector $y_i (i = 1, 2, \ldots, n) \in \{+1, -1\}$, $\mathbf{e}_1$ and $\mathbf{e}_2$ represent the identity column vectors of the appropriate dimension. And matrix $\mathbf{A} \in R^{n_1 \times d}$ is a set of points belonging to the $+1$ class in the sample, while matrix $\mathbf{B} \in R^{n_2 \times d}$ is a set of points belonging to the -1 class, where $n_1 + n_2 = n$.

A. GEPSVM

The purpose of GEPSVM [12] is to find two hyperplanes which are nonparallel and proximal optimal in space R^d

$$\mathbf{x}^T \mathbf{w}_1 + b_1 = 0, \qquad \mathbf{x}^T \mathbf{w}_2 + b_2 = 0. \tag{1}$$

making the first hyperplane closest to the points of class $+1$ and having the largest distance from the points of the -1 class, while the second hyperplane is the opposite performance, where $\mathbf{w}_1, \mathbf{w}_2 \in R^d, b_1, b_2 \in R$. Therefore, the goal of GEPSVM leads to the following two optimization problems

$$\min_{\mathbf{w}_1, b_1} \frac{\|\mathbf{A}\mathbf{w}_1 + \mathbf{e}_1 b_1\|^2 / \left\| [\mathbf{w}_1 \quad b_1]^T \right\|^2}{\|\mathbf{B}\mathbf{w}_1 + \mathbf{e}_2 b_1\|^2 / \left\| [\mathbf{w}_1 \quad b_1]^T \right\|^2} \tag{2}$$

and

$$\min_{\mathbf{w}_2, b_2} \frac{\|\mathbf{B}\mathbf{w}_2 + \mathbf{e}_2 b_2\|^2 / \left\| [\mathbf{w}_2 \quad b_2]^T \right\|^2}{\|\mathbf{A}\mathbf{w}_2 + \mathbf{e}_1 b_2\|^2 / \left\| [\mathbf{w}_2 \quad b_2]^T \right\|^2} \tag{3}$$

where $(\mathbf{w}_i, b_i) \neq 0$ $(i = 1, 2)$ and $\|.\|$ indicates the squared L2-norm operation, similarly, those also applies to the following formulas. The above objective functions can be optimized as

$$\min_{\mathbf{w}_1, b_1} \frac{\|\mathbf{A}\mathbf{w}_1 + \mathbf{e}_1 b_1\|^2}{\|\mathbf{B}\mathbf{w}_1 + \mathbf{e}_2 b_1\|^2} \tag{4}$$

and

$$\min_{\mathbf{w}_2, b_2} \frac{\|\mathbf{B}\mathbf{w}_2 + \mathbf{e}_2 b_2\|^2}{\|\mathbf{A}\mathbf{w}_2 + \mathbf{e}_1 b_2\|^2} \tag{5}$$

For the singularity problem caused by positive semi-definite matrix in the process of calculating, formulas (6) and (7) can obtain a more stable solution via introducing Tikhonov regularization term [65], as follows

$$\min_{\mathbf{w}_1, b_1} \frac{\|\mathbf{A}\mathbf{w}_1 + \mathbf{e}_1 b_1\|^2 + \delta \left\| [\mathbf{w}_2 \quad b_2]^T \right\|^2}{\|\mathbf{B}\mathbf{w}_2 + \mathbf{e}_2 b_2\|^2} \tag{6}$$

and

$$\min_{\mathbf{w}_2, b_2} \frac{\|\mathbf{B}\mathbf{w}_2 + \mathbf{e}_2 b_2\|^2 + \delta \left\| [\mathbf{w}_2 \quad b_2]^T \right\|^2}{\|\mathbf{A}\mathbf{w}_1 + \mathbf{e}_1 b_1\|^2} \tag{7}$$

where δ is a non-negative regularization parameter. Define $\mathbf{z}_1 = [\mathbf{w}_1 \quad b_1]^T$, $\mathbf{z}_2 = [\mathbf{w}_2 \quad b_2]^T$, $\mathbf{G} = [\mathbf{A} \quad \mathbf{e}_1]^T [\mathbf{A} \quad \mathbf{e}_1]$, $\mathbf{H} = [\mathbf{B} \quad \mathbf{e}_2]^T [\mathbf{B} \quad \mathbf{e}_2]$. Thence, the optimization problems (6) and (7) are equivalent to

$$\min_{\mathbf{z}_1} \frac{\mathbf{z}_1^T (\mathbf{G} + \delta \mathbf{I}) \mathbf{z}_1}{\mathbf{z}_1^T \mathbf{H} \mathbf{z}_1} \tag{8}$$

and

$$\min_{\mathbf{z}_2} \frac{\mathbf{z}_2^T (\mathbf{H} + \delta \mathbf{I}) \mathbf{z}_2}{\mathbf{z}_2^T \mathbf{G} \mathbf{z}_2} \tag{9}$$

where $\mathbf{I}$ is a unit square matrix that matching with $\mathbf{G}$ and $\mathbf{H}$. Obviously, both (8) and (9) are Rayleigh quotient problems. So the optimal solution of the objective function could be obtained easily by solving following two generalized eigenvalue problems

$$(\mathbf{G} + \delta \mathbf{I})\, \mathbf{z}_1 = \lambda_1 \mathbf{H} \mathbf{z}_1, \qquad \mathbf{z}_1 \neq \mathbf{0} \tag{10}$$

and

$$(\mathbf{H} + \delta \mathbf{I})\, \mathbf{z}_2 = \lambda_2 \mathbf{G} \mathbf{z}_2, \qquad \mathbf{z}_2 \neq \mathbf{0} \tag{11}$$

where λ_1 and λ_2 are the minimum eigenvalues of (10) and (11), $\mathbf{z}_1$ and $\mathbf{z}_2$ correspond to the eigenvectors of λ_1 and λ_2, respectively. And the first d components of $\mathbf{z}_1$ denote the weight vector $\mathbf{w}_i$ of the *i-th* hyperplane, the last component is bias b_i, with $i \in (1, 2)$.

$$\min_{\mathbf{z}_1\neq 0,\mathbf{z}_2\neq 0} \frac{\mathbf{z}_1^T\mathbf{G}_1\mathbf{z}_1+\mathbf{z}_2^T\mathbf{G}_2\mathbf{z}_2+\delta(\|\mathbf{z}_1\|^2+\|\mathbf{z}_2\|^2)+\varepsilon\left\|\mathbf{M}_{A_1}\mathbf{z}_1-\mathbf{M}_{A_2}\mathbf{z}_2\right\|^2}{\mathbf{z}_1^T\mathbf{H}_1\mathbf{z}_1+\mathbf{z}_2^T\mathbf{H}_2\mathbf{z}_2} \tag{14}$$

$$\min_{\xi_1\neq\mathbf{0},\xi_2\neq\mathbf{0}} \frac{\xi_1^T\mathbf{H}_1\xi_1+\xi_2^T\mathbf{H}_2\xi_2+\delta(\|\xi_1\|^2+\|\xi_2\|^2)+\varepsilon\left\|\mathbf{M}_{B_1}\xi_1-\mathbf{M}_{B_2}\xi_2\right\|^2}{\xi_1^T\mathbf{G}_1\xi_1+\xi_2^T\mathbf{G}_2\xi_2} \tag{21}$$

B. MvGSVM

Assuming there are two different views, each view has n samples $\mathbf{X}=(\mathbf{x}_1^i,\mathbf{x}_2^i,\ldots,\mathbf{x}_n^i)(i=1,2)$ and shares a common label vector $\mathbf{y}_j(j=1,2,\ldots,n)\in\{+1,-1\}$, of course, they have different dimensions. Matrix $\mathbf{A}_1\in R^{n_1\times d_1}$ denotes the first view of positive class, matrix $\mathbf{A}_2\in R^{n_1\times d_2}$ denotes the second view of this class. Similarly, we have $\mathbf{B}_1\in R^{n_2\times d_1}$ and $\mathbf{B}_2\in R^{n_2\times d_2}$ belong to the first view and the second view of negative class, where $n=n_1+n_2$.There are two hyperplanes for each view as shown

$$\begin{array}{lll} view\ 1 & : & \mathbf{x}_1^T\mathbf{w}_1+b_1=0\,, \quad \mathbf{x}_1^T\mathbf{u}_1+p_1=0 \\ view\ 2 & : & \mathbf{x}_2^T\mathbf{w}_2+b_2=0\,, \quad \mathbf{x}_2^T\mathbf{u}_2+p_2=0 \end{array} \tag{12}$$

where $\mathbf{x}_i\,(i=1,2)$ means the data points in the *i-th* view. And some abbreviations are defined as follows

$$\begin{aligned} &\mathbf{M}_{A_1}=[\mathbf{A}_1\ \ \mathbf{e}],\quad \mathbf{M}_{A_2}=[\mathbf{A}_2\ \ \mathbf{e}],\quad \mathbf{G}_1=\mathbf{M}_{A_1}^T\mathbf{M}_{A_1} \\ &\mathbf{M}_{B_1}=[\mathbf{B}_1\ \ \mathbf{e}],\quad \mathbf{M}_{B_2}=[\mathbf{B}_2\ \ \mathbf{e}],\quad \mathbf{G}_2=\mathbf{M}_{A_2}^T\mathbf{M}_{A_2} \\ &\mathbf{H}_1=\mathbf{M}_{B_1}^T\mathbf{M}_{B_1},\quad \mathbf{H}_2=\mathbf{M}_{B_2}^T\mathbf{M}_{B_2},\quad \mathbf{z}_1=[\mathbf{w}_1\ \ b_1]^T \\ &\xi_1=[\mathbf{u}_1\ \ p_1]^T,\quad \xi_2=[\mathbf{u}_2\ \ p_2]^T,\quad \mathbf{z}_2=[\mathbf{w}_2\ \ b_2]^T \end{aligned}$$

Apparently $\mathbf{G}_1$ and $\mathbf{G}_2$ are symmetric matrices in space $R^{(d_1+1)\times(d_1+1)}$ and $R^{(d_2+1)\times(d_2+1)}$ for this part, as same as $\mathbf{H}_1$ and $\mathbf{H}_2$. The goal of MvGSVM is to seek out two hyperplanes of each view, in which the positive class hyperplane is as close as possible to the $+1$ class samples and as far away as possible from the corresponding -1 class samples, contrarily, the negative class hyperplane is the opposite. In addition, the difference between two views should be as small as possible. So before giving the objective function, a vital process is the multi-view co-regularization of the algorithm

$$f=\left\|\mathbf{M}_{A_1}\mathbf{z}_1-\mathbf{M}_{A_2}\mathbf{z}_2\right\|^2 \tag{13}$$

where f is going to be as small as possible. Now the optimization objective of the first problem in MvGSVM is given as (14), as shown at the top of this page, where the significance of the multi-view co-regularization term is to minimize the disparity between two views, $\|\mathbf{z}_1\|^2$ and $\|\mathbf{z}_2\|^2$ are Tikhonov regularization terms, δ and ε are non-negative weight factors. Equally, the co-regularization term can be written as

$$\begin{aligned} &\left\|\mathbf{M}_{A_1}\mathbf{z}_1-\mathbf{M}_{A_2}\mathbf{z}_2\right\|^2 \\ &=\left\|\begin{bmatrix}\mathbf{M}_{A_1} & -\mathbf{M}_{A_2}\end{bmatrix}\begin{bmatrix}\mathbf{z}_1^T & \mathbf{z}_2^T\end{bmatrix}^T\right\|^2 \\ &=\begin{bmatrix}\mathbf{z}_1^T & \mathbf{z}_2^T\end{bmatrix}\begin{bmatrix}\mathbf{M}_{A_1} & -\mathbf{M}_{A_2}\end{bmatrix}^T\begin{bmatrix}\mathbf{M}_{A_1} & -\mathbf{M}_{A_2}\end{bmatrix}\begin{bmatrix}\mathbf{z}_1^T & \mathbf{z}_2^T\end{bmatrix}^T \end{aligned} \tag{15}$$

let $\mathbf{z}=[\mathbf{z}_1^T\quad \mathbf{z}_2^T]^T$, objection problem (14) is equivalent to

$$\begin{aligned} \min_{z\neq 0}\ & \frac{\mathbf{z}^T\begin{bmatrix}\mathbf{G}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{G}_2\end{bmatrix}\mathbf{z}+\delta\ \|\mathbf{z}\|^2}{\mathbf{z}^T\begin{bmatrix}\mathbf{H}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{H}_2\end{bmatrix}\mathbf{z}} \\ &+\frac{\varepsilon\mathbf{z}^T\begin{bmatrix}\mathbf{M}_{A_1} & -\mathbf{M}_{A_2}\end{bmatrix}^T\begin{bmatrix}\mathbf{M}_{A_1} & -\mathbf{M}_{A_2}\end{bmatrix}\mathbf{z}}{\mathbf{z}^T\begin{bmatrix}\mathbf{H}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{H}_2\end{bmatrix}\mathbf{z}} \end{aligned} \tag{16}$$

then define

$$\mathbf{K}_1=\begin{bmatrix}(1+\delta)\mathbf{G}_1 & -\delta\,\mathbf{M}_{A_1}^T\mathbf{M}_{A_2}\\ -\delta\,\mathbf{M}_{A_2}^T\mathbf{M}_{A_1} & (1+\delta)\mathbf{G}_2\end{bmatrix}+\varepsilon\mathbf{I},\quad \mathbf{T}_1=\begin{bmatrix}\mathbf{H}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{H}_2\end{bmatrix} \tag{17}$$

so the optimization problem is converted to a generalized Rayleigh quotient

$$\min_{\mathbf{z}\neq\mathbf{0}}\frac{\mathbf{z}^T\mathbf{K}_1\mathbf{z}}{\mathbf{z}^T\mathbf{T}_1\mathbf{z}} \tag{18}$$

It is easy to get the optimum solution by dealing with a generalized eigenvalue problem as follows

$$\mathbf{K}_1\mathbf{z}=\lambda_1\mathbf{T}_1\mathbf{z} \tag{19}$$

The second optimization problem can be solved in the same way as the first one. And the second multi-view co-regularization term is

$$\left\|\mathbf{M}_{B_1}\xi_1-\mathbf{M}_{B_2}\xi_2\right\|^2 \tag{20}$$

therefore, via drawing term (20) into the other objective function, the optimal problem can be rewritten as (21), as shown at the top of this page, by employing the same processing method with (15), the second minimization problem with $\xi=\begin{bmatrix}\xi_1^T & \xi_2^T\end{bmatrix}^T$ can be described as

$$\begin{aligned} \min_{\xi\neq 0}\ & \frac{\xi^T\begin{bmatrix}\mathbf{H}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{H}_2\end{bmatrix}\xi+\delta\ \|\xi\|^2}{\xi^T\begin{bmatrix}\mathbf{G}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{G}_2\end{bmatrix}\xi} \\ &+\frac{\varepsilon\xi^T\begin{bmatrix}\mathbf{M}_{B_1} & -\mathbf{M}_{B_2}\end{bmatrix}^T\begin{bmatrix}\mathbf{M}_{B_1} & -\mathbf{M}_{B_2}\end{bmatrix}\xi}{\xi^T\begin{bmatrix}\mathbf{G}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{G}_2\end{bmatrix}\xi} \end{aligned} \tag{22}$$

and let

$$\mathbf{K}_2=\begin{bmatrix}(1+\delta)\mathbf{H}_1 & -\delta\,\mathbf{M}_{B_1}^T\mathbf{M}_{B_2}\\ -\delta\,\mathbf{M}_{B_2}^T\mathbf{M}_{B_1} & (1+\delta)\mathbf{H}_2\end{bmatrix}+\varepsilon\mathbf{I},\quad \mathbf{T}_2=\begin{bmatrix}\mathbf{G}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{G}_2\end{bmatrix} \tag{23}$$

same as (18), the formula (22) is simplified to the following Rayleigh quotient

$$\min_{\xi \neq \mathbf{0}} \frac{\xi^T \mathbf{K}_2 \xi}{\xi^T \mathbf{T}_2 \xi} \tag{24}$$

Naturally, it is easy to get the optimum solution by resolving the following generalized eigenvalue problem

$$\mathbf{K}_2 \xi = \lambda_2 \mathbf{T}_2 \xi \tag{25}$$

where λ_1 and λ_2 have identical definitions with GEPSVM. So that there are four hyperplane parameters $\mathbf{z}_1, \mathbf{z}_2, \xi_1, \xi_2$ to determine four hyperplanes corresponding to each classes in the two views. Now, for given test samples $\mathbf{x}_1$ and $\mathbf{x}_2$, the Euclidean distance from samples to hyperplanes in each view can be calculated

$$\begin{aligned} view1: \quad \text{dist11} &= \frac{\left|\mathbf{x}_1^T \mathbf{w}_1 + b_1\right|}{\|\mathbf{w}_1\|}, \qquad \text{dist12} = \frac{\left|\mathbf{x}_1^T \mathbf{u}_1 + \gamma_1\right|}{\|\mathbf{u}_1\|} \\ view2: \quad \text{dist21} &= \frac{\left|\mathbf{x}_2^T \mathbf{w}_2 + b_2\right|}{\|\mathbf{w}_2\|}, \qquad \text{dist22} = \frac{\left|\mathbf{x}_2^T \mathbf{u}_2 + \gamma_2\right|}{\|\mathbf{u}_2\|} \end{aligned} \tag{26}$$

Thus the decision function for $\mathbf{x}_1$ and $\mathbf{x}_2$ is

$$\hat{y} = \text{sign}(\text{dist12} + \text{dist22} - \text{dist11} - \text{dist21}) \tag{27}$$

where $\hat{y}$ is the predicted result of the hybrid views.

III. Lp,s-MvGEPSVM

As shown that MvGEPSVM adopts squared L2-norm distance metric, which is not robust against outliers [33]. Inspired by our previous work [50], we proposed a new multi-view learning method with generalized eigenvalue proximal support vector machines, called Multi-view Learning with Robust Generalized Eigenvalue Proximal SVM (Lp,s-MvGEPSVM), which computes the distances from the positive and negative points to the plane by using Lp-norm distance and Ls-norm distance.

Let $\mathbf{X} = (\mathbf{x}_1^i, \mathbf{x}_2^i, \ldots, \mathbf{x}_n^i)(i = 1, 2)$ denotes n points in *i-th* view and $y_j(j = 1, 2, \ldots, n) \in \{+1, -1\}$ be the corresponding co-label. Define matrices $\mathbf{A}_1 \in R^{n_1 \times d_1}$ and $\mathbf{A}_2 \in R^{n_1 \times d_2}$ representing the first and second view of $+1$ class respectively. Similarly, $\mathbf{B}_1 \in R^{n_2 \times d_1}$ and $\mathbf{B}_2 \in R^{n_2 \times d_2}$ represent the -1 class of the corresponding views, with $n_1 + n_2 = n$. Let sign$(\cdot)$ be a sign function with sign$(\cdot) = 1$, where $(\cdot)$ is a positive value and opposite sign otherwise. For a given $p > 0$, the Lp-norm of a real vector $\mathbf{f} \in R^d$ is defined as $||\mathbf{f}||_p = (\sum_{i=1}^d |\mathbf{f}_i|^P)^{1/p}$. Pay attention that p is a variable that can be replace by an arbitrary character such as s.

The goal now is to find two nonparallel hyperplanes in each view that follows the same significance as MvGSVM

$$view\ 1 \quad : \quad \mathbf{x}_1^T \mathbf{w}_1 + b_1 = 0, \qquad \mathbf{x}_1^T \mathbf{u}_1 + p_1 = 0$$

$$view\ 2 \quad : \quad \mathbf{x}_2^T \mathbf{w}_2 + b_2 = 0, \qquad \mathbf{x}_2^T \mathbf{u}_2 + p_2 = 0 \tag{28}$$

However, MvGSVM is also based on the traditional squared L2-norm distance metric which broadens the outliers influence. L1-GEPSVM [39] substitutes L1-norm for squared L2-norm in GEPSVM, which is more robustness than the latter. And inspired by the related works in [50], with considering the degree of outliers act on the hyperplanes to distinct classes may be diverse, we supersede the squared L2-norm distance metric by Lp-norm and Ls-norm measures. In addition, we also employ Lp-norm to replace squared L2-norm for the multi-view co-regularization term.

Before giving the optimization problem, it is necessary to make some definitions as follows

$$\begin{aligned} &\mathbf{H}_1 = [\mathbf{A}_1 \ \ \mathbf{e}]^T, \quad \mathbf{H}_2 = [\mathbf{A}_2 \ \ \mathbf{e}]^T, \quad \mathbf{z}_1 = \begin{bmatrix} \mathbf{w}_1 \\ b_1 \end{bmatrix}, \quad \mathbf{z}_2 = \begin{bmatrix} \mathbf{w}_2 \\ b_2 \end{bmatrix} \\ &\mathbf{G}_1 = [\mathbf{B}_1 \ \ \mathbf{e}]^T, \quad \mathbf{G}_2 = [\mathbf{B}_2 \ \ \mathbf{e}]^T, \quad \xi_1 = \begin{bmatrix} \mathbf{u}_1 \\ \gamma_1 \end{bmatrix}, \quad \xi_2 = \begin{bmatrix} \mathbf{u}_2 \\ \gamma_2 \end{bmatrix} \end{aligned} \tag{29}$$

where $\mathbf{e}$ represents the identity column vector of the appropriate dimension. And $\left\|\mathbf{H}_1^T \mathbf{z}_1 - \mathbf{H}_2^T \mathbf{z}_2\right\|_p^p$ is a multi-view co-regularization term which minimizes the difference between two views. These bring about the first objective function (30), as shown at the bottom of this page, the formula above can be written in the form of vector operations by

$$\begin{aligned} \min_{\mathbf{z}_1, \mathbf{z}_2} & \frac{\sum_{i=1}^{n_1} \left|\mathbf{h}_i^{(1)^T} \mathbf{z}_1\right|^p + \sum_{i=1}^{n_1} \left|\mathbf{h}_i^{(2)^T} \mathbf{z}_2\right|^p}{\sum_{j=1}^{n_2} \left|\mathbf{g}_j^{(1)^T} \mathbf{z}_1\right|^s + \sum_{j=1}^{n_2} \left|\mathbf{g}_j^{(2)^T} \mathbf{z}_2\right|^s} \\ & + \frac{\sum_{i=1}^{n_1} \varepsilon \left|\mathbf{h}_i^{(1)^T} \mathbf{z}_1 - \mathbf{h}_i^{(2)^T} \mathbf{z}_2\right|^p}{\sum_{j=1}^{n_2} \left|\mathbf{g}_j^{(1)^T} \mathbf{z}_1\right|^s + \sum_{j=1}^{n_2} \left|\mathbf{g}_j^{(2)^T} \mathbf{z}_2\right|^s} \\ & + \frac{\sum_{i=1}^{d_1+1} \delta \left|\mathbf{e}_i^{(1)^T} \mathbf{z}_1\right|^p + \sum_{i=1}^{d_2+1} \delta \left|\mathbf{e}_i^{(2)^T} \mathbf{z}_2\right|^p}{\sum_{j=1}^{n_2} \left|\mathbf{g}_j^{(1)^T} \mathbf{z}_1\right|^s + \sum_{j=1}^{n_2} \left|\mathbf{g}_j^{(2)^T} \mathbf{z}_2\right|^s} \end{aligned} \tag{31}$$

where $\mathbf{e}_i^1(\mathbf{e}_i^2)$ denotes the vector of appropriate dimension in the first (second) view that the *i-th* element is 1 and others are 0, $\mathbf{h}_i^{(1/2)}, \mathbf{g}_j^{(1/2)}$ represent the column vectors of $\mathbf{H}_{1/2}, \mathbf{G}_{1/2}$, ε is the coefficient of multi-view co-regularization and δ represents the coefficient of Tikhonov regularization terms, which both are non-negative. This problem (31) makes the ratio of the distance from positive hyperplanes to $+1$ class and -1 class is as small as possible. The adverse impacts causing by outliers could be alleviated by setting a small value to p and s, and theoretical results shows the objective is robust when $0 < p < 2$, $0 < s < 2$. Obviously, MvGSVM with L2-norm is a special form of our algorithm. If set $p = 2$ and $s = 2$ in (30), we will obtain

$$\min_{\mathbf{z}_1, \mathbf{z}_2} \frac{\left\|\mathbf{H}_1^T \mathbf{z}_1\right\|_p^p + \left\|\mathbf{H}_2^T \mathbf{z}_2\right\|_p^p + \varepsilon \left\|\mathbf{H}_1^T \mathbf{z}_1 - \mathbf{H}_2^T \mathbf{z}_2\right\|_p^p + \delta \left\|\mathbf{z}_1\right\|_p^p + \delta \left\|\mathbf{z}_2\right\|_p^p}{\left\|\mathbf{G}_1^T \mathbf{z}_1\right\|_s^s + \left\|\mathbf{G}_2^T \mathbf{z}_2\right\|_s^s} \tag{30}$$

MvGSVM, which indicates our method is very adaptable. However, despite the improved robustness and flexibility of our approach, there is also a problem need to be solved. The introduction of non-convex proxy function makes it impossible to resolve for a generalized eigenvalue problem. Here, we propose solving (31) by transforming the problem into a difference formulation, inspired by [48].

Firstly, let

$$\lambda^{(t)} = \frac{\sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1^{(t)}\right|^p + \sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(2)^T}\mathbf{z}_2^{(t)}\right|^p}{\sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(1)^T}\mathbf{z}_1^{(t)}\right|^s + \sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(2)^T}\mathbf{z}_2^{(t)}\right|^s} + \frac{\sum_{i=1}^{n_1}\varepsilon\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1^{(t)} - \mathbf{h}_i^{(2)^T}\mathbf{z}_2^{(t)}\right|^p}{\sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(1)^T}\mathbf{z}_1^{(t)}\right|^s + \sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(2)^T}\mathbf{z}_2^{(t)}\right|^s} + \frac{\sum_{i=1}^{d_1+1}\delta\left|\mathbf{e}_i^{(1)^T}\mathbf{z}_1^{(t)}\right|^p + \sum_{i=1}^{d_2+1}\delta\left|\mathbf{e}_i^{(2)^T}\mathbf{z}_2^{(t)}\right|^p}{\sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(1)^T}\mathbf{z}_1^{(t)}\right|^s + \sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(2)^T}\mathbf{z}_2^{(t)}\right|^s} \quad (32)$$

where $\mathbf{z}_1^{(t)}$ and $\mathbf{z}_2^{(t)}$ denote the optimal solution vectors in the t-th iteration and $\lambda^{(t)}$ is the objective value at iteration t. After that, we get the solution of the $(t+1)$-th iteration by resolving the following problem

$$\begin{aligned}\mathbf{z}_1^{(t+1)}, \mathbf{z}_2^{(t+1)} = \arg\min_{\mathbf{z}^{(t)}} &\sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1\right|^p + \sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(2)^T}\mathbf{z}_2\right|^p \\ &+ \sum_{i=1}^{n_1}\varepsilon\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2\right|^p \\ &+ \sum_{i=1}^{d_1+1}\delta\left|\mathbf{e}_i^{(1)^T}\mathbf{z}_1\right|^p + \sum_{i=1}^{d_2+1}\delta\left|\mathbf{e}_i^{(2)^T}\mathbf{z}_2\right|^p \\ &- \lambda^{(t)}\left(\sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(1)^T}\mathbf{z}_1\right|^s + \sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(2)^T}\mathbf{z}_2\right|^s\right)\end{aligned} \quad (33)$$

after deriving $\mathbf{z}_1$ and $\mathbf{z}_2$ in (33) separately and setting them to zero, we get

$$\begin{aligned}&\sum_{i=1}^{n_1} p\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1\right|^{p-1}\mathrm{sign}(\mathbf{h}_i^{(1)^T}\mathbf{z}_1)\mathbf{h}_i^{(1)} \\ &+ \sum_{i=1}^{d_1+1} p\delta\left|\mathbf{e}_i^{(1)^T}\mathbf{z}_1\right|^{p-1}\mathrm{sign}(\mathbf{e}_i^{(1)^T}\mathbf{z}_1)\mathbf{e}_i^{(1)} \\ &+ \sum_{i=1}^{n_1} p\varepsilon\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2\right|^{p-1} \\ &\times \mathrm{sign}(\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2)\mathbf{h}_i^{(1)} \\ &- \lambda^{(t)}\sum_{j=1}^{n_2} s\left|\mathbf{g}_j^{(1)^T}\mathbf{z}_1\right|^{s-1}\mathrm{sign}(\mathbf{g}_j^{(1)^T}\mathbf{z}_1)\mathbf{g}_j^{(1)} = 0\end{aligned} \quad (34)$$

and

$$\begin{aligned}&\sum_{i=1}^{n_1} p\left|\mathbf{h}_i^{(2)^T}\mathbf{z}_2\right|^{p-1}\mathrm{sign}(\mathbf{h}_i^{(2)^T}\mathbf{z}_2)\mathbf{h}_i^{(2)} \\ &+ \sum_{i=1}^{d_2+1} p\delta\left|\mathbf{e}_i^{(2)^T}\mathbf{z}_2\right|^{p-1}\mathrm{sign}(\mathbf{e}_i^{(2)^T}\mathbf{z}_2)\mathbf{e}_i^{(2)} \\ &- \sum_{i=1}^{n_1} p\varepsilon\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2\right|^{p-1} \\ &\times \mathrm{sign}(\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2)\mathbf{h}_i^{(2)} \\ &- \lambda^{(t)}\sum_{i=1}^{n_2} s\left|\mathbf{g}_j^{(2)^T}\mathbf{z}_2\right|^{s-1}\mathrm{sign}(\mathbf{g}_j^{(2)^T}\mathbf{z}_2)\mathbf{g}_j^{(2)} = 0\end{aligned} \quad (35)$$

now, considering $\mathrm{sign}(\mathbf{z}^T h_i) = \mathbf{z}^T h_i / |\mathbf{z}^T h_i|$, equations (34) and (35) can be rewritten as following equivalent formulations

$$\begin{aligned}&\sum_{i=1}^{n_1}\frac{2p\mathbf{h}_i^{(1)^T}\mathbf{z}_1\mathbf{h}_i^{(1)}}{2\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1\right|^{2-p}} + \sum_{i=1}^{d_1+1}\frac{2p\delta\mathbf{e}_i^{(1)^T}\mathbf{z}_1\mathbf{e}_i^{(1)}}{2\left|\mathbf{e}_i^{(1)^T}\mathbf{z}_1\right|^{2-p}} \\ &+ \sum_{i=1}^{n_1}\frac{2p\varepsilon(\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2)\mathbf{h}_i^{(1)}}{2\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2\right|^{2-p}} \\ &- \lambda^{(t)}\sum_{j=1}^{n_2} s\left|\mathbf{g}_j^{(1)^T}\mathbf{z}_1\right|^{s-1}\mathrm{sign}(\mathbf{g}_j^{(1)^T}\mathbf{z}_1)\mathbf{g}_j^{(1)} = 0\end{aligned} \quad (36)$$

and

$$\begin{aligned}&\sum_{i=1}^{n_1}\frac{2p\mathbf{h}_i^{(2)^T}\mathbf{z}_2\mathbf{h}_i^{(2)}}{2\left|\mathbf{h}_i^{(2)^T}\mathbf{z}_2\right|^{2-p}} + \sum_{i=1}^{d_2+1}\frac{2p\delta\mathbf{e}_i^{(2)^T}\mathbf{z}_2\mathbf{e}_i^{(2)}}{2\left|\mathbf{e}_i^{(2)^T}\mathbf{z}_2\right|^{2-p}} \\ &- \sum_{i=1}^{n_1}\frac{2p\varepsilon(\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2)\mathbf{h}_i^{(2)}}{2\left|\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2\right|^{2-p}} \\ &- \lambda^{(t)}\sum_{j=1}^{n_2} s\left|\mathbf{g}_j^{(2)^T}\mathbf{z}_2\right|^{s-1}\mathrm{sign}(\mathbf{g}_j^{(2)^T}\mathbf{z}_2)\mathbf{g}_j^{(2)} = 0\end{aligned} \quad (37)$$

For (36), we construct a diagonal matrix $\mathbf{D}_1 \in R^{n_1\times n_1}$ and define $d_{i,i}^{(1)} = p/(2|\mathbf{h}_i^{(1)^T}\mathbf{z}_1|^{2-p})$ as the i-th diagonal entry of $\mathbf{D}_1$, as well as defining $u_{i,i}^{(1)} = p/(2|\mathbf{e}_i^{(1)^T}\mathbf{z}_1|^{2-p})$ and $v_{i,i} = p/(2|\mathbf{h}_i^{(1)^T}\mathbf{z}_1 - \mathbf{h}_i^{(2)^T}\mathbf{z}_2|^{2-p})$ as the i-th diagonal entry of $\mathbf{U}_1 \in R^{(d_1+1)\times(d_1+1)}$ and $\mathbf{V} \in R^{n_1\times n_1}$, respectively. Construct vector $\mathbf{k}_1 \in R^{n_2}$ with its i-th element as $k_j^{(1)} = s|\mathbf{g}_j^{(1)^T}\mathbf{z}_1|^{s-1}\mathrm{sign}(\mathbf{g}_j^{(1)^T}\mathbf{z}_1)$. Define $\mathbf{z} = [\,\mathbf{z}_1^T\ \mathbf{z}_2^T\,]^T$ and $\mathbf{h}_i = [\mathbf{h}_i^{(1)^T}\ -\mathbf{h}_i^{(2)^T}\,]^T$, thus the minus item could be converted into $\sum_{i=1}^{n_1} 2p\mathbf{h}_i^T\mathbf{z}\mathbf{h}_i^{(1)} \Big/ (2|\mathbf{h}_i^T\mathbf{z}|^{2-p})$. Then, the formula (36) can be equivalently reformed to a new matrix type in (38)

$$2\mathbf{H}_1\mathbf{D}_1\mathbf{H}_1^T\mathbf{z}_1 + 2\varepsilon\mathbf{H}_1\mathbf{V}\mathbf{H}^T\mathbf{z} + 2\delta\mathbf{E}_1\mathbf{U}_1\mathbf{E}_1^T\mathbf{z}_1 - \lambda^{(t)}\mathbf{G}_1\mathbf{k}_1 = \mathbf{0} \quad (38)$$

where $\mathbf{H}$ denotes $[\,\mathbf{H}_1^T\ -\mathbf{H}_2^T\,]^T$. And in the same manner as the conversion from (36) to (38), we define a diagonal matrix $\mathbf{D}_2 \in R^{n_1\times n_1}$ and let $d_{i,i}^{(2)} = p/(2|\mathbf{h}_i^{(2)^T}\mathbf{z}_2|^{2-p})$ be the i-th diagonal entry of $\mathbf{D}_2$. Set $u_{i,i}^{(2)} = p/(2|\mathbf{e}_i^{(2)^T}\mathbf{z}_2|^{2-p})$ as the i-th diagonal entry of $\mathbf{U}_2 \in R^{(d_2+1)\times(d_2+1)}$. Construct vector $\mathbf{k}_2 \in R^{n_2}$ with its i-th element as $k_j^{(2)} = s|\mathbf{g}_j^{(2)^T}\mathbf{z}_2|^{s-1}\mathrm{sign}(\mathbf{g}_j^{(2)^T}\mathbf{z}_2)$. Next, the equation (37) can be rewritten as

$$2\mathbf{H}_2\mathbf{D}_2\mathbf{H}_2^T\mathbf{z}_2 - 2\varepsilon\mathbf{H}_2\mathbf{V}\mathbf{H}^T\mathbf{z} + 2\delta\mathbf{E}_2\mathbf{U}_2\mathbf{E}_2^T\mathbf{z}_2 - \lambda^{(t)}\mathbf{G}_2\mathbf{k}_2 = \mathbf{0} \quad (39)$$

where $\mathbf{E}_1$ and $\mathbf{E}_2$ represent identity diagonal matrices corresponding to $\mathbf{z}_1$ and $\mathbf{z}_2$. Consolidate (38) and (39) we can

obtain

$$\begin{bmatrix}\mathbf{H}_1 & \\ & \mathbf{H}_2\end{bmatrix}\begin{bmatrix}\mathbf{D}_1 & \\ & \mathbf{D}_2\end{bmatrix}\begin{bmatrix}\mathbf{H}_1^T & \\ & \mathbf{H}_2^T\end{bmatrix}\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix} - \lambda^{(t)}\begin{bmatrix}\mathbf{G}_1 & \\ & \mathbf{G}_2\end{bmatrix}\begin{bmatrix}\mathbf{k}_1\\ \mathbf{k}_2\end{bmatrix} + \delta\begin{bmatrix}\mathbf{E}_1 & \\ & \mathbf{E}_2\end{bmatrix}\begin{bmatrix}\mathbf{U}_1 & \\ & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{E}_1^T & \\ & \mathbf{E}_2^T\end{bmatrix}\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix} + \varepsilon\begin{bmatrix}\mathbf{H}_1\\ -\mathbf{H}_2\end{bmatrix}[\mathbf{V}]\begin{bmatrix}\mathbf{H}_1^T & -\mathbf{H}_2^T\end{bmatrix}\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix} = \mathbf{0} \tag{40}$$

Let

$$\mathbf{M}_{H_1} = \begin{bmatrix}\mathbf{H}_1 & \\ & \mathbf{H}_2\end{bmatrix},\quad \mathbf{M}_{H_2} = \begin{bmatrix}\mathbf{H}_1\\ -\mathbf{H}_2\end{bmatrix},\quad \mathbf{D} = \begin{bmatrix}\mathbf{D}_1 & \\ & \mathbf{D}_2\end{bmatrix},$$
$$\mathbf{E} = \begin{bmatrix}\mathbf{E}_1 & \\ & \mathbf{E}_2\end{bmatrix},\quad \mathbf{U} = \begin{bmatrix}\mathbf{U}_1 & \\ & \mathbf{U}_2\end{bmatrix},$$
$$\mathbf{M}_{G_1} = \begin{bmatrix}\mathbf{G}_1 & \\ & \mathbf{G}_2\end{bmatrix}, \mathbf{k} = \begin{bmatrix}\mathbf{k}_1\\ \mathbf{k}_2\end{bmatrix} \tag{41}$$

according to the transformations described above, (40) can be written as

$$2\mathbf{M}_{H_1}\mathbf{D}\mathbf{M}_{H_1}^T\mathbf{z} + 2\varepsilon\mathbf{M}_{H_2}\mathbf{V}\mathbf{M}_{H_2}^T\mathbf{z} + 2\delta\mathbf{E}\mathbf{U}\mathbf{E}^T\mathbf{z} - \lambda^{(t)}\mathbf{M}_{G_1}\mathbf{k} = \mathbf{0} \tag{42}$$

which has the same solution as the following objective question

$$\mathbf{z}^{(t+1)} = \arg\min_{\mathbf{z}}\ \mathbf{z}^T\mathbf{M}_{H_1}\mathbf{D}\mathbf{M}_{H_1}^T\mathbf{z} + \varepsilon\mathbf{z}^T\mathbf{M}_{H_2}\mathbf{V}\mathbf{M}_{H_2}^T\mathbf{z} + \delta\mathbf{z}^T\mathbf{E}\mathbf{U}\mathbf{E}^T\mathbf{z} - \lambda^{(t)}\mathbf{z}^T\mathbf{M}_{G_1}\mathbf{k} \tag{43}$$

since $\mathbf{D}$, $\mathbf{V}$, $\mathbf{U}$ and $\mathbf{k}$ are unknown variables and connected to $\mathbf{z}$ that could be solved by an iterative optimization framework. In other words, we calculate $\mathbf{z}$ of the last iteration to get the current $\mathbf{D}$, $\mathbf{V}$, $\mathbf{U}$ and $\mathbf{k}$.

In the same way, by introducing another multi-view co-regularization term $\left\|\mathbf{G}_1^T\xi_1 - \mathbf{G}_2^T\xi_2\right\|$, the second objective optimization problem turns into (44), as shown at the bottom of this page, by using the same trick of (31), formula (44) is equivalent to (45), as shown at the bottom of this page, and the objective value $\varphi^{(t)}$ in the *t-th* iteration is (46), as shown at the bottom of this page, thus the objective function can be simplified from division to subtraction and find out optimal solution in the $(t+1)$-*th* iteration

$$\begin{aligned}\xi_1^{(t+1)}, \xi_2^{(t+1)} = \arg\min_{\xi^{(t)}} &\sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(1)^T}\xi_1\right|^p + \sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(2)^T}\xi_2\right|^p \\ &+ \sum_{j=1}^{n_2}\varepsilon\left|\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2\right|^p \\ &+ \sum_{i=1}^{d_1+1}\delta\left|\mathbf{e}_i^{(1)^T}\xi_1\right|^p + \sum_{i=1}^{d_2+1}\delta\left|\mathbf{e}_i^{(2)^T}\xi_2\right|^p \\ &- \phi^{(t)}\Big(\sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(1)^T}\xi_1\right|^s + \sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(2)^T}\xi_2\right|^s\end{aligned} \tag{47}$$

taking the derivative of ξ_1 and ξ_2 in (47) and set them to zero

$$\begin{aligned}&\sum_{j=1}^{n_2} p\left|\mathbf{g}_j^{(1)^T}\xi_1\right|^{p-1}\operatorname{sign}(\mathbf{g}_j^{(1)^T}\xi_1)\mathbf{g}_j^{(1)} \\ &+ \sum_{i=1}^{d_1+1} p\delta\left|\mathbf{e}_i^{(1)^T}\xi_1\right|^{p-1}\operatorname{sign}(\mathbf{e}_i^{(1)^T}\xi_1)\mathbf{e}_i^{(1)} \\ &+ \sum_{j=1}^{n_2} p\varepsilon\left|\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2\right|^{p-1} \\ &\times \operatorname{sign}(\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2)\mathbf{g}_j^{(1)} \\ &- \lambda^{(t)}\sum_{i=1}^{n_1} s\left|\mathbf{h}_i^{(1)^T}\xi_1\right|^{s-1}\operatorname{sign}(\mathbf{h}_i^{(1)^T}\xi_1)\mathbf{h}_i^{(1)} = 0\end{aligned} \tag{48}$$

and

$$\begin{aligned}&\sum_{j=1}^{n_2} p\left|\mathbf{g}_j^{(2)^T}\xi_2\right|^{p-1}\operatorname{sign}(\mathbf{g}_j^{(2)^T}\xi_2)\mathbf{g}_j^{(2)} \\ &+ \sum_{i=1}^{d_2+1} p\delta\left|\mathbf{e}_i^{(2)^T}\xi_2\right|^{p-1}\operatorname{sign}(\mathbf{e}_i^{(2)^T}\xi_2)\mathbf{e}_i^{(2)} \\ &- \sum_{i=1}^{n_2} p\varepsilon\left|\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2\right|^{p-1} \\ &\times \operatorname{sign}(\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2)\mathbf{g}_j^{(2)} \\ &- \lambda^{(t)}\sum_{i=1}^{n_1} s\left|\mathbf{h}_i^{(2)^T}\xi_2\right|^{s-1}\operatorname{sign}(\mathbf{h}_i^{(2)^T}\xi_2)\mathbf{h}_i^{(2)} = 0\end{aligned} \tag{49}$$

$$\min_{\xi_1,\xi_2}\ \frac{\left\|\mathbf{G}_1^T\xi_1\right\|_p^p + \left\|\mathbf{G}_2^T\xi_2\right\|_p^p + \varepsilon\left\|\mathbf{G}_1^T\xi_1 - \mathbf{G}_2^T\xi_2\right\|_p^p + \delta\left\|\xi_1\right\|_p^p + \delta\left\|\xi_2\right\|_p^p}{\left\|\mathbf{H}_1^T\xi_1\right\|_s^s + \left\|\mathbf{H}_2^T\xi_2\right\|_s^s} \tag{44}$$

$$\begin{aligned}\min_{\xi_1,\xi_2}\ &\frac{\sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(1)^T}\xi_1\right|^p + \sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(2)^T}\xi_2\right|^p + \sum_{j=1}^{n_2}\varepsilon\left|\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2\right|^p}{\sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(1)^T}\xi_1\right|^s + \sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(2)^T}\xi_2\right|^s} \\ &+ \frac{\sum_{i=1}^{d_1+1}\delta\left|\mathbf{e}_i^{(1)^T}\xi_1\right|^p + \sum_{i=1}^{d_2+1}\delta\left|\mathbf{e}_i^{(2)^T}\xi_2\right|^p}{\sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(1)^T}\xi_1\right|^s + \sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(2)^T}\xi_2\right|^s}\end{aligned} \tag{45}$$

$$\begin{aligned}\phi^{(t)} = &\frac{\sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(1)^T}\xi_1^{(t)}\right|^p + \sum_{j=1}^{n_2}\left|\mathbf{g}_j^{(2)^T}\xi_2^{(t)}\right|^p + \sum_{j=1}^{n_2}\varepsilon\left|\mathbf{g}_j^{(1)^T}\xi_1^{(t)} - \mathbf{g}_j^{(2)^T}\xi_2^{(t)}\right|^p}{\sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(1)^T}\xi_1^{(t)}\right|^s + \sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(2)^T}\xi_2^{(t)}\right|^s} \\ &+ \frac{\sum_{i=1}^{d_1+1}\delta\left|\mathbf{e}_i^{(1)^T}\xi_1^{(t)}\right|^p + \sum_{i=1}^{d_2+1}\delta\left|\mathbf{e}_i^{(2)^T}\xi_2^{(t)}\right|^p}{\sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(1)^T}\xi_1^{(t)}\right|^s + \sum_{i=1}^{n_1}\left|\mathbf{h}_i^{(2)^T}\xi_2^{(t)}\right|^s}\end{aligned} \tag{46}$$

with the analogic alternation in (36) and (37), the derivation formula above can be described by the following equivalence forms

$$\sum_{j=1}^{n_2} \frac{2p\mathbf{g}_j^{(1)^T}\xi_1\mathbf{g}_j^{(1)}}{2\left|\mathbf{g}_j^{(1)^T}\xi_1\right|^{2-p}} + \sum_{i=1}^{d_1+1} \frac{2p\delta\mathbf{e}_i^{(1)^T}\xi_1\mathbf{e}_i^{(1)}}{2\left|\mathbf{e}_i^{(1)^T}\xi_1\right|^{2-p}} + \sum_{j=1}^{n_2} \frac{2p\varepsilon(\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2)\mathbf{g}_j^{(1)}}{2\left|\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2\right|^{2-p}} - \lambda^{(t)}\sum_{i=1}^{n_1} s\left|\mathbf{h}_i^{(1)^T}\xi_1\right|^{s-1}\mathrm{sign}(\mathbf{h}_i^{(1)^T}\xi_1)\mathbf{h}_i^{(1)} = 0 \quad (50)$$

and

$$\sum_{j=1}^{n_2} \frac{2p\mathbf{g}_j^{(2)^T}\xi_2\mathbf{g}_j^{(2)}}{2\left|\mathbf{g}_j^{(2)^T}\xi_2\right|^{2-p}} + \sum_{i=1}^{d_2+1} \frac{2p\delta\mathbf{e}_i^{(2)^T}\xi_2\mathbf{e}_i^{(2)}}{2\left|\mathbf{e}_i^{(2)^T}\xi_2\right|^{2-p}} - \sum_{j=1}^{n_2} \frac{2p\varepsilon(\mathbf{g}_j^{(1)^T}\xi_2 - \mathbf{g}_j^{(2)^T}\xi_2)\mathbf{g}_j^{(2)}}{2\left|\mathbf{g}_j^{(1)^T}\xi_2 - \mathbf{g}_j^{(2)^T}\xi_2\right|^{2-p}} - \lambda^{(t)}\sum_{i=1}^{n_1} s\left|\mathbf{h}_i^{(2)^T}\xi_2\right|^{s-1}\mathrm{sign}(\mathbf{h}_i^{(2)^T}\xi_2)\mathbf{h}_i^{(2)} = 0 \quad (51)$$

Obviously, according to the treatment of the first problem in (38) and (39), (50) and (51) can be rewritten as

$$2\mathbf{G}_1\mathbf{F}_1\mathbf{G}_1^T\xi_1 + 2\varepsilon\mathbf{G}_1\mathbf{Q}\mathbf{G}^T\xi + 2\delta\mathbf{E}_1\mathbf{O}_1\mathbf{E}_1^T\xi_1 - \phi^{(t)}\mathbf{H}_1\eta_1 = \mathbf{0} \quad (52)$$

$$2\mathbf{G}_2\mathbf{F}_2\mathbf{G}_2^T\xi_2 - 2\varepsilon\mathbf{G}_2\mathbf{Q}\mathbf{G}^T\xi + 2\delta\mathbf{E}_2\mathbf{O}_2\mathbf{E}_2^T\xi_2 - \phi^{(t)}\mathbf{H}_2\eta_2 = \mathbf{0} \quad (53)$$

where both $\mathbf{F}_1$ and $\mathbf{F}_2$ are diagonal matrixes with dimension $n_2 \times n_2$, whose the *j-th* diagonal element are $f_{j,j}^{(1)} = p/(2|\mathbf{g}_j^{(1)^T}\xi_1|^{2-p})$ and $f_{j,j}^{(2)} = p/(2|\mathbf{g}_j^{(2)^T}\xi_2|^{2-p})$, severally. Define $q_{j,j} = p/(2|\mathbf{g}_j^{(1)^T}\xi_1 - \mathbf{g}_j^{(2)^T}\xi_2|^{2-p})$ as the *j-th* diagonal entry of the second multi-view co-regularization term $\mathbf{Q} \in R^{n_2\times n_2}$. $\mathbf{O}_1 \in R^{(d_1+1)\times(d_1+1)}$ and $\mathbf{O}_2 \in R^{(d_2+1)\times(d_2+1)}$ represent diagonal matrixes with their diagonal entries as $o_{i,i}^{(1)} = p/(2|\mathbf{e}_i^{(1)^T}\xi_1|^{2-p})$ and $o_{i,i}^{(2)} = p/(2|\mathbf{e}_i^{(2)^T}\xi_2|^{2-p})$. Define η_1 with its *i-th* element as $\eta_i^{(1)} = s|\mathbf{h}_i^{(1)^T}\xi_1|^{s-1}\mathrm{sign}(\mathbf{h}_i^{(1)^T}\xi_1)$, and η_2 with its *i-th* element as $\eta_i^{(2)} = s|\mathbf{h}_i^{(2)^T}\xi_2|^{s-1}\mathrm{sign}(\mathbf{h}_i^{(2)^T}\xi_2)$. Let

$$\mathbf{M}_{G_2} = \begin{bmatrix}\mathbf{G}_1\\ -\mathbf{G}_2\end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix}\mathbf{F}_1 & \\ & \mathbf{F}_2\end{bmatrix}, \quad \mathbf{O} = \begin{bmatrix}\mathbf{O}_1 & \\ & \mathbf{O}_2\end{bmatrix}, \quad \eta = \begin{bmatrix}\eta_1\\ \eta_2\end{bmatrix} \quad (54)$$

where $\mathbf{M}_{G_1}$, $\mathbf{M}_{H_1}$ and $\mathbf{E}$ have been defined in (40). Thus combining (52) and (53) leads to

$$2\mathbf{M}_{G_1}\mathbf{F}\mathbf{M}_{G_1}^T\xi + 2\varepsilon\mathbf{M}_{G_2}\mathbf{Q}\mathbf{M}_{G_2}^T\xi + 2\delta\mathbf{E}\mathbf{O}\mathbf{E}^T\xi - \phi^{(t)}\mathbf{M}_{H_1}\eta = \mathbf{0} \quad (55)$$

whose solution is equal as the minimum problem

$$\xi^{(t+1)} = \arg\min_{\xi}\ \xi^T\mathbf{M}_{G_1}\mathbf{F}\mathbf{M}_{G_1}^T\xi + \varepsilon\xi^T\mathbf{M}_{G_2}\mathbf{Q}\mathbf{M}_{G_2}^T\xi + \delta\xi^T\mathbf{E}\mathbf{O}\mathbf{E}^T\xi - \phi^{(t)}\xi^T\mathbf{M}_{H_1}\eta \quad (56)$$

where $\mathbf{F}$, $\mathbf{Q}$, $\mathbf{O}$ and η in (56) are determined by ξ which could be computed upon the ξ that obtained in the *i-th* iteration. And both the solutions of minimization problem (43) and (56) in each iteration can be simplified into the following linear system of equations as

$$\mathbf{z}^{(t+1)} = 0.5\lambda^{(t)}(\mathbf{M}_{H_1}\mathbf{D}^{(t)}\mathbf{M}_{H_1}^T + \varepsilon\mathbf{M}_{H_2}\mathbf{V}^{(t)}\mathbf{M}_{H_2}^T + \delta\mathbf{E}\mathbf{U}^{(t)}\mathbf{E}^T)^{-1}(\mathbf{M}_{G_1}\mathbf{k}^{(t)}) \quad (57)$$

and

$$\xi^{(t+1)} = 0.5\phi^{(t)}(\mathbf{M}_{G_1}\mathbf{F}^{(t)}\mathbf{M}_{G_1}^T + \varepsilon\mathbf{M}_{G_2}\mathbf{Q}^{(t)}\mathbf{M}_{G_2}^T + \delta\mathbf{E}\mathbf{O}^{(t)}\mathbf{E}^T)^{-1}(\mathbf{M}_{H_1}\eta^{(t)}) \quad (58)$$

Now the hyperplane parameters for Lp,s-MvGEPSVM can be obtained until the algorithm converges, for given test samples $\mathbf{x}_1$ and $\mathbf{x}_2$, the vertical distances can be computed as

$$\begin{aligned} view1: \quad \mathrm{dist11} &= \frac{|\mathbf{x}_1^T\mathbf{w}_1 + b_1|}{\|\mathbf{w}_1\|}, \quad \mathrm{dist12} = \frac{|\mathbf{x}_1^T\mathbf{u}_1 + \gamma_1|}{\|\mathbf{u}_1\|} \\ view2: \quad \mathrm{dist21} &= \frac{|\mathbf{x}_2^T\mathbf{w}_2 + b_2|}{\|\mathbf{w}_2\|}, \quad \mathrm{dist22} = \frac{|\mathbf{x}_2^T\mathbf{u}_2 + \gamma_2|}{\|\mathbf{u}_2\|} \end{aligned} \quad (59)$$

then we can provide the forecasting function as same as MvGSVM [35]

$$\hat{y} = \mathrm{sign}(\mathrm{dist12} + \mathrm{dist22} - \mathrm{dist11} - \mathrm{dist21}) \quad (60)$$

where $\hat{y}$ is a prediction result of a mixed decision function that combines two views. For simplification, the algorithm we designed is described in Algorithm1.

Algorithm 1 Lp,s-MvGEPSVM

Input: Training dataset $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{B}_1$, $\mathbf{B}_2$.
Initialize $\mathbf{z}$, ξ, give model parameters (ε, δ).
Until objective converges, **do**

1. Calculate $\lambda^{(t)}$, $\phi^{(t)}$, $\mathbf{D}^{(t)}$, $\mathbf{V}^{(t)}$, $\mathbf{U}^{(t)}$, $\mathbf{k}^{(t)}$, $\mathbf{F}^{(t)}$, $\mathbf{Q}^{(t)}$, $\mathbf{O}^{(t)}$ and $\eta^{(t)}$.
2. Obtain optimal solutions by solving (43) and (56).
3. Obtain hyperplane parameters ($\mathbf{w}_1$, b_1) and ($\mathbf{w}_2$, b_2) for the first view, as well as ($\mathbf{u}_1$, γ_1) and ($\mathbf{u}_2$, γ_2) for the second view.

end
Output: Determine the attribution of class $+1$ or -1 for the test sample.

For the nonlinear case, we choose Gaussian kernel and give the derivation of kernel Lp,s-MvGEPSVM. Firstly, for a given arbitrary data set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$, the definition of Gaussian kernel is

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2/2\sigma^2), \quad i, j \in (1, 2, \ldots, m) \quad (61)$$

$$\min \frac{\left\|\left[k(\mathbf{A}_1, \mathbf{C}_1^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{w}_1\\ b_1\end{bmatrix}\right\|_p^p + \left\|\left[k(\mathbf{A}_2, \mathbf{C}_2^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{w}_2\\ b_2\end{bmatrix}\right\|_p^p + \delta\left\|\begin{bmatrix}\mathbf{w}_1\\ b_1\end{bmatrix}\right\|_p^p}{\left\|\left[k(\mathbf{B}_1, \mathbf{C}_1^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{w}_1\\ b_1\end{bmatrix}\right\|_s^s + \left\|\left[k(\mathbf{B}_2, \mathbf{C}_2^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{w}_2\\ b_2\end{bmatrix}\right\|_s^s} + \frac{\varepsilon\left\|\left[k(\mathbf{A}_1, \mathbf{C}_1^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{w}_1\\ b_1\end{bmatrix} - \left[k(\mathbf{A}_2, \mathbf{C}_2^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{w}_2\\ b_2\end{bmatrix}\right\|_p^p + \delta\left\|\begin{bmatrix}\mathbf{w}_2\\ b_2\end{bmatrix}\right\|_p^p}{\left\|\left[k(\mathbf{B}_1, \mathbf{C}_1^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{w}_1\\ b_1\end{bmatrix}\right\|_s^s + \left\|\left[k(\mathbf{B}_2, \mathbf{C}_2^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{w}_2\\ b_2\end{bmatrix}\right\|_s^s} \tag{63}$$

$$\min \frac{\left\|\left[k(\mathbf{B}_1, \mathbf{C}_1^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{u}_1\\ p_1\end{bmatrix}\right\|_p^p + \left\|\left[k(\mathbf{B}_2, \mathbf{C}_2^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{u}_2\\ p_2\end{bmatrix}\right\|_p^p + \delta\left\|\begin{bmatrix}\mathbf{u}_1\\ p_1\end{bmatrix}\right\|_p^p}{\left\|\left[k(\mathbf{A}_1, \mathbf{C}_1^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{u}_1\\ p_1\end{bmatrix}\right\|_s^s + \left\|\left[k(\mathbf{A}_2, \mathbf{C}_2^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{u}_2\\ p_2\end{bmatrix}\right\|_s^s} + \frac{\varepsilon\left\|\left[k(\mathbf{B}_1, \mathbf{C}_1^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{u}_1\\ p_1\end{bmatrix} - \left[k(\mathbf{B}_2, \mathbf{C}_2^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{u}_2\\ p_2\end{bmatrix}\right\|_p^p + \delta\left\|\begin{bmatrix}\mathbf{u}_2\\ p_2\end{bmatrix}\right\|_p^p}{\left\|\left[k(\mathbf{A}_1, \mathbf{C}_1^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{u}_1\\ p_1\end{bmatrix}\right\|_s^s + \left\|\left[k(\mathbf{A}_2, \mathbf{C}_2^T)\ \ \mathbf{e}\right]\begin{bmatrix}\mathbf{u}_2\\ p_2\end{bmatrix}\right\|_s^s} \tag{64}$$

where σ is the width of Gaussian kernel, so that the kernel generated hyperplanes can be written as

$$\begin{aligned} &view1: k(\mathbf{x}_1^T, \mathbf{C}_1^T)w_1 + b_1 = 0, \quad k(\mathbf{x}_1^T, \mathbf{C}_1^T)u_1 + p_1 = 0\\ &view2: k(\mathbf{x}_2^T, \mathbf{C}_2^T)w_2 + b_2 = 0, \quad k(\mathbf{x}_2^T, \mathbf{C}_2^T)u_2 + p_2 = 0 \end{aligned} \tag{62}$$

where $\mathbf{C}_1^T = [\mathbf{A}_1^T\ \mathbf{B}_1^T]$, $\mathbf{C}_2^T = [\mathbf{A}_2^T\ \mathbf{B}_2^T]$, so that the two minimization objective problem can be described as (63) and (64), as shown at the top of this page, then, we give these definitions

$$\begin{aligned} &\mathbf{H}_1 = [k(\mathbf{A}_1, \mathbf{C}_1^T)\ \ \mathbf{e}], \quad \mathbf{H}_2 = [k(\mathbf{A}_2, \mathbf{C}_2^T)\ \ \mathbf{e}]\\ &\mathbf{G}_1 = [k(\mathbf{B}_1, \mathbf{C}_1^T)\ \ \mathbf{e}], \quad \mathbf{G}_2 = [k(\mathbf{B}_2, \mathbf{C}_2^T)\ \ \mathbf{e}]\\ &\mathbf{z}_1 = \begin{bmatrix}w_1\\ b_1\end{bmatrix}, \quad \mathbf{z}_2 = \begin{bmatrix}w_2\\ b_2\end{bmatrix}, \quad \xi_1 = \begin{bmatrix}u_1\\ p_1\end{bmatrix}, \quad \xi_2 = \begin{bmatrix}u_2\\ p_2\end{bmatrix} \end{aligned} \tag{65}$$

where $\mathbf{z}_1, \mathbf{z}_2$ correspond to the dimension of the kernel generation metrics $\mathbf{H}_1, \mathbf{H}_2$, as well as ξ_1, ξ_2 corresponding to $\mathbf{G}_1$ and $\mathbf{G}_2$.

Thus the optimization problems can be written in the same form as (30) and (44). With the same iteration solving as the linear case, kernel Lp,s-MvGEPSVM can be described in Algorithm 2.

However, the calculating of distances from hyperplanes to samples should be redefined as

$$\begin{aligned} view1: \quad \text{dist11} &= \frac{|k(\mathbf{x}_1, \mathbf{C}_1)^T\mathbf{w}_1 + b_1|}{\|\mathbf{w}_1\|},\\ \text{dist12} &= \frac{|k(\mathbf{x}_1, \mathbf{C}_1)^T\mathbf{u}_1 + p_1|}{\|\mathbf{u}_1\|}\\ view2: \quad \text{dist21} &= \frac{|k(\mathbf{x}_2, \mathbf{C}_2)^T\mathbf{w}_2 + b_2|}{\|\mathbf{w}_2\|},\\ \text{dist22} &= \frac{|k(\mathbf{x}_2, \mathbf{C}_2)^T\mathbf{u}_2 + p_2|}{\|\mathbf{u}_2\|} \end{aligned} \tag{66}$$

and the decision function is the same as (60).

Algorithm 2 kernel Lp,s-MvGEPSVM

Input: Training dataset $\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1, \mathbf{B}_2$.
Initialize $\mathbf{z}$, ξ, give model parameters $(\varepsilon, \delta, \sigma)$.
Until objective converges, **do**

1. Define $\mathbf{H}_1, \mathbf{H}_2, \mathbf{G}_1, \mathbf{G}_2$ by using (65).
2. Calculate $\lambda^{(t)}, \phi^{(t)}$, $\mathbf{D}^{(t)}, \mathbf{V}^{(t)}, \mathbf{U}^{(t)}$, $\mathbf{k}^{(t)}$, $\mathbf{F}^{(t)}, \mathbf{Q}^{(t)}, \mathbf{O}^{(t)}$ and $\eta^{(t)}$.
3. Obtain optimal solutions by solving (43) and (56).
4. Obtain hyperplane parameters $(\mathbf{w}_1, b_1)$ and $(\mathbf{w}_2, b_2)$ for the first view, as well as $(\mathbf{u}_1, \gamma_1)$ and $(\mathbf{u}_2, \gamma_2)$ for the second view.

end
Output: Determine the attribution of class $+1$ or -1 for the test sample.

IV. ALGORITHMIC ANALYSIS

The following procedures are given to prove the convergence of the new algorithm and it is necessary to introduce the lemma as follows

Lemma 1 [52]: For any nonzero scalarsθandω, there have$p|\omega|^{p-2}|\theta|^2 - 2|\theta|^p \geq p|\omega|^{p-2}|\omega|^2 - |\omega|^p$ when $0 < p \leq 2$.

Theorem 1: Algorithm 1 will monotonically decrease the objective value of (31) in each iteration when$s \geq 1$.

Proof: According to the first optimal objective of step 3 in Algorithm 1 for each iteration we have

$$\begin{aligned} &\mathbf{z}^{(t+1)^T}\mathbf{M}_{H_1}\mathbf{D}^{(t)}\mathbf{M}_{H_1}^T\mathbf{z}^{(t+1)} + \varepsilon\mathbf{z}^{(t+1)^T}\mathbf{M}_{H_2}\mathbf{V}^{(t)}\mathbf{M}_{H_2}^T\mathbf{z}^{(t+1)}\\ &\quad + \delta\mathbf{z}^{(t+1)^T}\mathbf{E}\mathbf{U}^{(t)}\mathbf{E}^T\mathbf{z}^{(t+1)} - \lambda^{(t)}\mathbf{z}^{(t+1)^T}\mathbf{M}_{G_1}\mathbf{k}^{(t)}\\ &\leq \mathbf{z}^{(t)^T}\mathbf{M}_{H_1}\mathbf{D}^{(t)}\mathbf{M}_{H_1}^T\mathbf{z}^{(t)} + \varepsilon\mathbf{z}^{(t)^T}\mathbf{M}_{H_2}\mathbf{V}^{(t)}\mathbf{M}_{H_2}^T\mathbf{z}^{(t)}\\ &\quad + \delta\mathbf{z}^{(t)^T}\mathbf{E}\mathbf{U}^{(t)}\mathbf{E}^T\mathbf{z}^{(t)} - \lambda^{(t)}\mathbf{z}^{(t)^T}\mathbf{M}_{G_1}\mathbf{k}^{(t)} \end{aligned} \tag{67}$$

For the convenience of calculation, the inequality (61) can be rewritten as

$$\begin{aligned}&\mathbf{z}^{(t+1)^T}\mathbf{M}\Lambda\Gamma^{(t)}\mathbf{M}^T\mathbf{z}^{(t+1)} - \lambda^{(t)}\mathbf{z}^{(t+1)^T}\mathbf{M}_{G_1}\mathbf{k}^{(t)}\\ &\le \mathbf{z}^{(t)^T}\mathbf{M}\Lambda\Gamma^{(t)}\mathbf{M}^T\mathbf{z}^{(t+1)} - \lambda^{(t)}\mathbf{z}^{(t)^T}\mathbf{M}_{G_1}\mathbf{k}^{(t)}\end{aligned} \tag{68}$$

where $\mathbf{M} = [\mathbf{M}_{H_1}\ \ \mathbf{M}_{H_2}\ \ \mathbf{E}] \in R^{(d_1+d_2+2)\times n}(n = 3n_1 + d_1 + d_2 + 2)$ with the *i-th* column vector as $m_i \in R^{d_1+d_2+2}$, and

$$\Gamma^{(t)} = diag(\mathbf{D}^{(t)}, \mathbf{V}^{(t)}, \mathbf{U}^{(t)}), \quad \Lambda = diag(\mathbf{I}, \varepsilon, \delta) \tag{69}$$

both Γ and Λ are in a real space of $R^{n\times n}$, $\alpha_{i,i}$ is the *i-th* diagonal element of the scalar diagonal array Λ which is non-negative and corresponding to Γ. So, according to the definitions of $\mathbf{D}^{(t)}$, $\mathbf{V}^{(t)}$ and $\mathbf{U}^{(t)}$, we can give the new definition for $\tau_{i,i} = p/(2|\mathbf{m}_i^T\mathbf{z}|^{2-p})$ as the *i-th* diagonal element of Γ. Thus the inequality (62) can be reformulated as

$$\begin{aligned}&\frac{p}{2}\sum_{i=1}^{n}\frac{\alpha_{i,i}\mathbf{z}^{(t+1)^T}\mathbf{m}_i\mathbf{m}_i^T\mathbf{z}^{(t+1)}}{|\mathbf{m}_i^T\mathbf{z}^{(t)}|^{2-p}} - \lambda^{(t)}\\ &\quad\times s\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^{s-1}sign(\mathbf{g}_i^T\mathbf{z}^{(t)})\mathbf{g}_i^T\mathbf{z}^{(t+1)}\\ &\le \frac{p}{2}\sum_{i=1}^{n}\frac{\alpha_{i,i}\mathbf{z}^{(t)^T}\mathbf{m}_i\mathbf{m}_i^T\mathbf{z}^{(t)}}{|\mathbf{m}_i^T\mathbf{z}^{(t)}|^{2-p}} - \lambda^{(t)}\\ &\quad\times s\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^{s-1}sign(\mathbf{g}_i^T\mathbf{z}^{(t)})\mathbf{g}_i^T\mathbf{z}^{(t)}\\ &= \frac{p}{2}\sum_{i=1}^{n}\frac{\alpha_{i,i}\mathbf{z}^{(t)^T}\mathbf{m}_i\mathbf{m}_i^T\mathbf{z}^{(t)}}{|\mathbf{m}_i^T\mathbf{z}^{(t)}|^{2-p}} - \lambda^{(t)}s\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^s\end{aligned} \tag{70}$$

let $\theta = \mathbf{m}_i^T\mathbf{z}^{(t+1)}$ and $\omega = \mathbf{m}_i^T\mathbf{z}^{(t)}$ in Lemma 2, we can obtain that

$$\begin{aligned}&\frac{p}{2}\sum_{i=1}^{n}\frac{\alpha_{i,i}\mathbf{z}^{(t+1)^T}\mathbf{m}_i\mathbf{m}_i^T\mathbf{z}^{(t+1)}}{|\mathbf{m}_i^T\mathbf{z}^{(t)}|^{2-p}} - \sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t+1)}|^p\\ &\ge \frac{p}{2}\sum_{i=1}^{n}\frac{\alpha_{i,i}\mathbf{z}^{(t)^T}\mathbf{m}_i\mathbf{m}_i^T\mathbf{z}^{(t)}}{|\mathbf{m}_i^T\mathbf{z}^{(t)}|^{2-p}} - \sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t)}|^p\end{aligned} \tag{71}$$

combining (64) and (65), we arrive at the new inequality shown as

$$\begin{aligned}&\sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t+1)}|^p - \lambda^{(t)}s\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^{s-1}\\ &\qquad\times sign(\mathbf{g}_i^T\mathbf{z}^{(t)})\mathbf{g}_i^T\mathbf{z}^{(t+1)}\\ &\le \sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t)}|^p - \lambda^{(t)}s\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^s\end{aligned} \tag{72}$$

Since the function $f(\mathbf{z}) = \sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}|^s$ is convex when $s \ge 1$, and according to the work [59], for any convex function $f(\mathbf{x})$, the inequality $f(\mathbf{x}) \ge f(\mathbf{x}^{(t)}) + \nabla f(\mathbf{x})|_{\mathbf{x}=\mathbf{x}^{(t)}}(\mathbf{x} - \mathbf{x}^{(t)})$ is true, where $\nabla_{\mathbf{x}} f(\mathbf{x})|_{\mathbf{x}=\mathbf{x}^{(t)}}$ represents the gradient of $f(\mathbf{x})$ at point $\mathbf{x}^{(t)}$. Combine the property mentioned above of $f(\mathbf{z})$ and the equality $\nabla f(\mathbf{z})|_{\mathbf{z}=\mathbf{z}^{(t)}} = s\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^{s-1}sign(\mathbf{g}_i^T\mathbf{z}^{(t)})\mathbf{g}_i$, we get

$$\begin{aligned}\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t+1)}|^s \ge &\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^s\\ &+\nabla f(\mathbf{z})|_{\mathbf{z}=\mathbf{z}^{(t)}}(\mathbf{z}^{(t+1)} - \mathbf{z}^{(t)})\end{aligned} \tag{73}$$

which results in

$$\begin{aligned}\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t+1)}|^s - s\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^{s-1}sign(\mathbf{g}_i^T\mathbf{z}^{(t)})\mathbf{g}_i^T\mathbf{z}^{(t+1)}\\ \ge (1-s)\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^s\end{aligned} \tag{74}$$

By subtracting (66) and (74) from each side, we obtain

$$\begin{aligned}&\sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t+1)}|^p - \lambda^{(t)}\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t+1)}|^s\\ &\quad\le \sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t)}|^p - \lambda^{(t)}\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^s\end{aligned} \tag{75}$$

where $\lambda^{(t)} = \sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t)}|^p \Big/ \sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^s$ is the *i-th* iteration objective value, and we can obtain the following form by substituting $\lambda^{(t)}$ into the right hand in (75)

$$\begin{aligned}&\sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t+1)}|^p - \lambda^{(t)}\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t+1)}|^s\\ &\quad\le \sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t)}|^p - \sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t)}|^p = 0\end{aligned} \tag{76}$$

which leads to

$$\frac{\sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t+1)}|^p}{\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t+1)}|^s} \le \lambda^{(t)} = \frac{\sum_{i=1}^{n}\alpha_{i,i}|\mathbf{m}_i^T\mathbf{z}^{(t)}|^p}{\sum_{i=1}^{2n_2}|\mathbf{g}_i^T\mathbf{z}^{(t)}|^s} \tag{77}$$

Thus, the objective value of (30) monotonically decreases in each iteration. □

Note that the above proof is motivated by [57] and for the first objective function, the second optimal function is in the same way as the first one.

V. EXPERIMENT

In this section, we evaluate the classification performance of our proposed method on several public datasets from the UCI repository, CUHK Face Sketch database (CUFS) [67] and AR Face database, comparing with relevant algorithms GEPSVM [12], IGEPSVM [17], L1-GEPSVM [45], MvTSVM, MvGSVM [35], MvIGSVM [35]. The details of these datasets are shown in table 1.

TABLE 1. List of dataset descriptions.

Name	Classes	Numbers	Dimension
Handwritten digits	10	2000	649
Spect	2	267	44
Wpbc	2	194	33
Housingdata	2	506	13
Germ	2	1000	24
Heart	2	270	13
Ionosphere	2	351	34
Sonar	2	208	60
CUFS	2	606	1024
AR	100	2600	2200

It should be noted that GEPSVM1 and GEPSVM2 are consider as different single view of GEPSVM, in the same way, IGEPSVM1 and IGEPSVM2, L1-GEPSVM1 and L1-GEPSVM2 are seen as diverse single view of IGEPSVM and L1-GEPSVM respectively. In terms of parameter settings,

the parameter δ of Tikhonov regularization term for all methods is selected from the set $\{2^i|i = -6, -5, \ldots, 5, 6\}$, while the co-regularization parameter ε of multi-view methods is selected from the set $\{i|i = 0.25, 0.5, 0.75, 1\}$. Except for the δ and ε are set as same as above, there are two more parameter selections for Lp,s-MvGEPSVM in which p is chosen from the set $\{p|p = 0.1, 0.2, \ldots, 1.9, 2\}$ and s is chosen from the set $\{s|s = 1, 1.1, \ldots, 1.9, 2\}$. All methods obtain optimal parameters via five-fold cross validation with an exhaustive search strategy. And we also consider the predictive function of the combined views [35] which is different from the single view for both MvTSVM, MvGSVM, MvIGSVM and Lp,s-MvGEPSVM. On UCI datasets, we first compare the accuracy and robustness of our algorithm with other relevant methods, then we observe the convergence of the new iteration program and finally the impact of changes in p and s are reported. For CUHK and AR database, part B shows the performance of classification and anti-noise of Lp,s-MvGEPSVM. All experiments are implemented in MATALB 2014b on a PC with an Intel(R) Core(TM) i5-4590S, CPU @3.00GHz, RAM for 4GB.

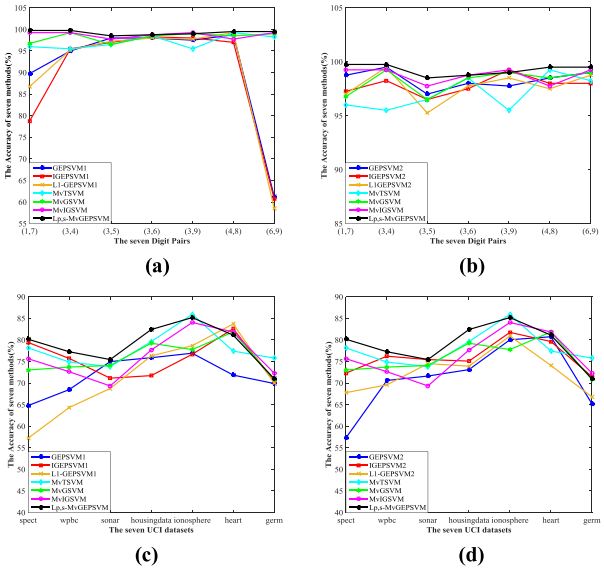


FIGURE 1. The accuracy of each algorithm on 7 Handwritten Digit Pairs and other UCI datasets.

A. UCI DATASETS

For the dataset Handwritten digits, it consists of six distinct feature sets (views) of handwritten numerals ('0'—'9') extracted from a collection of Dutch utility maps. There are 200 patterns per number (for a total of 2,000 patterns) have been digitized in binary images. In order to have comparability with previous method, we choose 76 Fourier coefficients of the character shapes and 64 Karhunen-Loéve coefficients as the first view and the second view respectively. Seven digit pairs (1, 7), (3, 4), (3, 5), (3, 6), (3, 9), (4, 8) and (6, 9) are employed to evaluate performance of all mentioned algorithms, and the range of corresponding parameter selections and the partition of datasets have been confirmed above.

As a result, Table 2 shows the average accuracy, standard deviation, and the running time of each algorithm on digit pairs datasets. Table 3 shows the performance of all methods on other seven UCI datasets (Spect, Wpbc, Housingdata, Germ, Heart, Ionosphere, Sonar), the second view are generated via PCA dimensionality reducing which contains 99% significant features, and note that the best results are shown in bold. In Figure 1 below, 1.a and 1.c are the classification accuracy on 7 digit pairs and UCI datasets in the absence of noise which display the comparison between GEPSVM1, IGEPSVM1, L1-GEPSVM1, MvTSVM, MvGSVM, MvIGSVM and Lp,s-MvGEPSVM, Figure 1.b and 1.d show the accuracy of GEPSVM2, IGEPSVM2, L1-GEPSVM2, MvTSVM, MvGSVM, MvIGSVM and Lp,s-MvGEPSVM.

From Table 2 and Table 3, we can see that the accuracy of the four multi-view methods are higher than other algorithms in most cases, which validates the classification performance of multi-view learning is better than single view learning. Owing to the multi-view learning considers the consistency and complementary of different views. There is a special situation worthy of attention in Figure 1 that the accuracy of the three single-view methods have a great divergence which may be due to the distribution of the dataset. In stark contrast MvTSVM, MvGSVM, MvIGSVM and Lp,s-MvGEPSVM still hold advantageous results, that can be explained by the stability of multi-view learning mechanism.

Separate Lp,s-MvGEPSVM out for comparison, the result of our proposed method in Figure 1.a-d is more accurate than other methods on most datasets, and we can also see a more smoother trends on different Handwritten digit pairs in Figure 1.a-b, which shows great stability and adaptability. Furthermore, the standard deviation of Lp,s-MvGEPSVM shown in Table 2 and Table 3 are smaller than other methods on most datasets, which implies its great stability and robustness again.

Since the superiority of our algorithm in terms of noise resistance theoretically, we add thirty percent Gaussian noise to the Handwritten digit pairs and twenty percent Gaussian noise to the other UCI datasets respectively to evaluate the robustness of Lp,s-MvGEPSVM, and the result is seen in Table 4 and Table 5. In addition, we conduct more detailed and persuasive experiments for Lp,s-MvGEPSVM. Firstly, MvGSVM and Lp,s-MvGEPSVM are compared on the seven UCI datasets with 5% and 10% percent noise to observe the ability of anti-noise, whose result shown in Figure 2. Next, Figure 3 displays the comparison between L1-GEPSVM and Lp,s-MvGEPSVM on the seven digit pairs, for the accuracy bar graphs with 30 percent Gaussian noise or not.

From Table 2 to Table 4, we can see the accuracy of all methods in the presence of noise decrease but Lp, s-MvGEPSVM is also the highest method that meets theoretical expectations, and the general reduction current is the most smooth in all methods.

TABLE 2. The classification accuracy on digit pairs datasets.

Datasets	(1,7) Test ± Std (%) Time(s)	(3,4) Test ± Std (%) Time(s)	(3,5) Test ± Std (%) Time(s)	(3,6) Test ± Std (%) Time(s)	(3,9) Test ± Std (%) Time(s)	(4,8) Test ± Std (%) Time(s)	(6,9) Test ± Std (%) Time(s)
GEPSVM1	89.75%±4.3598	95.00%±2.2361	98.00%±1.5000	98.00%±1.6956	97.50%±1.3693	98.75%±0.7906	61.25%±3.9528
	0.0087	0.0098	0.0092	0.0097	0.0095	0.0099	0.0092
GEPSVM2	98.75%±0.7906	99.50%±0.6124	97.00%±2.5739	98.00%±1.2748	97.75%±2.2913	98.50%±1.2247	99.00%±0.9354
	0.0069	0.0064	0.0066	0.0065	0.0067	0.0071	0.0067
IGEPSVM1	78.75%±10.7238	95.50%±2.8062	97.00%±1.2748	98.25%±1.2748	98.00%±1.6956	97.00%±0.6124	60.75%±4.2279
	0.0074	0.0077	0.0079	0.0075	0.0075	0.0076	0.0075
IGEPSVM2	97.25%±1.2247	98.25%±0.6124	96.50%±1.4577	97.50%±1.1180	**99.25%±1.0000**	98.00%±1.6956	98.00%±1.5000
	0.0071	0.0085	0.0057	0.0080	**0.0061**	0.0060	0.0063
L1-GEPSVM1	86.75%±4.5139	95.25%±1.8371	97.25%±1.4577	98.00%±1.0000	97.75%±1.6583	99.25%±0.6124	58.50%±5.2082
	0.0213	0.0224	0.0238	0.0214	0.0218	0.0272	0.0286
L1-GEPSVM1	97.00%±1.8708	99.50%±0.6124	95.25%±1.4577	97.75%±1.2247	98.50%±1.4577	97.50%±1.5811	98.75%±1.1180
	0.0167	0.0185	0.0194	0.0190	0.0178	0.0182	0.0176
MvTSVM	96.00%±2.0540	95.50%±5.4199	96.50%±3.2355	98.50%±1.6298	95.50%±2.5810	99.25%±2.7990	98.25%±2.2361
	3.0006	3.0071	2.9860	2.9085	3.1348	2.9643	3.1154
MvGSVM	96.75%±0.6124	99.25%±0.6124	96.50%±1.4577	98.50%±0.9354	99.00%±0.5000	98.50%±1.8371	99.00%±0.9354
	0.0353	0.0359	0.0364	0.0357	0.0356	0.0351	0.0358
MvIGSVM	99.25%±0.6124	99.25%±0.6124	97.75%±0.9354	**98.75%±0.7906**	**99.25%±0.6124**	97.75%±1.8371	99.25%±1.0000
	0.0238	0.0258	0.0244	**0.0232**	**0.0235**	0.0232	0.0228
Lp,s-MvGEPSVM	**99.75%±0.6124**	**99.75%±0.5000**	**98.50%±1.4577**	**98.75%±1.1180**	99.00%±1.4577	**99.50%±0.6124**	**99.50%±0.6124**
	0.1728	**0.1699**	**0.2241**	**0.2606**	0.2514	**0.2694**	**0.2423**

TABLE 3. The classification accuracy on UCI datasets.

Datasets	Spect Test ± Std (%) Time(s)	Wpbc Test ± Std (%) Time(s)	Housingdata Test ± Std (%) Time(s)	Germ Test ± Std (%) Time(s)	Ionosphere Test ± Std (%) Time(s)	Sonar Test ± Std (%) Time(s)	Heart Test ± Std (%) Time(s)
GEPSVM1	64.79%±9.6020	68.49%±6.6127	75.88%±3.3998	69.90%±1.3565	76.92%±3.6662	74.98%±6.3913	71.85%±2.7217
	0.0058	0.0036	0.0029	0.0029	0.0038	0.0024	0.0025
GEPSVM2	57.31%±3.5590	70.61%±5.0843	73.12%±4.3912	65.20%±3.9950	80.04%±4.9877	71.63%±2.4364	80.74%±1.4815
	0.0044	0.0032	0.0023	0.0028	0.0028	0.0028	0.0023
IGEPSVM1	79.41%±4.2579	75.71%±6.9466	71.73%±5.5051	70.30%±2.6192	76.67%±7.3444	71.14%±2.7656	82.59%±5.3158
	0.0031	0.0033	0.0025	0.0028	0.0034	0.0057	0.0025
IGEPSVM2	72.31%±6.4712	76.23%±5.7898	75.09%±2.6505	71.50%±4.4609	81.75%±4.0206	75.45%±4.5647	79.63%±3.3127
	0.0032	0.0026	0.0030	0.0030	0.0028	0.0032	0.0023
L1-GEPSVM1	57.25%±9.0210	64.33%±10.4471	76.29%±4.6346	70.00%±1.5811	78.66%±4.5410	68.76%±2.5032	**83.70%±3.9545**
	0.0153	0.0138	0.0155	0.0466	0.0139	0.0192	**0.0124**
L1-GEPSVM1	67.78%±4.4095	69.60%±4.3520	73.91%±4.4264	66.80%±2.3152	80.91%±4.7679	74.51%±3.9998	74.07%±5.6169
	0.0128	0.0127	0.0146	0.0541	0.0148	0.0145	0.0131
MvTSVM	78.15%±5.1387	74.87%±4.9321	79.61%±3.6278	**75.80%±4.1473**	**85.92%±4.6713**	73.81%±3.7646	77.41%±3.3127
	5.9712	1.3642	1.6709	**2.2949**	**1.8686**	1.6236	1.2357
MvGSVM	73.02%±5.7235	73.71%±3.3920	79.24%±4.9178	70.80%±4.2342	77.76%±3.2851	74.08%±4.6870	81.85%±2.4568
	0.0228	0.0203	0.0138	0.0110	0.0198	0.0204	0.0087
MvIGSVM	75.64%±5.3801	72.63%±6.0476	77.66%±4.5641	72.30%±3.8549	84.04%±1.4115	69.34%±9.2275	81.85%±2.4568
	0.0137	0.0107	0.0095	0.0104	0.0114	0.0163	0.0092
Lp,s-MvGEPSVM	**80.17%±2.6605**	**77.84%±4.4386**	**83.19%±4.6690**	71.00%±1.4832	84.90%±2.9343	**75.91%±3.7461**	81.11%±3.9545
	0.2563	**0.1311**	**0.1184**	0.3343	0.1111	**0.1412**	0.1399

In Table 5, the result shows L1-GEPSVM has lighter degradation than GEPSVM, IGEPSVM, MvGSVM and MvIGSVM. Besides, there is also a highest result on heart dataset for L1-GEPSVM, which both indicate L1-norm is more robust than the counterparts based on squared L2-norm distance measure. Figure 2 shows that MvGSVM fluctuates significantly in some data sets with the increase of noise ratio, but Lp,s-MvGEPSVM demonstrates a flat performance generally at the same time. Because Lp-norm minimization distance measures and Ls-norm

TABLE 4. Classification accuracy on Digit pairs datasets for 30% noise.

Datasets	(1,7) Test ± Std (%) Time(s)	(3,4) Test ± Std (%) Time(s)	(3,5) Test ± Std (%) Time(s)	(3,6) Test ± Std (%) Time(s)	(3,9) Test ± Std (%) Time(s)	(4,8) Test ± Std (%) Time(s)	(6,9) Test ± Std (%) Time(s)
GEPSVM1	68.25%±6.0017	60.58%±5.1602	66.73%±3.3086	62.50%±5.6067	59.81%±5.8139	43.08%±2.8782	48.65%±6.0995
	0.0079	0.0079	0.0081	0.0081	0.0088	0.0083	0.0080
GEPSVM2	76.54%±7.4750	88.65%±1.2756	86.35%±1.6543	88.27%±2.2261	87.88%±3.3640	85.19%±3.0769	86.15%±0.9806
	0.0062	0.0069	0.0067	0.0070	0.0069	0.0065	0.0071
IGEPSVM1	78.27%±1.3043	78.46%±5.5603	84.81%±1.5385	73.27%±11.3998	78.08%±7.4827	79.81%±9.4993	50.77%±4.6074
	0.0081	0.0079	0.0080	0.0076	0.0079	0.0078	0.0076
IGEPSVM2	87.69%±1.6543	86.35%±1.4132	86.54%±2.9165	86.92%±1.3043	87.69%±1.8645	87.69%±3.2408	85.38%±2.2261
	0.0056	0.0062	0.0057	0.0056	0.0058	0.0064	0.0057
L1-GEPSVM1	72.50%±2.8910	72.69%±2.1586	85.19%±3.8269	86.15%±3.0162	86.92%±1.9799	86.35%±3.0649	51.92%±5.5403
	0.0210	0.0230	0.0227	0.0220	0.0234	0.0212	0.0209
L1-GEPSVM2	86.73%±2.4627	88.08%±1.3043	84.81%±1.4132	87.69%±1.4132	86.73%±2.1414	86.92%±2.7601	89.23%±2.3077
	0.0191	0.0198	0.0198	0.0173	0.0177	0.0182	0.0206
MvTSVM	87.12%±1.6090	87.50%±0.9615	76.75%±22.3503	87.31%±1.7201	87.50%±0.9615	86.73%±3.4266	86.54%±2.6333
	3.1411	2.9863	3.1042	3.0960	2.9881	3.0631	3.0108
MvGSVM	85.00%±3.5251	88.27%±2.8782	78.46%±6.5101	87.12%±2.5512	85.19%±3.2522	81.73%±6.1418	88.27%±1.9612
	0.0346	0.0358	0.0363	0.0390	0.0363	0.0369	0.0386
MvIGSVM	83.33%±6.1802	89.38±0.7795	83.96%±4.3000	**89.38%±1.2148**	86.54%±1.3598	88.13%±2.6842	86.35%±3.5146
	0.0293	0.0285	0.0253	**0.0265**	0.0283	0.0266	0.0274
Lp,s-MvGEPSVM	**89.81%±2.5512**	**90.19%±3.4077**	**90.96%±1.4391**	89.23%±2.3865	**89.42%±1.8244**	**89.23%±1.4132**	**89.42%±1.3598**
	0.2237	**0.2033**	**0.1942**	0.2348	**0.2259**	**0.2282**	**0.2356**

TABLE 5. Classification accuracy on UCI datasets for 20% noise.

Datasets	Spect Test ± Std (%) Time(s)	Wpbc Test ± Std (%) Time(s)	Housingdata Test ± Std (%) Time(s)	Germ Test ± Std (%) Time(s)	Ionosphere Test ± Std (%) Time(s)	Sonar Test ± Std (%) Time(s)	Heart Test ± Std (%) Time(s)
GEPSVM1	58.65%±7.7994	69.82%±8.3255	66.66%±4.8095	65.50%±3.5688	71.06%±8.7999	59.23%±5.1612	55.94%±5.4453
	0.0052	0.0046	0.0044	0.0059	0.0044	0.0058	0.0080
GEPSVM2	51.13%±2.9336	70.27%±6.8665	66.51%±5.0459	61.00%±4.4845	64.61%±5.1276	56.85%±6.5499	68.44%±5.7960
	0.0048	0.0032	0.0028	0.0028	0.0039	0.0028	0.0028
IGEPSVM1	74.45%±3.8742	72.46%±3.9954	66.99%±2.3922	66.17%±2.7563	71.99%±5.6089	61.27%±4.4096	63.13%±7.8187
	0.0041	0.0033	0.0026	0.0032	0.0036	0.0056	0.0026
IGEPSVM2	68.78%±3.4966	70.71%±5.2490	68.15%±3.7736	67.25%±1.7989	71.26%±1.9179	57.66%±5.6782	71.56%±2.5000
	0.0043	0.0032	0.0025	0.0028	0.0029	0.0030	0.0023
L1-GEPSVM1	57.71%±4.9811	62.84%±5.6612	70.46%±3.8645	66.50%±1.7989	73.65%±5.1687	68.17%±4.2747	**77.19%±2.8980**
	0.0162	0.0150	0.0247	0.0624	0.0189	0.0178	**0.0194**
L1-GEPSVM2	67.83%±4.2498	71.57%±7.0868	68.64%±4.1751	62.92%±3.0732	71.99%±5.8711	63.34%±5.4256	70.94%±3.2174
	0.0242	0.0156	0.0163	0.0525	0.0161	0.0138	0.0156
MvTSVM	**77.50%±3.4266**	70.43%±3.9491	65.90%±2.9893	68.25%±1.8494	74.35%±4.0241	68.80%±5.9330	70.00%±4.6087
	6.6809	1.3268	1.7513	2.0862	1.9235	1.5946	1.2554
MvGSVM	59.60%±5.8631	68.47%±10.3894	69.64%±3.7351	67.00%±2.7437	71.06%±9.1163	58.06%±4.6858	64.06%±13.4411
	0.0145	0.0106	0.0089	0.0107	0.0142	0.0234	0.0110
MvIGSVM	69.41%±3.6660	69.79%±9.6737	71.94%±5.0017	67.75%±2.7714	74.62%±7.7575	58.98%±4.1597	71.25%±6.1397
	0.0139	0.0114	0.0092	0.0104	0.0128	0.0169	0.0113
Lp,s-MvGEPSVM	76.66%±4.5897	**74.21%±5.1670**	**75.40%±4.2642**	**69.42%±2.6431**	**78.64%±6.0978**	**71.36%±2.8248**	76.56%±3.5630
	0.1591	**0.1311**	**0.1464**	**0.6941**	**0.2054**	**0.1412**	0.1484

maximization distance measures have a strong advantage in anti-noise than squared L2-norm squared operation. According to Figure 3, we know Lp, s-MvGEPSVM is not only generally higher in accuracy than L1-GEPSVM, but also has a relatively lower change after adding noise. To sum up in this section, our proposed algorithm is confirmed to have a great robustness and stableness against noise.

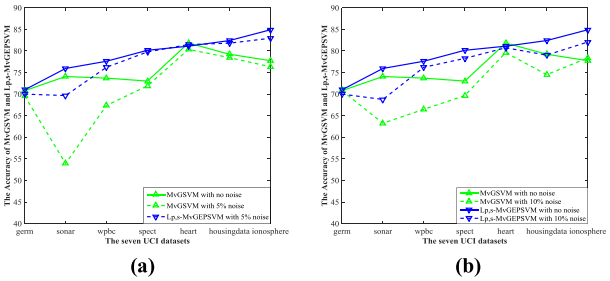


FIGURE 2. The accuracy of MvGSVM and Lp,s-MvGEPSVM on 7 UCI datasets with 0, 5% and 10% noise, respectively.

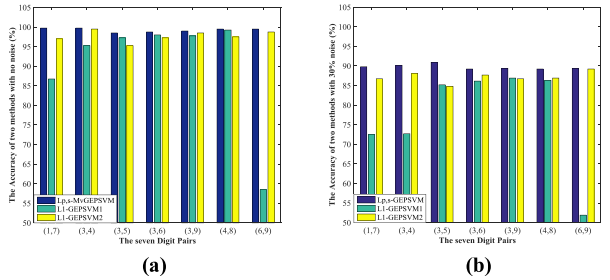


FIGURE 3. The accuracy of Lp,s-MvGEPSVM and L1-GEPSVM on 7 Digit Pairs with the comparison in the absence or presence of noise.

In addition to verifying classification performance and robustness advantages, the algorithm we designed is an iterative algorithm, convergence is the prerequisite to ensure the effectiveness of the algorithm. Despite the theoretical convergence analysis in the previous section, we also need to prove its convergence from the experiment. Several datasets are extracted randomly (digit pair (3, 4), digit pair (3, 9), wpbc dataset, sonar dataset) with the parameters $p, s, \delta, \varepsilon$ fixed, and we set the difference between the consecutive target values of the iteration is less than 0.001, the results are plotted in Figure 4. The horizontal axis represents the number of iterations and vertical axis is the difference between two continuous objective values.

From Figure 4.a-d, we can get that Lp,s-MvGEPSVM keeps monotonically decreasing along with the iteration processes until the D-value converges a small positive number close to zero, within about five to seven times, which means our algorithm is fast convergent and feasible on the computation cost.

Thus far we have described the efficiency and robustness of our proposed method. Next, we disentangle the influence of parameters on the performance. In order to verify the effect of the values of p and s on the classification accuracy, we first study the influence of p and s separately on the germ dataset, with the other three parameters fixed, and Figure 5 shows the result. Then the interaction of the values of p and s on the accuracy is tested on digit pairs (3,5) and (6,9), the classification performance is ploted in Figure 6. Finally, we compare the impact of p and s with or with no noise on the housingdata dataset. Figure 7.a shows the accuracy in the absence of noise on the housingdata dataset, and Figure 7.b shows the accuracy after introducing 20 percent Gaussian noise.

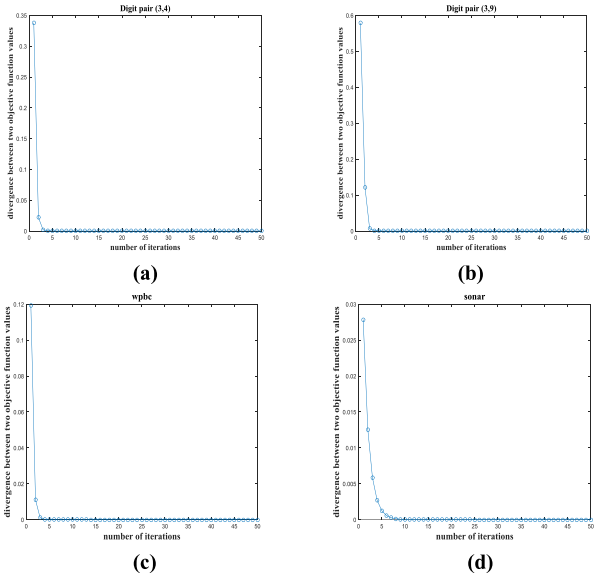


FIGURE 4. The convergence along with the number of iterations.

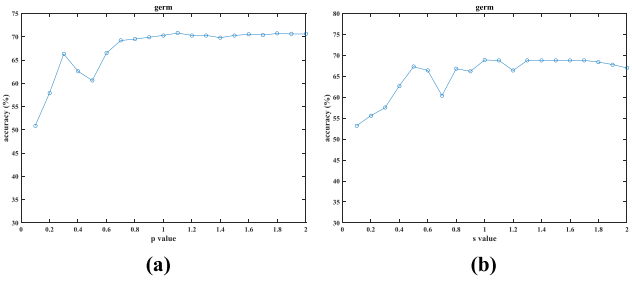
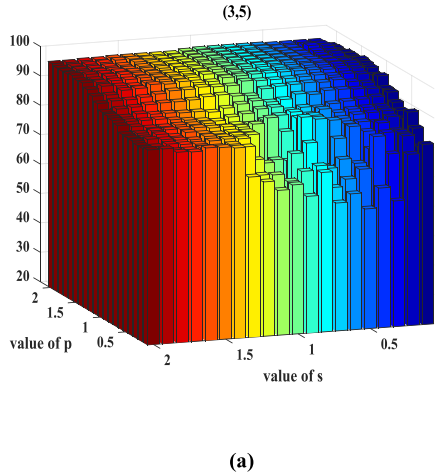


FIGURE 5. The accuracy of Lp,s-MvGEPSVM with the change of (a) the norm parameter p, and (b) the norm parameter s respectively on the germ dataset.

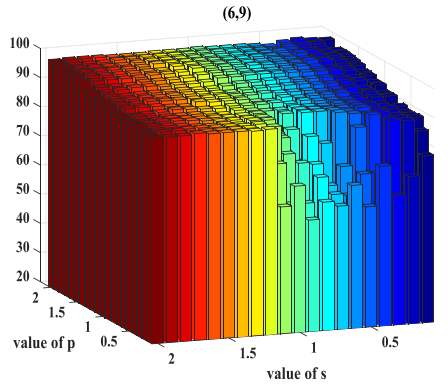
Comprehensive Figure 5 to Figure 7, it can be seen that the classification accuracy varies with the change of the p and s, which means that it is meaningful to measure the distance from hyperplanes to corresponding positive class and negative class by introducing the Lp-norm minimization and Ls-norm maximization. Specifically, with reference to Figure 5 and Figure 6, we can learn that the accuracy is lower and fluctuates greatly when p is between 0.1-0.5 and s is in the range of 0.1-1, synchronously. However, in the range of 0.5-2 for the value of p, the accuracy gradually increases with the increase of p, in the case of s value decreasing by degrees. Furthermore, an interesting result is when $p = 2$ and $s = 2$, the accuracy of Lp,s-MvGEPSVM is very close to MvGSVM, which shows the excellent flexibility of our proposed method. Finally, the comparison experiment on the housingdata dataset displays the low-order distance metric performs well against noise.

B. CUHK FACE DATASET (CUFS) and AR DATABASE

In CUFS, each face has another sketch drawn by an artist based on original photo taken in a frontal pose. So we choose 188 student faces from CUHK that taking the photos as the

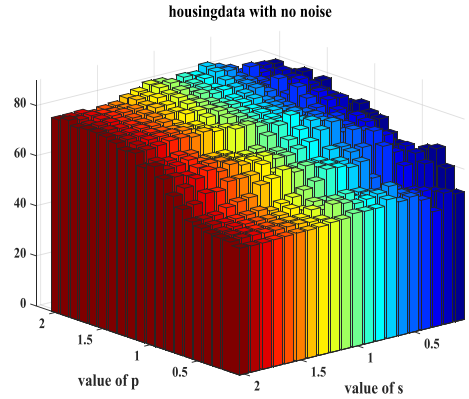


(a)

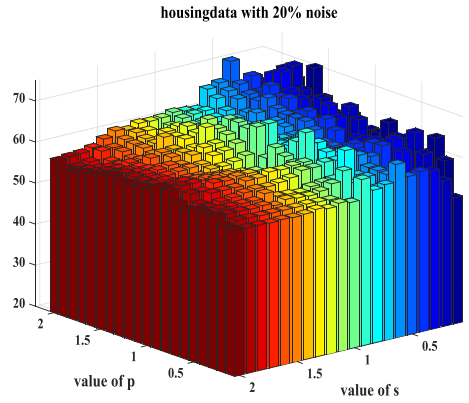


(b)

FIGURE 6. The accuracy of Lp,s-MvGEPSVM versus the variations of the parameters *p* and *s* on the digit pairs (3,5) and (6,9).



(a)



(b)

FIGURE 7. The accuracy of versus the variations of the parameters *p* and *s* on the housingdata dataset for (a) with no noise, and (b) with 20 % noise.

first view and sketches as the second view, which is shown in Figure 8.

Since the photos are under normal lighting condition, and with a neutral expression, we add random black and white block for 30 percent training face, which is displayed in Figure 9. All experimental settings has been mentioned in the beginning of this section. Table 6 and Table 7 record the experimental results, respectively, in the case of no noise and 30% noise. Figure 10 shows the performance of all methods after adding noise more intuitively.

For AR dataset, there are 26 pictures per person of 50 males and 50 females in different lighting, occlusion and expressions, so it can be seen as a noisy binary face database. Of course, the original pictures are choose to be the first view, and the second view is LBP feature of original pictures, which are exhibited in Figure 11. And Table 8 reports the experimental result. Figure 12 is the convergence trend of Lp,s-MvGEPSVM on CUFS and AR database, where the threshold is set to 0.001 and parameters fixed.

FIGURE 8. The photos and sketches of student faces from CUHK.

FIGURE 9. The examples after adding random black and white block.

From Figure 11, firstly we can learn that the accuracy of last four multi-view methods are higher than other single-view methods on noiseless faces. When adding noise, obviously, the difference between two results (the blue

TABLE 6. The classification accuracy on CUFS with no noise.

methods	GEPSVM1	GEPSVM2	IGEPSVM1	IGEPSVM2	L1-GEPSVM1	L1-GEPSVM2	MvTSVM	MvGSVM	MvIGSVM	Lp,s-MvGEPSVM
CUFS	71.19%	73.94%	87.18%	88.29%	84.01%	82.93%	92.63%	89.35%	92.53%	**96.26%**
Test(%)±Std	±18.1108	±11.0329	±4.4127	±1.3386	±5.4236	±6.4057	±3.9033	±5.3010	±3.1365	±**3.8423**
Time (s)	0.3974	0.1366	0.1369	0.1053	0.6638	0.1427	18.7546	3.0231	0.1452	**3.1797**

TABLE 7. The classification accuracy on CUFS with 30 noise.

methods	GEPSVM1	GEPSVM2	IGEPSVM1	IGEPSVM2	L1-GEPSVM1	L1-GEPSVM2	MvTSVM	MvGSVM	MvIGSVM	Lp,s-MvGEPSVM
CUFS	63.64%	62.25%	72.83%	73.91%	80.75%	82.45%	81.29%	79.64%	82.87%	**92.51%**
Test(%)±Std	±16.0288	±23.1848	±3.4432	±3.3157	±11.0940	±4.2749	±8.7888	±13.5300	±10.5372	**±2.8564**
Time (s)	0.4456	0.1588	0.1240	0.1138	0.5172	0.1538	17.9970	3.0661	0.1350	**3.4057**

TABLE 8. The classification accuracy on AR database.

methods	GEPSVM1	GEPSVM2	IGEPSVM1	IGEPSVM2	L1-GEPSVM1	L1-GEPSVM2	MvTSVM	MvGSVM	MvIGSVM	Lp,s-MvGEPSVM
AR	88.23%	72.46%	87.08%	72.69%	89.00%	81.92%	81.11%	87.12%	89.92%	**93.60%**
Test(%)±Std	±0.8720	±11.5490	±2.0975	±1.3756	±1.4766	±1.6363	±5.6890	±1.4236	±2.8331	**±3.8604**
Time (s)	0.5812	0.0066	0.3175	0.1155	30.6580	0.1732	1844	34.41	3.2535	**58.0697**

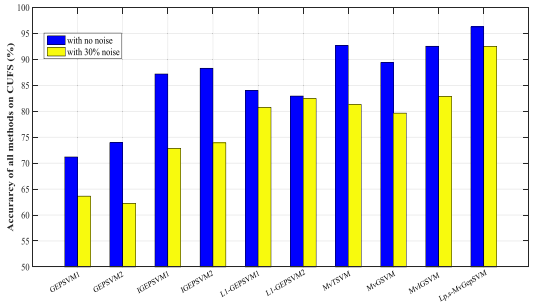


FIGURE 10. Accuracy comparison for all methods on CUFS.

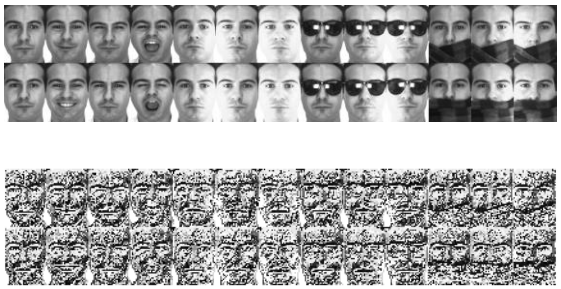


FIGURE 11. Grayscale and texture maps per person for AR.

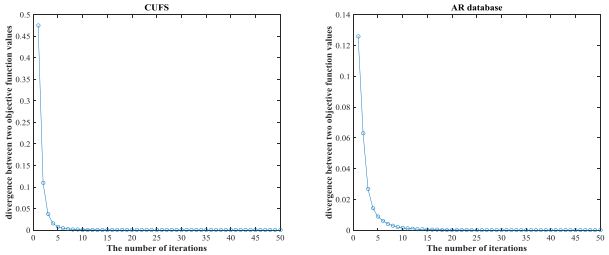


FIGURE 12. The convergence along with the number of iterations.

and yellow columns of each method) of L1-GEPSVM and Lp,s-MvGEPSVM are lower than other algorithms based on squared L2-norm distance measure, which means better robustness. In Table 6 and Table 7, the results of Lp, s-MvGEPSVM are both highest, and standard deviation get minimum after adding block noise. In Table 8, the proposed method is also the highest result. So, the experiment on CUFS and AR database demonstrate the advantages of our approach in classification accuracy, noise immunity and stability. However, despite the experimental on CUFS and AR have an ideal convergence rate, the calculation time is also the highest among all methods except MvTSVM, because of the iterative process.

In convergence of Lp,s-MvGEPSVM on CUFS and AR database, there are about ten times to converge for proposed algorithm.

VI. CONCLUSION

In this paper, a new robust multi-view generalized eigenvalue proximal SVM based on Lp-norm minimization and Ls-norm maximization distance measure is proposed termed

as Lp,s-MvGEPSVM. By setting special value of p and s, Lp,s-MvGEPSVM can be degenerate to previous versions such as MvGSVM, which represents the favorable flexibility of our method. Since our objective function is non-smooth and non-convex due to the introduction of Lp-norm and Ls-norm, we design an efficient iterative algorithm to solve this problem. Next, we give the theoretical proof of the convergence of the new algorithm. The result in a large number of experiments testifies the great generalization ability and robustness over all related methods of comparison, as well as a fast convergence speed within a finite number of times. In future work, the robust multi-view learning idea can be extended to other machine learning fields such as sparse learning which is meaningful to deal with a large-scale dataset problem. However, the major limitations of our research are the binary classification and just two views, we will extend the binary classification to multiple classification and more than two views in future work.

REFERENCES

[1] A. Ukil, "Support vector machine," *Comput. Sci.*, vol. 1, no. 4, pp. 1–28, 2002.

[2] V. N. Vapnik, "Statistical learning theory," in *Encyclopedia of the Sciences of Learning*, vol. 41, no. 4. 1998, p. 3185.

[3] N. Deng, Y. Tian, and C. Zhang, *Support Vector Machines: Optimization Based Theory, Algorithms, and Extensions*. Boca Raton, FL, USA: CRC Press, 2012.

[4] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Comput.*, vol. 10, no. 7, pp. 1895–1923, 1998.

[5] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. Annu. ACM Workshop Comput. Learn. Theory*, vol. 5, 1996, pp. 144–152.

[6] Q. Song, W. Hu, and W. Xie, "Robust support vector machine with bullet hole image classification," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 32, no. 4, pp. 440–448, Nov. 2002.

[7] P. Shih and C. Liu, "Face detection using discriminating feature analysis and support vector machine in video," in *Proc. Int. Conf. Pattern Recognit.*, Aug. 2004, pp. 407–410.

[8] B. Schïlkopf, K. Tsuda, and J. P. Vert, "Support vector machine applications in computational biology," in *Kernel Methods in Computational Biology*. Cambridge, MA, USA: MIT Press, 2004, p. 416.

[9] A. Subasi, "Classification of EMG signals using PSO optimized SVM for diagnosis of neuromuscular disorders," *Comput. Biol. Med.*, vol. 43, no. 5, pp. 576–586, 2013.

[10] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, pp. 389–422, Jan. 2002.

[11] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, "Nonlinear programming: Theory and algorithms," *Technometrics*, vol. 49, no. 1, p. 105, 1994.

[12] O. L. Mangasarian and E. W. Wild, "Multisurface proximal support vector machine classification via generalized eigenvalues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 69–74, Jan. 2006.

[13] Jayadeva, R. Khemchandani, and S. Chandra, "Twin support vector machines for pattern classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 905–910, May 2007.

[14] Jayadeva, R. Khemchandani, and S. Chandra, "Fuzzy multi-category proximal support vector classification via generalized eigenvalues," *Soft Comput.*, vol. 11, no. 7, pp. 679–685, 2007.

[15] X. Yang, S. Chen, and Y.-M. Yang, "Localized proximal support vector machine via generalized eigenvalues," *Chin. J. Comput.-Chin. Ed.*, vol. 30, no. 8, pp. 1227–1234, 2007.

[16] Q. Ye and N. Ye, "Improved proximal support vector machine via generalized eigenvalues," in *Proc. Int. Joint Conf. Comput. Sci. Optim.*, 2009, pp. 705–709.

[17] Y.-H. Shao, N.-Y. Deng, W.-J. Chen, and Z. Wang, "Improved Generalized Eigenvalue Proximal Support Vector Machine," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 213–216, Mar. 2013.

[18] W. J. Chen, Y. H. Shao, D. K. Xu, and Y. F. Fu, "Manifold proximal support vector machine for semi-supervised classification," *Appl. Intell.*, vol. 40, no. 4, pp. 623–638, 2014.

[19] M. H. Marghny and R. M. A. El-Aziz, "Differential search algorithm-based parametric optimization of fuzzy generalized eigenvalue proximal support vector machine," *Int. J. Comput. Appl.*, vol. 108, no. 19, pp. 38–46, 2015.

[20] J. Liang, F. Y. Zhang, X. X. Xiong, X. B. Chen, L. Chen, and G. H. Lan, "Manifold regularized proximal support vector machine via generalized eigenvalue," *Int. J. Comput. Intell. Syst.*, vol. 9, no. 6, pp. 1041–1054, 2016.

[21] M. R. Guarracino, M. Sangiovanni, G. Severino, G. Toraldo, and M. Viola, "On the regularization of generalized eigenvalues classifiers," in *Proc. AIP Conf.*, vol. 1776, no. 1, 2016, pp. 273–297.

[22] R. Khemchandani, K. Goyal, and S. Chandra, "Generalized eigenvalue proximal support vector regressor for the simultaneous learning of a function and its derivatives," *Int. J. Mach. Learn. Cybern.*, vol. 9, no. 12, pp. 2059–2070, May 2017.

[23] S. Ding, X. Zhang, Y. An, and Y. Xue, "Weighted linear loss multiple birth support vector machine based on information granulation for multi-class classification," *Pattern Recognit.*, vol. 67, pp. 32–46, Jul. 2017.

[24] X. Zhang, S. Ding, and Y. Xue, "An improved multiple birth support vector machine for pattern classification," *Neurocomputing*, vol. 252, pp. 119–128, Feb. 2017.

[25] S. Sun, "A survey of multi-view machine learning," *Neural Comput. Appl.*, vol. 23, nos. 7–8, pp. 2031–2038, 2013.

[26] X. Zhu, X. Li, and S. Zhang, "Block-row sparse multiview multilabel learning for image classification," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 450–461, Feb. 2015.

[27] S. Sun, F. Jin, and W. Tu, "View construction for multi-view semi-supervised learning," in *Advances in Neural Networks* (Lecture Notes in Computer Science), vol. 6675. Berlin, Germany: Springer, 2011, pp. 595–601.

[28] V. Sindhwani, P. Niyogi, and M. Belkin, "A co-regularization approach to semi-supervised learning with multiple views," in *Proc. ICML 22nd Workshop Learn. Multiple Views*, 2005, pp. 824–831.

[29] V. Sindhwani and D. S. Rosenberg, "An RKHS for multi-view learning and manifold co-regularization," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 976–983.

[30] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Comput.*, vol. 16, pp. 2639–2664, Dec. 2004.

[31] J. Farquhar, D. R. Hardoon, H. Meng, J. Shawe-Taylor, and S. Szedmák, "Two view learning: SVM-2K, theory and practice," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, vol. 18, 2006, pp. 355–362.

[32] V. Sindhwani, P. Niyogi, and M. Belkin, "A co-regularized approach to semi-supervised learning," in *Proc. ICML Workshop Learn. Multiple Views*, 2005, pp. 74–79.

[33] Z. Qi, Y. Tian, and Y. Shi, "Laplacian twin support vector machine for semi-supervised classification," *Neural Netw.*, vol. 35, no. 11, pp. 46–53, Nov. 2012.

[34] S. Sun, "Multi-view Laplacian support vector machines," in *Advanced Data Mining and Applications* (Lecture Notes in Computer Science), vol. 7121, no. 1. 2011, pp. 209–222.

[35] S. Sun, X. Xie, and C. Dong, "Multiview learning with generalized eigenvalue proximal support vector machines," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 688–697, 2018.

[36] J. Tang, Y. Tian, P. Zhang, and X. Liu, "Multiview privileged support vector machines," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3463–3477, Aug. 2018.

[37] J. Tang, D. Li, Y. Tian, and D. Liu, "Multi-view learning based on nonparallel support vector machine," *Knowl.-Based Syst.*, vol. 158, pp. 94–108, Oct. 2018.

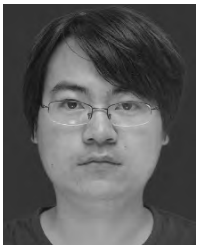
[38] N. Zhang, S. Ding, H. Liao, and W. Jia, "Multimodal correlation deep belief networks for multi-view classification," *Appl. Intell.*, vol. 49, no. 5, pp. 1925–1936, 2018.

[39] Y. He, Y. Tian, and D. Liu, "Multi-view transfer learning with privileged learning framework," *Neurocomputing*, vol. 335, pp. 131–142, Mar. 2019.

[40] M. White, Y. Yu, X. Zhang, D. Schuurmans, and Y.-L. Yu, "Convex multi-view subspace learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.

[41] Z. Xu and S. Sun, "An algorithm on multi-view Adaboost," in *Neural Information Processing. Theory and Algorithms* (Lecture Notes in Computer Science), vol. 6443. 2010, pp. 355–362.
[42] A. Kumar, P. Rai, and H. Daumé, "Co-regularized multi-view spectral clustering," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, vol. 24, 2011, pp. 1413–1421.
[43] S. Gao, Q. Ye, and N. Ye, "1-norm least squares twin support vector machines," *Neurocomputing*, vol. 74, no. 17, pp. 3590–3597, 2011.
[44] J. J. Liang and W. De, "Sparse least square support vector machine with L1 norm," *Comput. Eng. Des.*, vol. 35, no. 1, pp. 293–296, 2014.
[45] A. H. Yan, "The GEPSVM classifier based on L1-norm distance metric," in *Pattern Recognition*. Singapore: Springer, 2016.
[46] R. Yan, Q. Ye, D. Zhang, N. Ye, and X. Li, "1-norm projection twin support vector machine," in *Pattern Recognition*. Singapore: Springer, 2016.
[47] H. Yan, Q. Ye, T. Zhang, D. Yu, X. Yuan, Y. Xu, and L. Fu, "Least squares twin bounded support vector machines based on L1-norm distance metric for classification," *Pattern Recognit.*, vol. 74, pp. 434–447, Feb. 2018.
[48] C.-N. Li, Y.-H. Shao, and N.-Y. Deng, "Robust L1-norm non-parallel proximal support vector machine," *Optimization*, vol. 65, no. 1, pp. 1–15, 2016.
[49] Z. Gu, Z. Zhang, J. Sun, and B. Li, "Robust image recognition by L1-norm twin-projection support vector machine," *Neurocomputing*, vol. 223, pp. 1–11, Feb. 2017.
[50] H. Yan, Q.-L. Ye, and D.-J. Yu, "Efficient and robust TWSVM classification via a minimum L1-norm distance metric criterion," *Mach. Learn.*, vol. 108, no. 6, pp. 993–1018, 2018.
[51] H. K. Achanta, B. Misganaw, and M. Vidyasagar, "A multi-view ℓ_1-norm SVM algorithm for data integration in biological applications," in *Proc. IEEE Amer. Control Conf.*, May 2017, pp. 3753–3757.
[52] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 9, pp. 1672–1680, Sep. 2008.
[53] N. Tsagkarakis, P. P. Markopoulos, G. Sklivanitis, and D. A. Pados, "L1-norm principal-component analysis of complex data," *IEEE Trans. Signal Process.*, vol. 66, no. 12, pp. 3256–3267, Mar. 2018.
[54] S. Chamadia and D. A. Pados, "Optimal sparse L_1-norm principal-component analysis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2017, pp. 2686–2690.
[55] F. Zhong and J. Zhang, "Linear discriminant analysis based on L1-norm maximization," *IEEE Trans. Image Process.*, vol. 22, no. 8, pp. 3018–3027, Aug. 2013.
[56] Y. Liu, Q. Gao, S. Miao, X. Gao, F. Nie, and Y. Li, "A non-greedy algorithm for L1-norm LDA," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 684–695, Feb. 2017.
[57] Q. Ye, L. Fu, Z. Zhang, H. Zhao, and M. Naiem, "Lp- and Ls-norm distance based robust linear discriminant analysis," *Neural Netw.*, vol. 105, pp. 393–404, Sep. 2018.
[58] F. Nie, H. Huang, and C. Ding, "Low-rank matrix recovery via efficient schatten p-norm minimization," in *Proc. Int. Joint Conf. Artif. Intell.*, 2012, pp. 1–7.
[59] H. Wang, F. Nie, W. Cai, and H. Huang, "Semi-supervised robust dictionary learning via efficient 1norms minimization," in *Proc. IEEE Conf. CVPR*, Dec. 2013, pp. 1145–1152.
[60] N. Kwak, "Principal component analysis by lp-norm maximization," *IEEE Trans. Cybern.*, vol. 44, no. 5, pp. 594–609, Apr. 2014.
[61] J. Wang, "Generalized 2-D principal component analysis by Lp-norm for image analysis," *IEEE Trans. Cybern.*, vol. 46, no. 3, pp. 792–803, 2016.
[62] J. S. Qi, X. Liang, and R. Xu, "A multiple kernel learning model based on p-norm," in *Computational Intelligence and Neuroscience*. 2018, pp. 1–7.
[63] H. Yang, Z. Xu, J. Ye, I. King, and M. R. Lyu, "Efficient sparse generalized multiple kernel learning," *IEEE Trans. Neural Netw.*, vol. 22, no. 3, pp. 433–446, 2011.
[64] X. Q. Sun, Y. J. Chen, Y.-H. Shao, C.-N. Li, and C.-H. Wang, "Robust non-parallel proximal support vector machine with Lp-norm regularization," *IEEE Access*, vol. 6, pp. 20334–20347, 2018.
[65] A. N. Tikhonov, A. Goncharsky, V. V. Stepanov, and A. G. Yagola, *Numerical Methods for the Solution of Ill-Posed Problems*. Amsterdam, The Netherlands: Springer, 1995.
[66] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.
[67] X. Wang and X. Tang, "Face photo-sketch synthesis and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 11, pp. 1955–1967, Sep. 2009.

PENG HUANG was born in 1996. He is currently pursuing the master's degree with the College of Information Science and Technology, Nanjing Forestry University, China. His main research interests include pattern recognition, machine learning, and data mining.



QIAOLIN YE received the B.S. degree in computer science from the Nanjing Institute of Technology, Nanjing, China, in 2007, the M.S. degree in computer science and technology from Nanjing Forestry University, Nanjing, in 2009, and the Ph.D. degree in pattern recognition and intelligence system from the Nanjing University of Science and Technology, China, in 2013. He is currently an Associate Professor with the Computer Science Department, Nanjing Forestry University, Nanjing. He has authored over 50 scientific papers. Some of them are published in the IEEE TNNLS, the IEEE TIFS, and the IEEE TCSVT. His research interests include machine learning, data mining, and pattern recognition.

YAN LI was born in 1998. She is currently pursuing the bachelor's degree with the College of Information Science and Technology, Nanjing Forestry University, China. Her main research interests include pattern recognition and data mining.

GUOWEI YANG received the B.S. and M.S. degrees in mathematics from Jiangxi Normal University, Nanchang, China, in 1985 and 1988, respectively, and the Ph.D. degree from the University of Science and Technology Beijing, China, in 2004. He was a Professor with Qingdao University, China, in 1999. He is currently a Professor with Qingdao University and also with Nanjing Audit University. His current research interests include artificial intelligence, intelligent information processing, artificial neural networks, and pattern recognition.

YINGAN LIU received the B.S. degree in mathematics from Anhui Normal University, Wuhu, China, in 1988, the M.S. degree in forestry engineering from Nanjing Forestry University, Nanjing, China, in 1991, and the Ph.D. degree in systems engineering from the Southeast University, China, in 2003. He has been a Professor at Nanjing Forestry University, Nanjing, since 2006. He has authored over 30 scientific papers. His research interests include statistical diagnosis, data mining, and forestry statistical analysis.