F-SVM: Combination of Feature Transformation and SVM Learning via Convex Relaxation

Xiaohe Wu, Wangmeng Zuo, *Senior Member, IEEE*, Liang Lin, *Senior Member, IEEE*, Wei Jia, *Member, IEEE*, and David Zhang, *Fellow, IEEE*

***Abstract*— The generalization error bound of the support vector machine (SVM) depends on the ratio of the radius and margin. However, conventional SVM only considers the maximization of the margin but ignores the minimization of the radius, which restricts its performance when applied to joint learning of feature transformation and the SVM classifier. Although several approaches have been proposed to integrate the radius and margin information, most of them either require the form of the transformation matrix to be diagonal, or are nonconvex and computationally expensive. In this paper, we suggest a novel approximation for the radius of the minimum enclosing ball in feature space, and then propose a convex radius-margin-based SVM model for joint learning of feature transformation and the SVM classifier, i.e., F-SVM. A generalized block coordinate descent method is adopted to solve the F-SVM model, where the feature transformation is updated via the gradient descent and the classifier is updated by employing the existing SVM solver. By incorporating with kernel principal component analysis, F-SVM is further extended for joint learning of nonlinear transformation and the classifier. F-SVM can also be incorporated with deep convolutional networks to improve image classification performance. Experiments on the UCI, LFW, MNIST, CIFAR-10, CIFAR-100, and Caltech101 data sets demonstrate the effectiveness of F-SVM.**

***Index Terms*— Convex relaxation, max margin, radius-margin error bound, support vector machine (SVM).**

I. Introduction

THE support vector machine (SVM) and its extensions are one class of the most successful machine learning methods [1], and have been widely adopted in various application fields [2], [3]. Actually, SVM aims to seek the optimal hyperplane with the maximum margin principle, but the generalization error of SVM actually is a function of the ratio of the radius and margin, i.e., radius-margin error bound [4]. When feature mapping is given, the radius is fixed and can be ignored, and thus SVM can safely minimize the generalization error by maximizing the margin. However, for joint learning of feature transformation and the classifier, the radius information is valuable and cannot be ignored.

Given a sample $\mathbf{x}$, the feature transformation is defined as a linear projection $\mathbf{Ax}$, where $\mathbf{A}$ is the transformation matrix. Denote by $(\mathbf{u}, b)$ a linear classifier. The radius-margin error bound can then be utilized to guide the joint learning of feature transformation $\mathbf{A}$ and classifier $(\mathbf{u}, b)$, resulting in the classifier $\mathbf{u}^\top\mathbf{Ax} + b$. When the matrix $\mathbf{A}$ is constrained to be diagonal, it becomes a joint feature weighting and classifier learning problem [2]. Since the radius-margin error bound is nonconvex, relaxation and approximation of the radius are generally adopted in the existing models [5], [6]. Several approaches have been proposed from the perspective of the radius-margin error [2], [5]–[7], but most suffer from the limitations of computational burden and simplified forms of transformation. Relative margin machine (RMM) [5] only considers the spread of the data along the direction perpendicular to the classification hyperplane. Radius-margin-based SVMs, e.g., margin-radius SVM (MR-SVM) [2], metric learning-based radius-margin SVM (R-SVM$^+$), and radius-margin SVM for feature selection (RSVM$^+_{\boldsymbol{\mu}}$) [7], are only applicable to feature weighting and selection.

Another strategy is to incorporate metric learning with SVM. Metric learning can be adopted to learn a better linear transformation matrix [6], [8], [9]. One simple approach to combine metric learning and SVM is to directly deploy the transformation obtained using metric learning into SVM. This approach, however, usually cannot lead to a satisfying performance [10]. Therefore, other approaches have been proposed to integrate metric learning into SVM, e.g., support vector metric learning (SVML) [10] and metric learning with SVM (MSVM) [6]. But SVML [10] is designed for SVM with Gaussian radial basis function kernel (RBF-SVM) and ignores the radius information, while MSVM [6] is nonconvex.

In this paper, we propose a novel radius-margin-based SVM model for joint learning of feature transformation and the SVM classifier, i.e., F-SVM. Compared with existing radius-margin-based SVM methods, we derive novel lower and upper bounds for the relaxation of the radius. Unlike MR-SVM [2], R-SVM$^+$ and R-SVM$^+_\mu$ [7] which are suggested for joint feature weighting and SVM learning, F-SVM can simultaneously learn feature transformation $\mathbf{M} = \mathbf{A}^\top\mathbf{A}$ and the classifier $(\mathbf{w}, b)$. Compared with the existing metric learning for SVM methods, our F-SVM model considers both the radius and the margin information. Compared with MSVM [6] which aims to learn $\mathbf{A}$ and $(\mathbf{u}, b)$, and is nonconvex, our F-SVM jointly

Manuscript received June 25, 2016; revised April 14, 2017 and October 23, 2017; accepted January 3, 2018. Date of publication February 5, 2018; date of current version October 16, 2018. This work was supported in part by the National Defense Science and Technology Innovation Special Zone Project of China under Grant 17-163-11-ZT-003-024-01, and in part by the National Science Foundation of China under Grant 61671182, Grant 61271093, and Grant 61673157. *(Corresponding author: Wangmeng Zuo.)*

X. Wu and W. Zuo are with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: xhwu.cpsl.hit@gmail.com; cswmzuo@gmail.com).

L. Lin is with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China (e-mail: linliang@ieee.org).

W. Jia is with the School of Computer and Information, Hefei University of Technology, Hefei 230031, China (e-mail: jiawei@hfut.edu.cn).

D. Zhang is with the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen 518172, China (e-mail: csdzhang@comp.polyu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2018.2791507

learns $\mathbf{M}$ and $(\mathbf{w}, b)$. And the united inequality constraint is further introduced to improve the robustness and to reduce the computational budget. Benefitted from the introduction of $\mathbf{M}$ and united inequality constraint, we present a convex model for joint feature transformation and classifier learning. A generalized coordinate descent (GBCD) algorithm is proposed to solve our F-SVM model, which iterates alternately by updating the feature transformation and classifier. Note that kernel SVM is equivalent to performing linear SVM in the kernel principal component analysis (PCA) space. We further extend linear F-SVM in the kernel PCA space for joint learning of the nonlinear transformation and classifier. Experiments have been conducted on the 20 UCI data sets and the LFW database. The results show that F-SVM outperforms SVM and the existing radius-margin-based SVMs. Furthermore, we incorporate F-SVM with deep convolutional networks (CNNs) for image classification, and achieve state-of-the-art performance on the MNIST, CIFAR-10, CIFAR-100 and Caltech101 data sets. To sum up, the main contributions of this paper are four fold.

1) Novel lower and upper bounds are derived for the radius of the minimum enclosing ball (MEB). The bounds not only offer a novel approximation of the radius, but also lay a solid theoretical foundation to our F-SVM model.
2) A novel convex formulation of a radius-margin-based SVM model, i.e., F-SVM, is proposed. In F-SVM, all the constraints on distance are aggregated into one *united inequality constraint*. Instead of learning $\mathbf{A}$, our F-SVM jointly learns $\mathbf{M} = \mathbf{A}^\top\mathbf{A}$ and $(\mathbf{w}, b)$, and thus can be formulated into a convex program. To the best of our knowledge, F-SVM is the first convex model for joint learning of feature transformation and SVM classifier.
3) Benefitted from the united inequality constraint, we develop a semiwhitened PCA method for initializing $\mathbf{M}$. A generalized block coordinate descent (GBCD) algorithm is suggested to solve our F-SVM model. GBCD can converge to the global optimum, and is much more efficient than RMM [5], R-SVM$^+$, and R-SVM$^+_{\boldsymbol{\mu}}$ [7] in training.
4) By revealing the equivalence of kernel SVM and linear SVM in kernel PCA space, we further suggest a *kernel F-SVM* model by conducting linear F-SVM in the kernel PCA space.

The remainder of this paper is organized as follows. Section II reviews the related work on radius-margin-based SVM methods. Section III describes the model and algorithm of F-SVM. Section IV extends F-SVM to the kernelized version for nonlinear classification. Section V provides the experimental results on the UCI, LFW, MNIST, CIFAR-10, CIFAR-100, and Caltech101 data sets. Finally, the conclusions are drawn in Section VI.

II. Related Work

The radius-margin error bound not only provides a theoretical explanation of the generalization performance of SVM [1], but also has been extensively adopted for improving kernel classification methods, e.g., model selection [11] and multiple kernel learning [12], [13]. Denote by $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ a training set, where $\mathbf{x}_i \in \mathbf{R}^d$ stands for the ith training sample and $y_i \in \{-1, +1\}$ stands for the corresponding class label of $\mathbf{x}_i$. Given a mapping $\Phi : \mathbf{x} \mapsto \mathcal{H}$ to map the sample $\mathbf{x}$ to some feature space $\mathcal{H}$, the radius R of the MEB [12] is defined as

$$\min_{R,\mathbf{x}_0} R^2, \quad \text{s.t. } \|\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_0)\|_2^2 \le R^2 \quad \forall i. \tag{1}$$

Suppose that the training set is linearly separable in the feature space with the optimal hyperplane defined by $\mathbf{u}^\top\Phi(\mathbf{x})+b = 0$. Vapnik [1] shows that the expectation of the misclassification probability depends not only on the margin but also on the radius, and is bounded by the function of $R^2\|\mathbf{u}\|_2^2$.

The SVM is well known as a max-margin model, which only considers the margin $1/\|\mathbf{u}\|_2^2$. When the feature space is fixed, the radius is a constant and can thus be safely ignored. But in many learning tasks, the model parameters [11], combination of basis kernels [12], feature reweighting, or transformation should usually be learned or tuned based on the training data by incorporating both margin and radius information [7], [13], [14].

This paper aims to jointly learn SVM together with the feature transformation by minimizing the radius-margin ratio, i.e., $R^2\|\mathbf{u}\|_2^2$. Thus, a more detailed review is given on this topic. Except for [6], most existing approaches [2], [7] require the transformation matrix to be diagonal, and thus are only applicable to feature reweighting and selection. Direct use of the radius-margin ratio in SVM results in a nonconvex optimization problem, which makes the learning algorithm computationally expensive and unstable. In feature reweighting and selection, the feature transformation matrix should be diagonal, i.e., $\mathbf{D}_{\sqrt{\boldsymbol{\mu}}} = \text{Diag}\{\sqrt{\boldsymbol{\mu}}\}$ with $\sqrt{\boldsymbol{\mu}} = [\sqrt{\mu_1}, \ldots, \sqrt{\mu_k}, \ldots, \sqrt{\mu_d}]^\top$, where $\sqrt{\mu_k}$ is a scaling factor for the kth feature.

Do *et al.* [2] suggest that the radius is bounded with $\max_k \mu_k R_k^2 \le R_{\boldsymbol{\mu}}^2 \le \sum_k \mu_k R_k^2$, where R_k is the radius of dimension k. By approximating $R_{\boldsymbol{\mu}}^2$ with its upper bound $\sum_k \mu_k R_k^2$, MR-SVM [2] solves the following convex relaxation problem:

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi},\boldsymbol{\mu}} \ & \frac{1}{2}\sum_k \frac{w_k^2}{\mu_k} + \frac{C}{\sum_k \mu_k R_k^2}\sum_i \xi_i^2 \\ \text{s.t. } & y_i(\mathbf{w}^\top\mathbf{x}_i + b) \ge 1 - \xi_i \quad \forall i \\ & \sum_k \mu_k = 1, \mu_k \ge 0 \quad \forall k \end{aligned} \tag{2}$$

where $\mathbf{w} = [w_1, w_2, \ldots, w_d]^\top$ is the normal vector to the classification hyperplane, and $b/\|\mathbf{w}\|_2$ is the offset of the hyperplane from the origin along $\mathbf{w}$. ξ_i denotes the ith slack variable, and C stands for the tradeoff parameter. R_O is denoted by the half value of the maximum pairwise distance. Do and Kalousis [7] introduce a tighter bound of the radius $R_O \le R_{\boldsymbol{\mu}} \le ((1+\sqrt{3})/2)R_O$, and propose another convex model, i.e., R-SVM$^+_{\boldsymbol{\mu}}$

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi},\boldsymbol{\mu},r} \ & \frac{1}{2}\sum_k \frac{w_k^2}{\mu_k} + \lambda r + C\sum_i \xi_i \\ \text{s.t. } & y_i(\mathbf{w}^\top\mathbf{x}_i + b) \ge 1 - \xi_i \quad \forall i \\ & \xi_i \ge 0, \quad i = 1, 2, \ldots, n \\ & \sum_k \mu_k = 1, \mu_k \ge 0 \quad \forall k \\ & \frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j)^\top\mathbf{D}_{\boldsymbol{\mu}}(\mathbf{x}_i - \mathbf{x}_j) \le r \quad \forall i, j \end{aligned} \tag{3}$$

where $\mathbf{D}_{\boldsymbol{\mu}} = \text{Diag}\{\boldsymbol{\mu}\}$. Furthermore, R-SVM$^+$ [7] is developed by controlling both the radius and margin with $\mathbf{w}$.

Zhu *et al.* [6] propose a metric learning with the SVM (MSVM) method for joint learning of the linear transformation and SVM classifier. Given the transformation matrix $\mathbf{A}$, an alternative $\bar{R} = \max_i \|\mathbf{A}\mathbf{x}_i - \mathbf{A}\bar{\mathbf{x}}\|_2^2$ of the radius R is adopted, where $\bar{\mathbf{x}}$ is the mean of the training samples. Although Zhu *et al.* [6] claim that $R = \bar{R}$, as demonstrated in **Theorem 1** of this paper, $\bar{R}$ is an upper bound of R. The MSVM model [6] is formulated as

$$\begin{aligned} \min_{\mathbf{u},b,\boldsymbol{\xi},\mathbf{A}} \ & \frac{1}{2}\|\mathbf{u}\|_2^2 + C\sum_i \xi_i^2 \\ \text{s.t. } & y_i(\mathbf{u}^\top \mathbf{A}\mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \|\mathbf{A}\mathbf{x}_i - \mathbf{A}\bar{\mathbf{x}}\|^2 \leq 1 \quad \forall i. \end{aligned} \tag{4}$$

MSVM is nonconvex and can be solved using gradient projection.

In this paper, we propose a novel relaxed convex model of the radius-margin-based SVM, i.e., F-SVM, for joint learning of the feature transformation and SVM classifier. Compared with existing radius-margin-based SVM methods, F-SVM has some distinguishing advantages. MSVM [6] is nonconvex, while our F-SVM model is convex and our GBCD algorithm converges to the global optimum. Unlike RMM [5], the transformation in F-SVM is learned to minimize the radius of the enclosing ball of all samples rather than to only shrink the sample spanned along the direction perpendicular to the hyperplane. Moreover, F-SVM is also different from MR-SVM [2], R-SVM$^+$, and R-SVM$_{\mu}^+$ [7] from three aspects.

1) Instead of feature weighting, F-SVM can simultaneously learn the feature transformation and classifier.
2) F-SVM adopts a new approximation for the radius of MEB in feature space.
3) In F-SVM, individual inequality constraints are combined into one holistic inequality constraint to improve the robustness and training efficiency.

All these make F-SVM very promising for joint learning of the feature transformation and SVM classifier, and the experimental results further validate the effectiveness of F-SVM.

III. Radius-Margin-Based Support Vector Machine

A. Problem Formulation

Given the training set $\mathcal{S}$, by introducing the slack variables $\xi_i (i = 1, 2, \ldots, n)$, the SVM can be formulated as

$$\begin{aligned} \min_{\mathbf{u},b,\boldsymbol{\xi}} \ & \frac{1}{2}\|\mathbf{u}\|_2^2 + C\sum_i \xi_i \\ \text{s.t. } & y_i(\mathbf{u}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, n \end{aligned} \tag{5}$$

where $(\mathbf{u}, b)$ are the parameters used to describe the learned hyperplane $\mathbf{u}^\top \mathbf{x} + b = 0$. The objective function in (5) aims to maximize the margin $\gamma = 1/\|\mathbf{u}\|^2$ while minimizing the empirical risk $\sum_{i=1}^n \xi_i$. For joint learning, we introduce a linear transformation matrix $\mathbf{A}$ and integrate the radius information, resulting in the following radius-margin-based SVM model:

$$\begin{aligned} \min_{\mathbf{u},b,\boldsymbol{\xi},\mathbf{A},R} \ & \frac{1}{2}\|\mathbf{u}\|_2^2 R^2 + C\sum_i \xi_i \\ \text{s.t. } & y_i(\mathbf{u}^\top \mathbf{A}\mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, n \end{aligned} \tag{6}$$

where the radius R is defined as

$$\min_{R,\mathbf{x}_0} R^2, \quad \text{s.t. } \|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_0\|_2^2 \leq R^2, \quad i = 1, 2, \ldots, n. \tag{7}$$

Note that R^2 depends on matrix $\mathbf{A}$ and the problem in (6) is nonconvex [7]. We introduce $\bar{\mathbf{x}}$ to denote the mean vector of the training samples, i.e., $\bar{\mathbf{x}} = (1/n)\sum_{i=1}^n \mathbf{x}_i$, and $\bar{R}$ to denote the largest Euclidean distance between the training samples and the mean vector in the transformation space, i.e., $\bar{R} = \max_i \|\mathbf{A}\mathbf{x}_i - \mathbf{A}\bar{\mathbf{x}}\|_2^2$. Then, it can be proved that the radius R is bounded by $\bar{R}$.

Theorem 1: The radius R is bounded by $\bar{R}$ by

$$\frac{1}{2}\overline{R} \leq R \leq \overline{R}. \tag{8}$$

Please refer to **Appendix A** the proof of **Theorem 1**. $e = \bar{R} - R$ is denoted as the error of approximation. From **Theorem 1**, we have $0 \leq e \leq R$, and thus $\bar{R}$ can serve as a reasonable approximation with theoretical guarantee. Zhu *et al.* [6] claim that $R = \bar{R}$. From **Theorem 1**, $\bar{R}$ is only an approximation of R, and counter examples can be easily found to illustrate $R \neq \bar{R}$. Let $\mathbf{w} = \mathbf{A}^\top \mathbf{u}$ and $\mathbf{M} = \mathbf{A}^\top \mathbf{A}$ which is positively definite. Since the radius R is upper bounded by $\bar{R}$, we can approximate R with $\bar{R}$. With simple algebra, the radius-margin SVM model in (6) is relaxed into the following formulation:

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi},\mathbf{M},\bar{R}} \ & F(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{M}, \bar{R}) = \frac{1}{2}(\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w})\bar{R}^2 + C\sum_{i=1}^n \xi_i \\ \text{s.t. } & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, n \\ & (\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M}(\mathbf{x}_i - \bar{\mathbf{x}}) \leq \bar{R}^2. \end{aligned} \tag{9}$$

Theorem 2: The problem in (9) is equivalent to the following problem:

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi},\mathbf{M}} \ & L(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{M}) = \left\{ \frac{1}{2}(\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}) + C\sum_{i=1}^n \xi_i \right\} \\ \text{s.t. } & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, n \\ & (\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M}(\mathbf{x}_i - \bar{\mathbf{x}}) \leq 1 \quad \forall i \\ & \mathbf{M} \succ 0. \end{aligned} \tag{10}$$

Proof: Denote by $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \hat{\mathbf{M}}, \hat{R})$ the optimal solution to the problem in (9). Let $\tilde{\mathbf{M}} = \hat{\mathbf{M}}/\hat{R}^2$ and $\tilde{R} = 1$. It is obvious that $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}}, \tilde{R})$ is also the optimal solution to the problem in (9) because $F(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \hat{\mathbf{M}}, \hat{R}) = F(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}}, \tilde{R})$.

Next we will show that $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}})$ is the optimal solution to the problem in (10). If $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}})$ is not the optimal solution to (10), there must exist some $(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*, \mathbf{M}^*)$ that

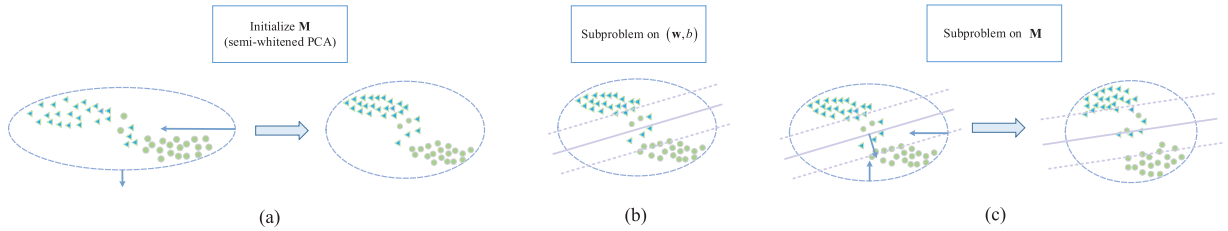


Fig. 1. Intuitive explanation on the goals of the three main steps of our algorithm. (a) By assuming $\|\mathbf{w}\|_2 = 1$, a semiwhitened PCA method is adopted for the initialization of $\mathbf{M}$, which is more reasonable than whitened PCA by considering both the radius and the margin information. (b) Subproblem on $(\mathbf{w}, b)$ can then be solved by the off-the-shelf SVM solvers to maximize the margin. (c) $\mathbf{M}$ is updated by balancing the following two terms: 1) shrinking $\mathbf{M}$ based on the weighted covariance matrix $\mathbf{S}$ and 2) expanding $\mathbf{M}$ along the direction of $\mathbf{w}$. As a result, the updated $\mathbf{M}$ not only can decrease the radius of MEB, but also may even increase the margin.

satisfies all inequality constraints and $L(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*, \mathbf{M}^*) < L(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}})$. Then we can define $\tilde{R} = 1$ and have $F(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*, \mathbf{M}^*, \tilde{R}) < F(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}}, \tilde{R})$, which is contradictory with the assumption that $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}}, \tilde{R})$ is the optimal solution to (9). Thus, we can solve the problem in (10) with the optimal solution $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}})$, and then obtain the optimal solution $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}}, \tilde{\mathbf{M}}, \tilde{R})$ to (9). ■

Without loss of generality, we assume $\tilde{R} = 1$ and seek the corresponding optimal $\mathbf{w}$ and $\mathbf{M}$ by solving (10). Moreover, to make the model robust against outliers and noisy samples, we combine the individual inequality constraints $(\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M}(\mathbf{x}_i - \bar{\mathbf{x}}) \leq 1$, $i = 1, 2, \ldots, n$ into one integrated inequality constraint [13]. By emphasizing more on the samples far from the mean $\bar{\mathbf{x}}$, the integrated inequality constraint is defined as $\sum_{i=1}^n \omega_i (\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M}(\mathbf{x}_i - \bar{\mathbf{x}}) \leq \varepsilon$ with $\omega_i = ((\exp(\|\mathbf{x}_i - \bar{\mathbf{x}}\|_2^2))/(\sum_{j=1}^n \exp(\|\mathbf{x}_j - \bar{\mathbf{x}}\|_2^2)))$, resulting in the following radius-margin-based SVM model:

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi},\mathbf{M}} \ & \frac{1}{2}(\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}) + C\sum_{i=1}^n \xi_i \\ \text{s.t. } & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, n \\ & \sum_{i=1}^n \omega_i (\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M}(\mathbf{x}_i - \bar{\mathbf{x}}) \leq \varepsilon \\ & \mathbf{M} \succ 0. \end{aligned} \tag{11}$$

The model above is a constrained optimization problem. The constraints $\mathbf{M} \succ 0$, $\xi_i \geq 0$ and $y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i$ $(i = 1, 2, \ldots, n)$ define a convex set. Let the weighted covariance matrix $\mathbf{S} = \sum_{i=1}^n \omega_i (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top$. The constraint $\sum_{i=1}^n \omega_i (\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M}(\mathbf{x}_i - \bar{\mathbf{x}}) \leq \varepsilon$ can be equivalently written as $\text{tr}(\mathbf{MS}) \leq \varepsilon$ and also defines a convex set. The objective function of (11) consists of two terms, i.e., $\sum_{i=1}^n \xi_i$ and $\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}$. It is obvious that $\sum_{i=1}^n \xi_i$ is linear to $\boldsymbol{\xi}$. According to **Appendix C**, $\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}$ is convex to $\mathbf{w}$ and $\mathbf{M} \succ 0$. To sum up, all the constraints define a convex set, and objective function is convex. Thus, the model in (11) is convex and can be equivalently formulated as

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi},\mathbf{M}} \ & \frac{1}{2}(\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}) + C\sum_{i=1}^n \xi_i + \rho \text{tr}(\mathbf{MS}) \\ \text{s.t. } & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, n \\ & \mathbf{M} \succ 0 \end{aligned} \tag{12}$$

where ρ is the regularization parameter determined by ε. Based on the method of Lagrange multipliers [15], for every $\varepsilon \geq 0$, there is a ρ such that the Karush–Kuhn–Tucker conditions are satisfied and the problems in (11) and (12) have the same solutions. Similarly, for every $\rho \geq 0$, there is also a ε. In (11), the parameter ε should be manually set or determined using cross validation. In this paper, instead of setting ε and finding the optimal ρ' (refer to **Lemma B.1**), we empirically set ρ for (12) that corresponds to the best average classification accuracy in our experiments. Please refer to **Appendix B** for the analysis of the equivalent of the two formulations. In the following, we show that our F-SVM model is convex.

Theorem 3: The F-SVM model in (12) is a convex optimization problem.

The proof can be found in **Appendix C**.

B. Optimization Algorithm

In this section, we propose an efficient GBCD algorithm to solve the F-SVM model. Fig. 1 intuitively explains the goals of its main steps (i.e., the initialization of $\mathbf{M}$, the subproblem on $(\mathbf{w}, b)$, and the subproblem on $\mathbf{M}$). Fig. 2 illustrates the subproblem and solution involved in each step. In the following, we explain each step in detail.

1) Initialization of $\mathbf{M}$*:* Proper initialization is helpful in improving computational efficiency. To this end, by further relaxing the F-SVM model in (12), we propose a semiwhitened PCA-based initialization method of $\mathbf{M}$.

Note that $\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}$ is upper bounded by [16]

$$\begin{aligned} \mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w} &= \text{tr}(\mathbf{w}\mathbf{w}^\top \mathbf{M}^{-1}) \\ &\leq \|\mathbf{w}\|_2^2 \|\mathbf{M}^{-1}\|_2 \\ &\leq \|\mathbf{w}\|_2^2 \|\mathbf{M}^{-1}\|_* \end{aligned} \tag{13}$$

where $\|\cdot\|_2$ and $\|\cdot\|_*$ denote the ℓ_2-norm and the nuclear norm of a matrix, respectively. The nuclear norm of a matrix, also known as the trace norm, is defined as the summation of all its singular values [17]. Based on (12) and (13), by setting $\mathbf{B} = \mathbf{M}^{-1}$, the subproblem of $\mathbf{M}$ can be rewritten as the problem of $\mathbf{B}$

$$\begin{aligned} \min_{\mathbf{B}} \ & \mathcal{L}(\mathbf{B}) = \|\mathbf{B}\|_* + \tau' \text{tr}(\mathbf{B}^{-1}\mathbf{S}) \\ \text{s.t. } & \mathbf{B} \succ 0 \end{aligned} \tag{14}$$

where $\tau' = \rho/\|\mathbf{w}\|^2$. The eigenvalue decomposition of $\mathbf{S}$ is $\mathbf{S} = \mathbf{U}\Lambda\mathbf{U}^\top$, where $\Lambda = \text{Diag}\{\lambda_1, \lambda_2, \ldots, \lambda_d\}$ $(\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0)$, and λ_i and the ith column of $\mathbf{U}$ denote the

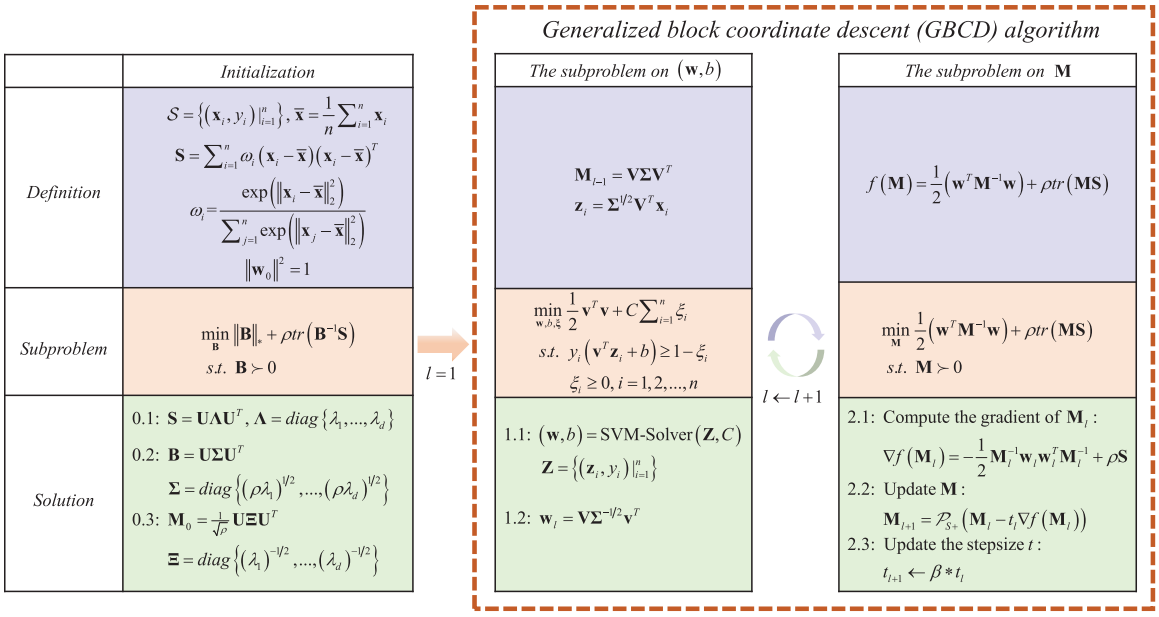


Fig. 2. Illustration of the subproblems with their solutions in our optimization algorithms. In the initialization stage, we initialize $\mathbf{M}$ by solving a nuclear norm optimization problem. In the GBCD algorithm, we solve the subproblem on $(\mathbf{w}, b)$ by using the off-the-shelf SVM solver, and solve the subproblem on $\mathbf{M}$ by using the projected gradient descent (PGD) algorithm. Our GBCD algorithm alternates between updating $(\mathbf{w}, b)$ and $\mathbf{M}$ until convergence.

ith eigenvalue and eigenvector, respectively. With $\mathbf{U}$ and $\mathbf{\Lambda}$, we define $\hat{\mathbf{B}}$ as

$$\hat{\mathbf{B}} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^\top, \quad \mathbf{\Sigma} = \mathrm{Diag}\{(\tau'\lambda_1)^{1/2}, \ldots, (\tau'\lambda_d)^{1/2}\}. \tag{15}$$

Theorem 4: Given a symmetric positively defined (SPD) matrix $\mathbf{S}$ and $\tau' > 0$, $\hat{\mathbf{B}}$ defined in (15) is the optimal solution to the problem

$$\hat{\mathbf{B}} = \arg\min_{\mathbf{B}} \{L(\mathbf{B}, \tau') = \|\mathbf{B}\|_* + \tau'(\mathrm{tr}(\mathbf{B}^{-1}\mathbf{S}))\}. \tag{16}$$

The proof can be found in **Appendix D**. With $\hat{\mathbf{B}}$, the initialization of $\mathbf{M}$ in (12) is then defined as

$$\mathbf{M}_0 = \frac{1}{\sqrt{\tau'}}\mathbf{U}\mathbf{\Xi}\mathbf{U}^\top, \quad \mathbf{\Xi} = \mathrm{Diag}\{(\lambda_1)^{-1/2}, \ldots, (\lambda_d)^{-1/2}\}. \tag{17}$$

Note that we assume $\|\mathbf{w}\|^2$ is known for the initialization of $\mathbf{M}$. From (17), $\|\mathbf{w}\|^2$ only affects the scale factor $\sqrt{\tau'}$ to the linear transformation. Thus, we simply let $\|\mathbf{w}\|^2 = 1$ in our implementation.

It is interesting to point out that $\mathbf{M}_0$ in (17) is closely related with PCA and whitened PCA, and can be regarded as a semiwhitened PCA. $\mathbf{S} = \mathbf{U}\Lambda\mathbf{U}^\top$ is denoted as the eigenvalue decomposition of the covariance matrix $\mathbf{S}$. PCA, whitened PCA, and our semiwhitened PCA are described as follows.

1) In standard PCA, the linear transformation matrix is defined as $\mathbf{A} = \mathbf{U}^\top = \mathbf{\Lambda}^0\mathbf{U}^\top$. The Euclidean distance in the transformation space can be written as $\|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_j\|^2 = (\mathbf{x}_i - \mathbf{x}_j)^\top\mathbf{U}\mathbf{\Lambda}^0\mathbf{U}^\top(\mathbf{x}_i - \mathbf{x}_j)$.
2) In the whitened PCA, the whitening transformation matrix is defined as $\mathbf{A} = \mathbf{\Lambda}^{-(1/2)}\mathbf{U}^\top$ [18]. The Euclidean distance in the transformation space can be written as $\|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_j\|^2 = (\mathbf{x}_i - \mathbf{x}_j)^\top\mathbf{U}\mathbf{\Lambda}^{-1}\mathbf{U}^\top(\mathbf{x}_i - \mathbf{x}_j)$. Whitening is a useful preprocessing strategy and has been widely exploited in many applications, e.g., face recognition [19] and object detection [20].
3) Based on (17), we define $\mathbf{A} = \mathbf{\Lambda}^{-(1/4)}\mathbf{U}^\top$, and the Euclidean distance in the transformation space can then be written as $\|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_j\|^2 = (\mathbf{x}_i - \mathbf{x}_j)^\top\mathbf{M}_0(\mathbf{x}_i - \mathbf{x}_j)$. Note that $\mathbf{A}$ is defined as $\mathbf{\Lambda}^0\mathbf{U}^\top$ for standard PCA and as $\mathbf{\Lambda}^{-(1/2)}\mathbf{U}^\top$ for whitened PCA, we call $\mathbf{A} = \mathbf{\Lambda}^{-(1/4)}\mathbf{U}^\top$ together with $\mathbf{M}_0 = \mathbf{U}\mathbf{\Lambda}^{-(1/2)}\mathbf{U}^\top$ the semiwhitened PCA. Compared with whitened PCA, semiwhitened PCA seems to be a more reasonable choice by considering both the radius and margin information.

In addition, the proposed initialization method is also connected with eigenvalue power normalization (EPN), where $\tilde{\mathbf{S}} = \mathbf{U}\mathbf{\Lambda}^p\mathbf{U}^\top (0 \le p \le 1)$ is adopted for the normalization of $\mathbf{S}$. EPN has been used to measure the distances between SPD matrices [21], [22], and has achieved promising performance in image classification [22]. Considering its connections with PCA, whitened PCA and EPN, it is natural to expect that our semiwhitened PCA can find more applications in various learning tasks.

2) Subproblem of $(\mathbf{w}, b)$: Given $\mathbf{M}$, the F-SVM model can be formulated as

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi}} \ & \frac{1}{2}\mathbf{w}^\top\mathbf{B}\mathbf{w} + C\sum_{i=1}^{n}\xi_i \\ \text{s.t. } & y_i(\mathbf{w}^\top\mathbf{x}_i + b) \ge 1 - \xi_i \quad \forall i \\ & \xi_i \ge 0, \quad i = 1, 2, \ldots, n \end{aligned} \tag{18}$$

where $\mathbf{B} = \mathbf{M}^{-1}$. The eigenvalue decomposition of $\mathbf{M}$ is $\mathbf{M} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^\top$. By introducing $\mathbf{M} = \mathbf{L}^\top\mathbf{L}$, the transformation matrix $\mathbf{L}$ can be rewritten as $\mathbf{L} = \mathbf{\Sigma}^{(1/2)}\mathbf{V}^\top$. Let $\mathbf{z}_i = \mathbf{L}\mathbf{x}_i$ and $\mathbf{v} = \mathbf{\Sigma}^{-(1/2)}\mathbf{V}^\top\mathbf{w}$. With simple algebra, the problem in (18)

can be reformulated as

$$\begin{aligned}\min_{\mathbf{v},b,\xi}\ & \frac{1}{2}\mathbf{v}^\top\mathbf{v}+C\sum_{i=1}^{n}\xi_i\\ \text{s.t. }& y_i(\mathbf{v}^\top\mathbf{z}_i+b)\geq 1-\xi_i\quad \forall i\\ & \xi_i\geq 0,\quad i=1,2,\ldots,n\end{aligned}\tag{19}$$

which can be solved using off-the-shelf SVM solvers. Given the solution $\mathbf{v}$, $\mathbf{w}=\mathbf{V}\boldsymbol{\Sigma}^{(1/2)}\mathbf{v}$ can then be obtained.

3) Subproblem of $\mathbf{M}$*:* Given $(\mathbf{w},b)$, the subproblem of $\mathbf{M}$ can be reformulated as

$$\begin{aligned}\min_{\mathbf{M}}\ & f(\mathbf{M})=\frac{1}{2}(\mathbf{w}^\top\mathbf{M}^{-1}\mathbf{w})+\rho\text{tr}(\mathbf{MS})\\ \text{s.t. }& \mathbf{M}\succ 0.\end{aligned}\tag{20}$$

Since the objective function in (20) is convex and differentiable with respect to $\mathbf{M}\succ 0$, the gradient-projection method is adopted to update $\mathbf{M}$. The gradient of $f(\mathbf{M})$ can be obtained by

$$\nabla f(\mathbf{M})=-\frac{1}{2}\mathbf{M}^{-1}\mathbf{w}\mathbf{w}^\top\mathbf{M}^{-1}+\rho\mathbf{S}.\tag{21}$$

As presented in **Algorithm 1**, we use gradient projection

$$\mathbf{M}=\mathcal{P}_{\mathcal{S}_+}(\mathbf{M}-t\nabla f(\mathbf{M}))\tag{22}$$

to update $\mathbf{M}$ by choosing the proper stepsize t and gradually decreasing it along with iterations, where $\mathcal{P}_{\mathcal{S}_+}(\cdot)$ projects a matrix onto the cone of positive semidefinite matrices. Since $\mathbf{M}$ should be strictly positively defined to guarantee that its inverse matrix $\mathbf{M}^{-1}$ exists, we add $\gamma\mathbf{I}$ to matrix $\mathbf{M}$, where γ is a small positive value and is set as $\gamma=10^{-4}$ in our implementation. We note that the computational complexity of positive semi-definite (PSD) projection is $O(d^3)$. Fortunately, the dimension d usually is small. Moreover, due to the warm initialization method, our GBCD algorithm generally can obtain satisfying result within only a few (e.g., 10) iterations.

Intuitively, $\mathbf{M}$ is updated by balancing the two terms: 1) shrinking along the directions of the eigenvectors of the weighted covariance matrix $\mathbf{S}$ by its eigenvalues and 2) expanding along the direction of $\mathbf{w}$. Thus, as illustrated in Fig. 2(c), the above algorithm not only can decrease the radius of MEB, but also may even increase the margin to minimize the approximated radius-margin error bound. Finally, the whole GBCD algorithm is summarized in **Algorithm 1**.

4) Convergence Analysis: The proposed **Algorithm** 1 is a GBCD method. Xu and Yin [23] provide a unified framework to analyze the convergence of a regularized block multiconvex optimization problem

$$\min_{\boldsymbol{\Theta}}\left\{F(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\ldots,\boldsymbol{\theta}_s)=f(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\ldots,\boldsymbol{\theta}_s)+\sum_{i=1}^{s}r_i(\boldsymbol{\theta}_i)\right\}$$

where $\boldsymbol{\Theta}=(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\ldots,\boldsymbol{\theta}_s)$. The GBCD method minimizes F by cyclically updating each $\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\ldots,\boldsymbol{\theta}_s$ while fixing the remaining blocks. In GBCD, each block can be updated by solving either the original subproblem, proximal subproblem, or prox-linear subproblem. Xu and Yin [23] analyze the convergence of the algorithm in two steps. First, under the assumptions of continuity and block convexity, GBCD can globally converge to a single Nash point if the bounded sequence and the isolated Nash points hold. Second, the global convergence can be further established by applying the Kurdyka−Lojasiewicz inequality [24].

Algorithm 1 The GBCD algorithm for training F-SVM

Input: Training set $\mathcal{S}=\{(\mathbf{x}_1,y_1),...,(\mathbf{x}_n,y_n)\}$.
Output: Optimal positive definite matrix $\mathbf{M}^*$ and $(\mathbf{w}^*,b^*)$.
1: $l=1$, $t_1=1$, $\beta=0.9$
2: Initialize $\mathbf{M}_1$:
3: $\quad\mathbf{S}=\sum_{i=1}^{n}\omega_i(\mathbf{x}_i-\bar{\mathbf{x}})(\mathbf{x}_i-\bar{\mathbf{x}})^\top$
4: $\quad\omega_i=\frac{\exp(\|\mathbf{x}_i-\bar{\mathbf{x}}\|_2^2)}{\sum_{j=1}^{n}\exp(\|\mathbf{x}_j-\bar{\mathbf{x}}\|_2^2)}$
5: $\quad\mathbf{S}=\mathbf{U}\Lambda\mathbf{U}^\top$, $\Lambda=\text{Diag}\{\lambda_1,\lambda_2,\cdots,\lambda_d\}$
6: $\quad\mathbf{M}_1=\sqrt{\tau'}\mathbf{U}\Xi\mathbf{U}^\top$, $\Xi=\text{Diag}\{\lambda_1^{-\frac{1}{2}},\lambda_2^{-\frac{1}{2}},\cdots,\lambda_d^{-\frac{1}{2}}\}$
7: **repeat**
8: $\quad$// Lines 9-11: updating $(\mathbf{w},b)$.
9: $\quad$Eigenvalue decomposition on $\mathbf{M}_l$: $\mathbf{M}_l=\mathbf{V}\Sigma\mathbf{V}^\top$
10: $\quad$Perform linear transformation on $\mathbf{x}_i$: $\mathbf{z}_i\leftarrow\Sigma^{1/2}\mathbf{V}^\top\mathbf{x}_i$
11: $\quad$Update the SVM classifier $(\mathbf{w}_l,b_l)$ based on $\mathbf{Z}$
12: $\quad$// Lines 13-15: updating $\mathbf{M}$.
13: $\quad$Compute the gradient of $\mathbf{M}_l$:
14: $\quad\quad\nabla f(\mathbf{M}_l)=-\frac{1}{2}\mathbf{M}_l^{-1}\mathbf{w}_l\mathbf{w}_l^\top\mathbf{M}_l^{-1}+\rho\mathbf{S}$
15: $\quad$Update $\mathbf{M}$: $\mathbf{M}_{l+1}=\mathcal{P}_{\mathcal{S}_+}(\mathbf{M}_l-t_l\nabla f(\mathbf{M}_l))$
16: $\quad$Update the stepsize t: $t_{l+1}\leftarrow\beta*t_l$
17: $\quad l\leftarrow l+1$
18: **until** $\mathbf{M}$ and $(\mathbf{w},b)$ converge

Our method in **Algorithm** 1 is a GBCD method and can converge to a global optimal solution. Let $\boldsymbol{\theta}_1=(\mathbf{w},b,\boldsymbol{\xi})$ and $\boldsymbol{\theta}_2=\mathbf{M}$. We define

$$\begin{aligned}f(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)&=\frac{1}{2}(\mathbf{w}^\top\mathbf{M}^{-1}\mathbf{w})+C\sum_{i=1}^{n}\xi_i\\ \text{s.t. }& y_i(\mathbf{w}^\top\mathbf{x}_i+b)\geq 1-\xi_i\\ & \xi_i\geq 0,\quad i=1,2,\ldots,n\\ & r_1(\boldsymbol{\theta}_1)=0\\ r_2(\boldsymbol{\theta}_2)&=\begin{cases}0, & \text{if }\ \mathbf{M}\succ 0\\ +\infty, & \text{else.}\end{cases}\end{aligned}$$

Then our F-SVM can be rewritten as the the regularized block multiconvex optimization problem

$$\min_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2}\left\{F(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)=f(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)+\sum_{i=1}^{2}r_i(\boldsymbol{\theta}_i)\right\}.$$

Given $\mathbf{M}=\mathbf{B}^{-1}$, we update $\boldsymbol{\theta}_1$ by solving the subproblem

$$\begin{aligned}\{\mathbf{w},b,\boldsymbol{\xi}\}=\arg\min_{\mathbf{w},b,\boldsymbol{\xi}}\ & \frac{1}{2}(\mathbf{w}^\top\mathbf{B}\mathbf{w})+C\sum_{i=1}^{n}\xi_i\\ \text{s.t. }& y_i(\mathbf{w}^\top\mathbf{x}_i+b)\geq 1-\xi_i\\ & \xi_i\geq 0,\quad i=1,2,\ldots,n.\end{aligned}$$

Given $\{\mathbf{w},b,\boldsymbol{\xi}\}$, we update $\boldsymbol{\theta}_2$ by solving the prox-linear subproblem

$$\begin{aligned}\mathbf{M}_l=\arg\min_{\mathbf{M}}\ & \langle\nabla f(\mathbf{M}_{l-1}),\mathbf{M}-\mathbf{M}_{l-1}\rangle\\ & +\frac{1}{2t}\|\mathbf{M}-\mathbf{M}_{l-1}\|^2+r_2(\mathbf{M})\end{aligned}$$

and the closed form solution of this subproblem is exactly the updating rule adopted in (22). Note that $f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is locally strongly convex and thus satisfies the Kurdyka−Lojasiewicz inequality [24]. The proposed algorithm can converge to a globally optimum. Our experiments also empirically validate the convergence of our algorithm.

C. Discussion

Compared with the other radius-margin-based SVMs [2], [5]–[7], the proposed F-SVM method has several important advantages. RMM [5] is suggested to maximize the margin while restricting the spread of the data along the direction perpendicular to the separating hyperplane. Actually, the generalization error is bounded by the radius and margin ratio, and the radius is determined by the spread along all possible directions rather than only the direction perpendicular to the separating hyperplane, making F-SVM theoretically more promising. Different with RMM, our F-SVM is proposed to minimize the convex relaxation of the radius-margin ratio, and is expected to achieve a better classification performance.

MR-SVM [2], R-SVM$^+$, and R-SVM$^+_{\boldsymbol{\mu}}$ [7] aim to learn the diagonal feature transformation $\mathbf{D}_{\boldsymbol{\mu}} = \mathrm{Diag}(\boldsymbol{\mu})$ with $\mu_k \geq 0$, while F-SVM is developed for joint learning of the feature transformation and SVM classifier. Both R-SVM$^+$ and R-SVM$^+_{\boldsymbol{\mu}}$ need to solve a quadratically constrained quadratic programming optimization problem, which is computationally more expensive than the alternating minimization method used in our F-SVM. Moreover, R-SVM$^+$ and R-SVM$^+_{\boldsymbol{\mu}}$ adopt another approximation R_O of the radius. In F-SVM, a new approximation $\bar{R}$ of the radius is proposed, which is tighter than that used in MR-SVM [2]. Moreover, the individual inequality constraints on $\bar{R}$ are combined to improve the robustness against outliers.

Besides, MSVM [6] has also been suggested for joint learning of the linear transformation and SVM classifier. However, the source code of MSVM [6] is unavailable, and our F-SVM is distinctly different with MSVM. First, MSVM is built upon an evidently wrong claim on the radius of MEB. Instead, we derive novel lower and upper bounds for the radius of MEB, which lay a solid theoretical foundation to F-SVM.

Second, in MSVM the inequality constraints on distance are treated individually, i.e., $\|\mathbf{A}\mathbf{x}_i - \mathbf{A}\bar{\mathbf{x}}\|^2 \leq 1, \forall i$. In F-SVM, all the constraints are aggregated into one united inequality, i.e., $\sum_{i=1}^n \omega_i (\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M} (\mathbf{x}_i - \bar{\mathbf{x}}) \leq \varepsilon$. The united inequality constraint can be used to improve the robustness and to reduce computational budget of F-SVM. Moreover, the introduction of united inequality constraint also makes it feasible to initialize $\mathbf{M}$ with semiwhitened PCA, which can greatly improve the training efficiency. To the best of our knowledge, our F-SVM is the first work to introduce the united inequality constraint in radius-margin-based SVMs.

Third, compared with MSVM which aims to learn $\mathbf{A}$ and $(\mathbf{w}, b)$, our F-SVM jointly learns $\mathbf{M} = \mathbf{A}^\top \mathbf{A}$ and $(\mathbf{w}, b)$. Benefitted from the introduction of $\mathbf{M}$, our F-SVM can be formulated into a convex optimization model and can attain globally optimal solution while MSVM is nonconvex. To the best of our knowledge, F-SVM is the first convex model for joint learning of feature transformation and SVM classifier.

Fourth, benefitted from the united inequality constraint and the convex formulation of F-SVM, we develop a semiwhitened PCA method for initialization and a GBCD algorithm to solve our F-SVM model. Our algorithm is guaranteed to converge to the global optimum, while the gradient-projection algorithm used in MSVM can only reach the locally optimal solution. Moreover, our F-SVM can be extended to its kernelized version for nonlinear classification.

IV. Kernelization of F-SVM

With the incorporation of kernel PCA, linear F-SVM can be extended to a kernelized version for nonlinear classification. First, we show that kernel SVM is equivalent to performing linear SVM in kernel PCA space. Then, kernel F-SVM is introduced by conducting linear F-SVM in the kernel PCA space.

Suppose the kernel function is $K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)^\top \Phi(\mathbf{x}_j)$, where $\Phi(\mathbf{x})$ defines an implicit mapping of feature space. For the training set $\mathcal{S}$, we use $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{D_0}]$ to denote all the PCA eigenvectors corresponding to positive eigenvalues. Let $\overline{\mathbf{W}}$ be a set of basis vectors in the complementary space of $\mathbf{W}$. Assuming the training set is centered, we have $\overline{\mathbf{W}}^\top \Phi(\mathbf{x}_i) = 0$, and can get

$$\begin{aligned} K(\mathbf{x}_i, \mathbf{x}_j) &= \Phi(\mathbf{x}_i)^\top \mathbf{W}\mathbf{W}^\top \Phi(\mathbf{x}_j) + \Phi(\mathbf{x}_i)^\top \overline{\mathbf{W}}\, \overline{\mathbf{W}}^\top \Phi(\mathbf{x}_j) \\ &= \Phi(\mathbf{x}_i)^\top \mathbf{W}\mathbf{W}^\top \Phi(\mathbf{x}_j). \end{aligned} \tag{23}$$

Let $\mathbf{f}_i = \mathbf{W}^\top \Phi(\mathbf{x}_i)$. The dual problem of SVM in the kernel PCA space can be formulated as

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \;& Q(\boldsymbol{\alpha}) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle \mathbf{f}_i, \mathbf{f}_j \rangle \\ \text{s.t.} \;& \sum_i \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq C \quad \forall i \end{aligned} \tag{24}$$

where $\langle \mathbf{f}_i, \mathbf{f}_j \rangle = K(\mathbf{x}_i, \mathbf{x}_j)$. Therefore, kernel SVM is equivalent to performing linear SVM in the kernel PCA space. To extend F-SVM to its kernelized version, we first project each training sample $\mathbf{x}_i$ to the kernel PCA space $\mathbf{f}_i = \mathbf{W}^\top \Phi(\mathbf{x}_i)$, and then solve the following F-SVM model:

$$\begin{aligned} \min_{\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{M}} \;& \frac{1}{2} (\mathbf{w}^\top \mathbf{M}^{-1} \mathbf{w}) + C \sum_{i=1}^n \xi_i + \rho \mathrm{tr}(\mathbf{M}\mathbf{S}_f) \\ \text{s.t.} \;& y_i (\mathbf{w}^\top \mathbf{f}_i + b) \geq 1 - \xi_i \quad \forall i \\ & \xi_i \geq 0 \quad \forall i \\ & \mathbf{M} \succ 0 \end{aligned} \tag{25}$$

where $\mathbf{S}_f = \sum_{i=1}^n \omega_i \mathbf{f}_i \mathbf{f}_i^\top$. **Algorithm** 1 can be adopted to solve the model in (25). In our implementation, instead of using all the eigenvectors, we only employ the PCA eigenvectors corresponding to the first D largest eigenvalues, i.e., $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_D]$.

V. Experiments

Experiments are conducted on 20 UCI data sets as described in Table I, the labeled faces in the wild (LFW), and

TABLE I
DESCRIPTION OF THE 20 UCI DATA SETS[1] USED IN THE EXPERIMENTS

Dataset	Dataset ID	# of samples	# of classes	Feature dimensions	Dataset	Dataset ID	# of samples	# of classes	Feature dimensions
Balance	1	625	3	4	Pima	11	768	2	8
Breast	2	277	2	9	Segment	12	2310	7	18
German	3	1000	2	24	Sonar	13	208	2	60
Glass	4	214	6	9	Vote	14	435	2	16
Heart	5	303	2	13	Waveform3	15	5000	3	21
Ionosphere	6	351	2	34	Wine	16	178	3	13
Iris	7	150	3	4	Wpbc	17	198	2	33
Liver	8	345	2	6	Zoo	18	101	7	16
Musk	9	6598	2	166	Ringnorm	19	7400	2	20
Parkinsons	10	195	2	22	Twonorm	20	7400	2	20

four large-scale image classification data sets, i.e., MNIST, CIFAR-10 [25], CIFAR-100 [25], and Caltech101 [26]. On the UCI and LFW data sets, the F-SVM method is compared with several related methods, including the standard SVM, RMM [5], R-SVM$^+$, and R-SVM$^+_{\boldsymbol{\mu}}$ [7]. On LFW we also compare F-SVM with several recent face verification methods, including discriminative deep metric learning (DDML) [27], HPEN+HD-Gabor+JB [28], and multi-directional multi-level dual-cross patterns (MDML-DCPs) [29]. On MNIST, CIFAR-10, CIFAR-100 and Caltech101, motivated by the recent success of CNN, we stack the F-SVM model upon the deep CNNs, and compare with the state-of-the-art methods with similar network architecture.

On the UCI and LFW data sets, we adopt the average classification accuracy (%) obtained by 10 runs of the tenfold cross validation (CV) as the performance metric. To avoid the use of the test set for parameter tuning, we employ a modified tenfold CV which involves both outer tenfold CV and inner fivefold CVs. In our tenfold CV, the training set of n samples is randomly partitioned into ten folds of size $n/10$. By retaining one fold for testing, we further invoke an inner fivefold CV on the remained nine folds to determine the optimal hyper parameters. Then, we train the classifier on the nine folds with the optimal hyper parameters, and evaluate the learned classifier using the retained test fold. Finally, the results on the ten test folds are averaged to produce a single estimation. Moreover, the running time of each method is provided based on the ten runs of our tenfold CV. Following the standard protocol on the LFW, beside the mean classification accuracy with the standard deviation, we also provide the receiver operating characteristic (ROC) curve as [27]–[29]. All the experiments are carried out on a desktop PC with Intel(R) Xeon(R) CPU (3.30 GHz) and 32GB RAM under the MATLAB 2015b programming environment. The CNN-based experiments are implemented using the MatConvNet package on the Tesla K80 GPU.

In experiments, a coarse-to-fine search strategy is adopted for determining the hyper parameters within the inner fivefold CVs. The grid search method is first adopted for coarse searching, and then the line bisection method is exploited to refine the hyper parameters within a small range. Concretely, we set $C \in \{2^{c_{\min}:c_{\text{step}}:c_{\max}}\}$ with $c_{\min} = -10$, $c_{\text{step}} = 1$, $c_{\max} = 20$, and $\sigma \in \{2^{\sigma_{\min}:\sigma_{\text{step}}:\sigma_{\max}}\}$ with $\sigma_{\min} = -20$, $\sigma_{\text{step}} = 1$, $\sigma_{\max} = 5$ for Gaussian RBF kernel. We also tune the optimal parameter ρ for each data set from $\rho \in \{2^{\rho_{\min}:\rho_{\text{step}}:\rho_{\max}}\}$ with $\rho_{\min} = -2$, $\rho_{\text{step}} = 0.5$, $\rho_{\max} = 6$. When updating $\mathbf{M}$ through the gradient-projection method, the stepsize t is initialized as 1, and decreases gradually controlled by $\beta = 0.9$ as shown in **Algorithm 1**. The source codes of F-SVM and Kernel F-SVM are online available.[2]

TABLE II
COMPARISON OF THE AVERAGE CLASSIFICATION ACCURACY (%) BY LINEAR SVM, LINEAR RMM [5], LINEAR R-SVM$^+$ [7], LINEAR R-SVM$^+_{\boldsymbol{\mu}}$ [7], AND LINEAR F-SVM

Dataset	SVM	RMM	R-SVM$^+_{\boldsymbol{\mu}}$	R-SVM$^+$	F-SVM
Balance	91.11	89.41	90.00	90.21	**91.42**
Breast	70.71	72.71	73.46	73.05	**75.00**
German	77.50	77.02	77.10	77.22	**78.40**
Glass	66.36	67.51	68.22	68.35	**70.45**
Heart	87.42	87.14	87.06	87.37	**88.39**
Ionosphere	89.17	90.64	90.96	90.83	**91.67**
Iris	96.67	96.67	97.33	97.33	**98.00**
Liver	70.00	71.70	71.71	71.01	**74.79**
Musk	91.98	92.20	92.20	92.00	**92.52**
Parkinsons	89.06	90.01	90.00	**91.10**	90.22
Pima	77.53	78.51	78.00	78.67	**79.48**
Segment	90.61	91.36	91.67	91.77	**92.60**
Sonar	81.43	82.82	83.33	83.67	**85.71**
Vote	95.00	95.20	95.01	95.22	**95.68**
Waveform3	87.40	88.50	87.05	**88.51**	87.80
Wine	96.67	97.39	97.50	97.77	**98.89**
Wpbc	79.50	82.50	81.97	82.10	**84.00**
Zoo	94.55	95.67	96.79	96.67	**98.18**
Ringnorm	75.58	75.60	75.67	75.77	**76.65**
Twonorm	96.65	96.60	96.67	96.00	**97.82**

A. Results on the UCI Data Sets

1) Evaluation on Linear F-SVM: Table II and Fig. 3 present the average classification accuracy and run time of our linear F-SVM and the competing methods. As shown in Table II, RMM [5], R-SVM$^+$, and R-SVM$^+_{\boldsymbol{\mu}}$ [7] generally outperform the standard SVM with an average accuracy of 0.8% above, which indicates that the incorporation of the radius is effective in improving classification performance. In addition, F-SVM achieves the best or the second best classification accuracy on most data sets, and obtains an average improvement of 2.14% over SVM. Specifically, the improvement of F-SVM over SVM is higher than 4.0% by accuracy on 6 data sets,

[1] http://archive.ics.uci.edu/ml/index.php

[2] https://github.com/tourmaline612/FSVM

TABLE III

COMPARISON OF THE AVERAGE CLASSIFICATION ACCURACY (%) OF KERNEL SVM, KERNEL RMM [5], KERNEL R-SVM$^+$ [7], KERNEL R-SVM$^+_\mu$ [7], AND KERNEL F-SVM

Dataset	SVM	RMM	R-SVM$^+_\mu$	R-SVM$^+$	F-SVM
Balance	97.78	98.00	97.89	97.60	**99.05**
Breast	75.71	75.65	75.17	75.97	**80.71**
German	75.50	76.59	75.93	76.01	**79.50**
Glass	73.18	73.34	73.54	73.22	**74.55**
Heart	86.45	88.19	87.96	87.83	**89.35**
Ionosphere	91.94	92.80	93.52	93.22	**96.94**
Iris	96.67	96.33	96.59	96.79	**97.33**
Liver	74.00	74.20	75.00	75.00	**76.86**
Musk	99.73	98.20	99.00	98.90	**99.86**
Parkinsons	85.37	86.77	86.33	86.50	**88.42**
Pima	77.79	77.00	78.00	78.67	**79.87**
Segment	97.36	97.81	97.88	97.56	**98.23**
Sonar	86.19	88.63	89.50	89.32	**95.71**
Vote	95.23	95.73	95.63	95.70	**97.27**
Waveform3	87.56	**87.81**	87.10	87.36	87.64
Wine	96.67	97.34	97.77	97.00	**98.89**
Wpbc	78.00	83.00	82.50	82.67	**85.00**
Zoo	93.64	94.87	94.67	94.96	**97.27**
Ringnorm	98.01	98.30	97.00	97.67	**99.04**
Twonorm	97.81	97.20	97.00	97.22	**98.00**

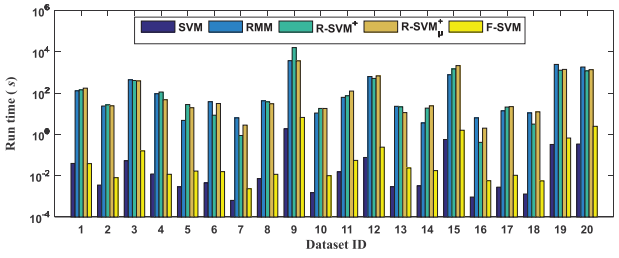


Fig. 3. Comparison of the run time (in seconds, s) of linear SVM, linear RMM [5], linear R-SVM$^+$ [7], linear R-SVM$^+_\mu$ [7], and linear F-SVM.

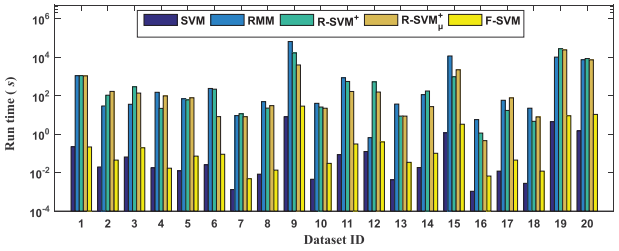


Fig. 4. Comparison of the run time (in seconds, s) of kernel SVM, kernel RMM [5], kernel R-SVM$^+$ [7], kernel R-SVM$^+_\mu$ [7], and kernel F-SVM.

i.e., *Breast*, *Glass*, *Liver*, *Sonar*, *Wpbc*, and *Zoo*. Fig. 3 shows the run time of the competing methods in training. Benefitted from our initialization scheme and the GBCD algorithm, F-SVM is about 10^3 times faster than RMM, R-SVM$^+$ and R-SVM$^+_\mu$, and is moderately slower than SVM.

2) Evaluation on Kernel F-SVM: We further compare kernel F-SVM with kernel SVM, kernel RMM [5], kernel R-SVM$^+$, and kernel R-SVM$^+_\mu$ [7] on the UCI data sets. Due to the combination of kernel PCA, we take a strategy based on eigenvalues to choose the optimal dimension. A ratio of 0.9 between the cumulative eigenvalues and the sum of all eigenvalues is set for dimension selection. From Table III, kernel F-SVM achieves the highest classification accuracy on 19 of the 20 data sets among the competing methods, and obtains an average improvement of 2.7% over kernel SVM. In contrast, the kernel RMM, kernel R-SVM$^+$ and kernel R-SVM$^+_\mu$ only have an average improvement of about 0.7%. On the data sets *Breast*, *Ionosphere*, *Sonar*, and *Wpbc*, the improvement of kernel F-SVM over kernel SVM is higher than 5.0% by accuracy. From Fig. 4, the run time of kernel F-SVM remains about 10^3 times faster than that of kernel RMM, kernel R-SVM$^+$, and kernel R-SVM$^+_\mu$. Note that the reported time does not include the time of kernel PCA. It can be observed that, on most UCI data sets, the kernel versions of F-SVM, SVM and other radius-margin-based SVM variants perform comparatively to their linear models, which is consistent with that in [5] and [7]. It can be attributed to the small scale data sets and their linear separability.

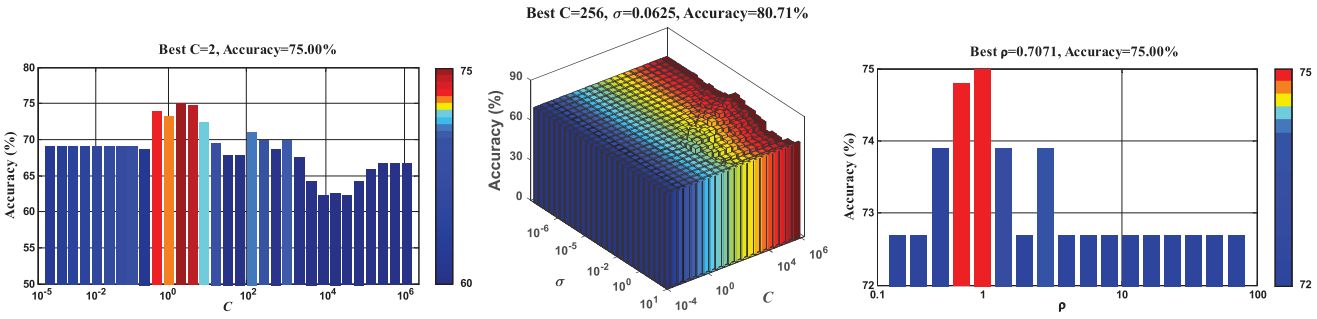
3) Parameter Effect: Using the *Breast* data set, we evaluate the effect of hyper parameters, including the tradeoff C, the kernel parameter σ in kernel F-SVM, and the parameter ρ. It can be seen from Fig. 5(a), when $C < 0.1$, the accuracy is relatively low. The classification accuracy can be significantly improved along with the increase of C to 2, and drops significantly when $C > 10^3$. From Fig. 5(b), one can see that better accuracy can be obtained by using larger C (e.g., $C = 256$) and smaller σ (e.g., $\sigma = 0.0625$). It can be seen from Fig. 5(c), the optimal range for the parameter ρ is $[1/\sqrt{2}, \sqrt{2}]$ for linear F-SVM. Similar conclusion can be obtained from other data sets.

4) Convergence Analysis: Using the *Breast* data set, we further show the convergence of the GBCD algorithm. As illustrated in Fig. 6, the objective value rapidly decreases and converges within 10 iterations.

B. Results on the LFW Face Database

The LFW database consists of more than 13 233 face images from 5749 persons. The images in LFW are collected from the Internet, and vary in pose, illumination, expression, and age, making LFW very challenging for unconstrained face verification. The face recognition method can be evaluated with two test protocols for LFW: the restricted and the unrestricted settings. Under the restricted setting, the only available information is whether each pair of training images is matched or not, and the performance is evaluated by our tenfold CV on a set of 300 positive and 300 negative image pairs. In our experiments, we adopt the restricted setting with the face images aligned by the funneling method [30].

On LFW, three groups of experiments are conducted by using the Attributes [31], scale-invariant feature transform (SIFT) features, and Oxford Visual Geometry Group Face-CNN (VGG-Face) deep features [32]. The 73-dimensional attribute features include both the describable characteristics (e.g., gender, nose size, and expression) and similes (similarity with a specific reference person) to describe face image. The SIFT features are extracted at nine fiducial points on three scales, resulting in a 3456-dimensional feature vector. The VGG-Face deep features are computed using VGG-16 CNN architecture provided by Parkhi *et al.* [32] Different from [32], we simply crop a 160×160 map from the center pixel, resize it to 224×224 for the deep network, and output a 4096-dimensional feature vector.



(a) (b) (c)

Fig. 5. Parameters analysis on the *Breast* data set. (a) C in linear F-SVM. (b) C and σ in kernel F-SVM. (c) Parameter ρ in linear F-SVM.

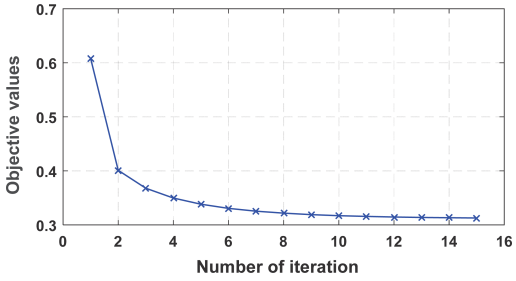


Fig. 6. Convergence curve on the *Breast* data set.

TABLE IV
COMPARISON RESULTS OF THE KERNEL SVM-BASED METHODS ON LFW BY USING THREE TYPES OF FEATURES

Methods	Attribute	SIFT	Deep Feature
SVM	80.52	74.83	95.82
RMM	80.77	74.90	96.32
R-SVM$^+_\mu$	81.02	75.32	96.20
R-SVM$^+$	81.30	75.35	96.18
F-SVM	**82.13**	**76.48**	**97.15**

Based on the comparison results of linear and kernel F-SVM on UCI data set, we evaluate kernel F-SVM on LFW and compare it with kernel SVM, kernel RMM [5], kernel R-SVM$^+$, and kernel R-SVM$^+_\mu$ [7]. Due to the limitation of computational and memory cost of RMM, R-SVM$^+$ and R-SVM$^+_\mu$, we use the PCA for dimensionality reduction with the same strategy introduced in Section V-A.

From Table IV, among the radius-margin-based SVM variants, kernel F-SVM achieves the best verification accuracies on all three types of features. In comparison to the SVM baselinethe, the improvement of F-SVM is 1.61% on Attribute and 1.65% on SIFT features. Using deep features, all the competing methods can attain an obvious performance improvement. Moreover, even SVM can achieve comparable accuracy with the radius-margin-based SVM variants RMM, R-SVM$^+_\mu$, and R-SVM$^+$, F-SVM can still get an improvement of 1.33% over SVM.

We further compare kernel F-SVM with several recent face verification methods, i.e., DDML [27], HPEN+HD-Gabor+JB [28], and MDML-DCPs [29], which are also based on deep CNNs. Fig. 7 shows the ROC curves of the competing methods and Table V lists the accuracy and AUC scores. Our kernel F-SVM with deep features achieves a verification accuracy of 97.15% and the AUC of 0.9958, which are higher than the competing methods.

TABLE V
VERIFICATION ACCURACY (%) AND AUCS BASED ON DEEP FEATURES

Methods	Accuracy (%)	AUC
SVM	95.82±0.89	0.9922
RMM [5]	96.20±0.59	0.9940
R-SVM$^+_\mu$ [7]	96.18±1.05	0.9938
R-SVM$^+$ [7]	96.32±0.99	0.9944
DDML [27]	90.68±1.41	0.9647
HPEN+HD-Gabor+JB [28]	95.25±0.36	0.9902
MDML-DCPs [29]	95.58±0.34	0.9911
F-SVM	**97.15±0.77**	**0.9958**

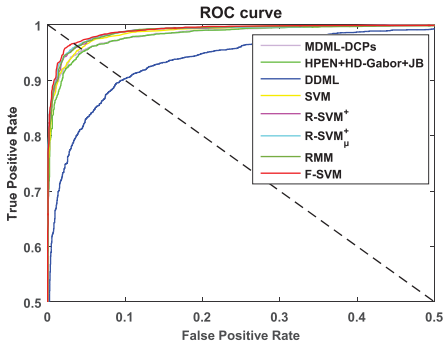


Fig. 7. ROC curves on LFW.

C. Results on Image Classification Data Sets

Finally, experiments are conducted to evaluate F-SVM on four large-scale image classification data sets, including MNIST, CIFAR-10 [25], CIFAR-100 [25], and Caltech101 [26]. For fair comparison with the state of the arts, we stack the F-SVM model upon the deep CNNs (e.g., AlexNet). Fig. 8 illustrates the architecture of AlexNet (F-SVM). To incorporate F-SVM with deep CNN, the model can be formulated as

$$\min_{\mathbf{w},b,\mathbf{M},\mathbf{W}} \frac{1}{2}(\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}) + \mathcal{L}_\mathbf{w} + \mathcal{L}_\mathbf{M} \quad \text{s.t. } \mathbf{M} \succ 0 \tag{26}$$

where $\mathbf{W}$ denotes the CNN parameters. The loss on $\mathbf{w}$ is defined as $\mathcal{L}_\mathbf{w} = C\sum_{i=1}^n \max(0, 1 - y_i(\mathbf{w}^\top \Phi(\mathbf{x}_i; \mathbf{W}) + b))$. The loss on $\mathbf{M}$ is defined as $\mathcal{L}_\mathbf{M} = \rho \sum_{i=1}^n (\Phi(\mathbf{x}_i; \mathbf{W}) - \bar{\Phi}_\mathbf{x})^\top \mathbf{M}(\Phi(\mathbf{x}_i; \mathbf{W}) - \bar{\Phi}_\mathbf{x})$, $\bar{\Phi}_\mathbf{x} = (1/n)\sum_{i=1}^n \Phi(\mathbf{x}_i; \mathbf{W})$. As illustrated in Fig. 8, we replace the softmax layer in AlexNet with the classifier $(\mathbf{w}, b)$ minimized by hinge

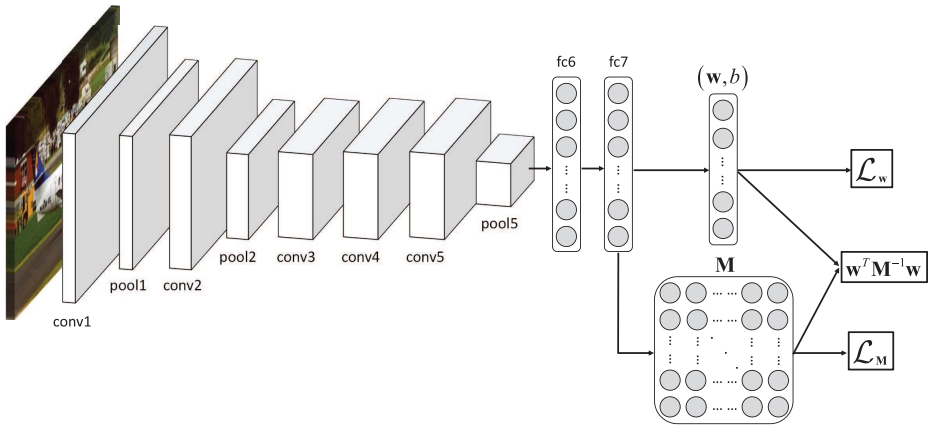


Fig. 8. Diagram of stacking the F-SVM on the deep neural network.

loss. Furthermore, a matrix $\mathbf{M}$ is introduced to constrain $\mathbf{w}$ and $\Phi(\mathbf{x}_i; \mathbf{W})$. Note that all the parameters $\mathbf{w}, b, \mathbf{M}, \mathbf{W}$ are differentiable, and can be end-to-end trained. Specifically, $\mathbf{M}$ is updated by PGD, and the PSD projection is empirically adopted after each c iterations for efficiency. In the implementation, the mean $\bar{\Phi}_{\mathbf{X}}$ is computed in a mini-batch manner. Moreover, AlexNet (F-SVM) can be extended to other CNNs, e.g., ResNet-50 [33] and ResNet-110 [33], to obtain ResNet-50 (F-SVM) and ResNet-110 (F-SVM).

Considering the input image size, we deploy different CNNs to the four data sets, i.e., ResNet-110 on MNIST (28×28), CIFAR-10 and CIFAR-100 ($32 \times 32 \times 3$) data sets, and ResNet-50 on Caltech101. As for Caltech101, since most competing methods report their results based on AlexNet [34], for fair comparison we also provide the result by AlexNet (F-SVM). In the experiments, we also implement a variant of the CNN by replacing the softmax loss with hinge loss, i.e., CNN (SVM). To better connect the ResNet and F-SVM, a fully connected layer with a 512-dimensional output is added for feature extraction. For ResNet-110, the learning rate starts from 10^{-1}, and is decreased by a factor of 10 after every 80 epochs. we use a weight decay of 10^{-4} and a batch size of 128. For ResNet-50, we use the pretrained model on ImageNet, and the images are resized to $224 \times 224 \times 3$. A weight decay of 10^{-4} and the batch size of 128 are utilized. For AlexNet, the images are resized to $227 \times 227 \times 3$. The learning rate starts from 10^{-2}, and is divided by 10 after every 40 epochs. The weight decay is set as 5×10^{-4} and the batch size is set to 64. The training is terminated after 200 epochs for all networks.

1) MNIST: Due to the small image size, we conduct experiment based on ResNet-110 and the results are presented in Table VI. Using the default softmax loss, ResNet-110 can obtain the error rate of 0.45%. In contrast, ResNet-110 with hinge loss obtains the error rate of 0.43%, which indicates that softmax loss and hinge loss result in similar performance on MNIST. Moreover, by incorporating the radius information with hinge loss, ResNet-110 (F-SVM) attains the error rate of 0.35%, which is lower than those by ResNet-110, ResNet-110 (SVM), and the state of the arts. We note that most recent methods can achieve very high accuracy on MNIST, making the improvement of ResNet-110 (F-SVM) not so notable. Thus, more experiments are conducted on three more challenging data sets, i.e., CIFAR-10, CIFAR-100, and Caltech101.

2) CIFAR-10: The CIFAR-10 data set [25] consists of 60 000 natural color images from 10 classes, with 6000 images per class. The data set consists of a training set of 50 000 images and a test set of 10 000 images. Following [35], all the input images are preprocessed by global contrast normalization and zero-phase component analysis whitening. For ResNet-110, we follow the data augmentation in [36] for training: four pixels are padded on each side, and a 32×32 crop is randomly sampled from the padded image or its horizontal flip. When testing, we only evaluate a single view of the original 32×32 image. We compare ResNet-110 (SVM) with several state-of-the-art deep models [35]–[39]. From Table VI, one can see that softmax loss performs similar to hinge loss, while ResNet-110 (F-SVM) outperforms ResNet-110 by 1.72% and ResNet-110 (SVM) by 1.22%. Besides, ResNet-110 (F-SVM) also performs better than most state-of-the-art methods, demonstrating the superiority of the incorporation of radius-margin based loss.

3) CIFAR-100: CIFAR-100 contains 32×32 color images with the same training/testing split to CIFAR-10, but has 100 classes. From Table VI, ResNet-110 (F-SVM) achieves the lowest error rate of 30.94% in comparison of the state of the arts. The improvement of ResNet-110 (F-SVM) is 3.40% over ResNet-110, and 2.44% over ResNet-110 (SVM).

4) Caltech101: The Caltech101 data set [26] consists of 9144 images from 102 classes. For each class, there are about 40 to 800 images, and we randomly select 30 samples per class for training. Table VII presents the results by our methods, the baselines and the state of the arts. For fairness, we compare AlexNet (F-SVM) with several AlexNet-based CNNs, including Zeiler and Fergus [40], DeCAF [41], and Chatfield *et al.* [42]. It can be seen from Table VII, AlexNet (F-SVM) achieves the error rate of 9.83%, and outperforms the other AlexNet-based methods. Specifically, AlexNet (F-SVM) outperforms its counterparts, i.e., AlexNet and AlexNet (SVM) with a large margin. Furthermore, we perform experiments on ResNet-50 pretrained on ImageNet and each image is resized to $224 \times 224 \times 3$ as input. As given in Table VII, ResNet-50

TABLE VI
TESTING ERROR RATE ON MNIST, CIFAR-10, AND CIFAR-100 DATA SETS

Methods	MNIST	CIFAR-10	CIFAR-100
Maxout [35]	0.45	9.38	38.57
NIN [37]	0.47	8.81	35.68
DSN [36]	0.39	7.97	34.57
dasNet [38]	-	9.22	33.78
Highway Network [39]	0.45	**7.54**	32.24
ResNet-110	0.45	9.51	34.34
ResNet-110 (SVM)	0.43	9.01	33.38
ResNet-110 (F-SVM)	**0.35**	7.79	**30.94**

TABLE VII
TESTING ERROR RATE ON CALTECH101 BY USING 30 TRAINING SAMPLES PER CLASS

Methods	Testing error (%)
Zeiler & Fergus [40]	13.50
DeCAF [41]	13.09
Chatfield *et al.* [42]	11.60
He *et al.* [43]	6.58
AlexNet	15.80
AlexNet (SVM)	14.50
AlexNet (F-SVM)	9.83
ResNet-50	8.20
ResNet-50 (SVM)	7.60
ResNet-50 (F-SVM)	**5.89**

(F-SVM) not only outperforms its counterparts, i.e., ResNet-50 and ResNet-50 (SVM), but also achieves the testing error rate of 5.89%, which is comparable to the state of the arts.

VI. CONCLUSION

In this paper, we proposed a convex radius-margin-based SVM model, dubbed as F-SVM, for joint learning of the feature transformation and SVM classifier. For the formulation of F-SVM, lower and upper bounds of the radius of the MEB are introduced to derive a novel radius approximation, and all the individual inequality constraints are combined into one integrated inequality constraint, resulting in a convex relaxation of the radius-margin-based SVM model. For model optimization, a semiwhitened PCA-based method is proposed for initialization, and a GBCD algorithm is adopted to learn the feature transformation and classifier. Further, F-SVM is kernelized by using kernel PCA. Experimental results show that F-SVM obtains higher classification accuracy than SVM and state-of-the-art radius-margin-based SVM methods [5], [7], and is efficient in training. In the future work, we will extend the relaxed radius-margin-based error bound to other learning models and extend F-SVM for learning other forms of feature transformation tailored for specific applications.

APPENDIX A

Lemma A.1: $\bar{R} \geq R$. *Proof:* Based on the definition of the radius, we get

$$\begin{aligned} R^2 &= \min_{\mathbf{x}_0} \max_i \|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_0\|_2^2 \\ &\leq \max_i \|\mathbf{A}\mathbf{x}_i - \mathbf{A}\bar{\mathbf{x}}\|_2^2 \\ &= \bar{R}^2. \end{aligned}$$

■

Denote by R_p the maximum pairwise distance. We have

$$R_p = \max_{i,j} \left\{\|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_j\|_2^2\right\}.$$

Lemma A.2 [7]: $R \geq R_p/2$.

Lemma A.3: $\bar{R} \leq R_p$.

Proof: Let $\mathbf{x}'_i = \mathbf{A}\mathbf{x}_i - \mathbf{A}\bar{\mathbf{x}}$. We have $\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_j = \mathbf{x}'_i - \mathbf{x}'_j$. Based on the definition of $\bar{R}$

$$\bar{R}^2 = \max_i \left\{\left\|\mathbf{x}'_i\right\|^2\right\} = \left\|\mathbf{x}'_{i^*}\right\|^2$$

we will prove that there exist some j^* which makes $\|\mathbf{x}'_{i^*} - \mathbf{x}'_{j^*}\|^2 \geq \|\mathbf{x}'_{i^*}\|^2$. Based on the definition of $\bar{\mathbf{x}}$, we have $\sum_j \mathbf{x}'_j = 0$. Then, we derive

$$\sum_j \mathbf{x}'_j \mathbf{x}'_{i^*} = 0 \Rightarrow \min_j \left\{\mathbf{x}'_j \mathbf{x}'_{i^*}\right\} = \mathbf{x}'_{j^*}\mathbf{x}'_{i^*} \leq 0.$$

Since $\|\mathbf{x}'_{j^*}\|^2 \geq 0$ and $-2\mathbf{x}'_{j^*}\mathbf{x}'_{i^*} \geq 0$, it can be seen that

$$\left\|\mathbf{x}'_{i^*} - \mathbf{x}'_{j^*}\right\|^2 \geq \left\|\mathbf{x}'_{i^*}\right\|^2.$$

Based on the definition of R_p

$$R_p^2 \geq \left\|\mathbf{x}'_{i^*} - \mathbf{x}'_{j^*}\right\|^2.$$

Combining the above two inequalities, we prove $\bar{R} \leq R_p$. ■

Finally, by combining **Lemmas** 1∼3, we get:

Theorem 1: The margin R is bounded by $\bar{R}$ by

$$\frac{1}{2}\overline{R} \leq R \leq \overline{R}.$$

APPENDIX B

Lemma B.1: The model in (11) can be equivalently reformulated into the F-SVM model in (12).

Proof: We define

$$\begin{aligned} f(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{M}) = \frac{1}{2}(\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}) + C\sum_{i=1}^n \xi_i \\ \text{s.t. } y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \\ \xi_i \geq 0, \quad i = 1, 2, \ldots, n \\ \mathbf{M} \succ 0. \end{aligned}$$

The model in (11) can be rewritten as

$$\min_{\mathbf{w},b,\boldsymbol{\xi},\mathbf{M}} \; f(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{M}), \quad \text{s.t. } \sum_{i=1}^n \omega_i(\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M}(\mathbf{x}_i - \bar{\mathbf{x}}) \leq \varepsilon$$

and the associated Lagrangian function [44] is defined as

$$\begin{aligned} &\min_{\mathbf{w},b,\boldsymbol{\xi},\mathbf{M}} \max_{\rho'} \; L(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{M}, \rho') \\ &= \frac{1}{2}(\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}) + C\sum_{i=1}^n \xi_i \\ &\quad + \rho'\left(\sum_{i=1}^n \omega_i(\mathbf{x}_i - \bar{\mathbf{x}})^\top \mathbf{M}(\mathbf{x}_i - \bar{\mathbf{x}}) - \varepsilon\right) \\ &\quad \text{s.t. } y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \; \forall i, \; \mathbf{M} \succ 0. \end{aligned}$$

According to [15], for every $\varepsilon \geq 0$, there exists some ρ corresponding to the corresponding optimal solution to ρ'.

With the optimal solution ρ, the problem above can be rewritten as

$$\min_{\mathbf{w},b,\boldsymbol{\xi},\mathbf{M}} \left\{ \frac{1}{2}(\mathbf{w}^\top \mathbf{M}^{-1}\mathbf{w}) + C\sum_{i=1}^{n}\xi_i + \rho \mathrm{tr}(\mathbf{MS}) - \rho\varepsilon \right\}$$
$$\text{s.t. } y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \ \forall i, \ \mathbf{M} \succ 0$$

where $\mathbf{S} = \sum_{i=1}^{n} \omega_i(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top$. Note that $\rho\varepsilon$ is independent of $\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{M}$ and can be safely discarded. Thus, the subproblem of $\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{M}$ can be reformulated into (12). ■

APPENDIX C

Lemma C.1 [45]: Given two SPD matrices $\mathbf{A}$ and $\mathbf{B}$, we have

$$(\mathbf{A}+\mathbf{B})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}(\mathbf{A}^{-1}+\mathbf{B}^{-1})^{-1}\mathbf{A}^{-1} \quad (27)$$
$$(\mathbf{A}+\mathbf{B})^{-1} = \mathbf{A}^{-1}(\mathbf{A}^{-1}+\mathbf{B}^{-1})^{-1}\mathbf{B}^{-1} = \mathbf{B}^{-1}(\mathbf{A}^{-1}+\mathbf{B}^{-1})^{-1}\mathbf{A}^{-1}. \quad (28)$$

Theorem 3: The F-SVM model in (12) is a convex optimization problem.

Proof: Note that all the constraints define a convex set, and $\sum_i \xi_i$ and $\mathrm{tr}(\mathbf{MS})$ are linear to $\boldsymbol{\xi}$ and $\mathbf{M}$, respectively. Then the key step is to prove that the function $\mathbf{w}^\top\mathbf{M}^{-1}\mathbf{w}$ is convex for $\mathbf{M} \succ 0$, i.e., for any $1 \geq \theta \geq 0$

$$\theta\mathbf{w}_1^\top\mathbf{M}_1^{-1}\mathbf{w}_1 + (1-\theta)\mathbf{w}_2^\top\mathbf{M}_2^{-1}\mathbf{w}_2 \geq (\theta\mathbf{w}_1 + (1-\theta)\mathbf{w}_2)^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1} \times(\theta\mathbf{w}_1 + (1-\theta)\mathbf{w}_2).$$

$(\theta\mathbf{w}_1 + (1-\theta)\mathbf{w}_2)^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1}(\theta\mathbf{w}_1 + (1-\theta)\mathbf{w}_2)$ contains three terms

$$\theta^2\mathbf{w}_1^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1}\mathbf{w}_1$$
$$(1-\theta)^2\mathbf{w}_2^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1}\mathbf{w}_2$$
$$\theta(1-\theta)\mathbf{w}_1^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1}\mathbf{w}_2.$$

First, we have

$$\begin{aligned}
&\theta^2(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1} \\
&= \theta\left(\mathbf{M}_1 + \frac{(1-\theta)}{\theta}\mathbf{M}_2\right)^{-1} \\
&= \theta\left(\mathbf{M}_1^{-1} - \mathbf{M}_1^{-1}\left(\mathbf{M}_1^{-1} + \left(\frac{(1-\theta)}{\theta}\mathbf{M}_2\right)^{-1}\right)^{-1}\mathbf{M}_1^{-1}\right) \\
&= \theta\mathbf{M}_1^{-1} - \mathbf{M}_1^{-1}((\theta\mathbf{M}_1)^{-1} + ((1-\theta)\mathbf{M}_2)^{-1})^{-1}\mathbf{M}_1^{-1}
\end{aligned} \quad (29)$$

and then we obtain

$$\theta\mathbf{w}_1^\top\mathbf{M}_1^{-1}\mathbf{w}_1 - \theta^2\mathbf{w}_1^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1}\mathbf{w}_1 = \mathbf{w}_1^\top\mathbf{M}_1^{-1}((\theta\mathbf{M}_1)^{-1} + ((1-\theta)\mathbf{M}_2)^{-1})^{-1}\mathbf{M}_1^{-1}\mathbf{w}_1. \quad (30)$$

Analogously, we obtain

$$(1-\theta)\mathbf{w}_2^\top\mathbf{M}_2^{-1}\mathbf{w}_2 - (1-\theta)^2\mathbf{w}_2^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1}\mathbf{w}_2 = \mathbf{w}_2^\top\mathbf{M}_2^{-1}((\theta\mathbf{M}_1)^{-1} + ((1-\theta)\mathbf{M}_2)^{-1})^{-1}\mathbf{M}_2^{-1}\mathbf{w}_2. \quad (31)$$

With (29), we have

$$\theta(1-\theta)\mathbf{w}_1^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1}\mathbf{w}_2 = \mathbf{w}_1^\top\mathbf{M}_1^{-1}((\theta\mathbf{M}_1)^{-1} + ((1-\theta)\mathbf{M}_2)^{-1})^{-1}\mathbf{M}_2^{-1}\mathbf{w}_2. \quad (32)$$

Combining (29)–(32), we obtain

$$\begin{aligned}
&\theta\mathbf{w}_1^\top\mathbf{M}_1^{-1}\mathbf{w}_1 + (1-\theta)\mathbf{w}_2^\top\mathbf{M}_2^{-1}\mathbf{w}_2 \\
&\quad -(\theta\mathbf{w}_1 + (1-\theta)\mathbf{w}_2)^\top(\theta\mathbf{M}_1 + (1-\theta)\mathbf{M}_2)^{-1} \\
&\quad \times(\theta\mathbf{w}_1 + (1-\theta)\mathbf{w}_2) \\
&= \mathbf{w}_1^\top\mathbf{M}_1^{-1}((\theta\mathbf{M}_1)^{-1} + ((1-\theta)\mathbf{M}_2)^{-1})^{-1}\mathbf{M}_1^{-1}\mathbf{w}_1 \\
&\quad +\mathbf{w}_2^\top\mathbf{M}_2^{-1}((\theta\mathbf{M}_1)^{-1} + ((1-\theta)\mathbf{M}_2)^{-1})^{-1}\mathbf{M}_2^{-1}\mathbf{w}_2 \\
&\quad -2\mathbf{w}_1^\top\mathbf{M}_1^{-1}((\theta\mathbf{M}_1)^{-1} + ((1-\theta)\mathbf{M}_2)^{-1})^{-1}\mathbf{M}_2^{-1}\mathbf{w}_2 \\
&= \left\|((\theta\mathbf{M}_1)^{-1} + ((1-\theta)\mathbf{M}_2)^{-1})^{-\frac{1}{2}}\left(\mathbf{M}_1^{-1}\mathbf{w}_1 - \mathbf{M}_2^{-1}\mathbf{w}_2\right)\right\|^2 \\
&\geq 0.
\end{aligned}$$

Thus, the F-SVM model is convex. ■

APPENDIX D

Theorem 4: Given a SPD matrix $\mathbf{S}$ and $\tau' > 0$, $\hat{\mathbf{B}}$ defined in (15) is the optimal solution to the problem

$$\hat{\mathbf{B}} = \arg\min_{\mathbf{B}}\{\mathcal{L}(\mathbf{B}, \tau') = \|\mathbf{B}\|_* + \tau'(\mathrm{tr}(\mathbf{B}^{-1}\mathbf{S}))\}.$$

Proof: $\mathcal{L}(\mathbf{B}, \tau')$ is strictly convex with respect to $\mathbf{B}$ [28]. Given $g(\mathbf{B}) = \tau'\mathrm{tr}(\mathbf{B}^{-1}\mathbf{S})$, we have

$$\frac{\partial g}{\partial \mathbf{B}} = -\tau'(\mathbf{B}^{-1}\mathbf{S}\mathbf{B}^{-1})^\top.$$

From [17] and [46], the set of subgradients of the nuclear norm $\partial\|\mathbf{B}\|_*$ can be represented as

$$\partial\|\mathbf{B}\|_* = \{\bar{\mathbf{U}}\bar{\boldsymbol{\Sigma}}\bar{\mathbf{U}}^\top + \mathbf{W} | \mathbf{W} \in \mathbf{R}^{d\times d}, \bar{\mathbf{U}}^\top\mathbf{W} = \mathbf{0}, \mathbf{W}\bar{\mathbf{U}} = \mathbf{0}, \|\mathbf{W}\|_2 \leq 1\}$$

where $\bar{\mathbf{U}}\bar{\boldsymbol{\Sigma}}\bar{\mathbf{U}}^\top$ is the eigenvalue decomposition of $\mathbf{B}$, each column of $\bar{\mathbf{U}}$ is a eigenvector, $\bar{\boldsymbol{\Sigma}}$ is a diagonal matrix with $\bar{\boldsymbol{\Sigma}} = \mathrm{Diag}(\sigma_1, \sigma_2, \ldots, \sigma_d)$ $(0 \leq \sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_d)$.

To prove that $\hat{\mathbf{B}}$ is the optimal solution, we will show that

$$\mathbf{0} \in -\tau'(\hat{\mathbf{B}}^{-1}\mathbf{S}\hat{\mathbf{B}}^{-1})^\top + \partial\|\hat{\mathbf{B}}\|_*$$

where $\mathbf{0}$ denote a zero matrix (i.e., a matrix consisting of all 0s). With the matrix $\hat{\mathbf{B}}$ in (14), we have

$$\tau'(\hat{\mathbf{B}}^{-1}\mathbf{S}\hat{\mathbf{B}}^{-1})^\top = \mathbf{U}\mathbf{U}^\top.$$

Let $\mathbf{W} = \mathbf{0}$. We have $\mathbf{U}^\top\mathbf{W} = \mathbf{0}$, $\mathbf{W}\mathbf{U} = \mathbf{0}$, and $\|\mathbf{W}\|_2 \leq 1$. Thus $\mathbf{U}\mathbf{U}^\top \in \partial\|\hat{\mathbf{B}}\|_*$, and $\hat{\mathbf{B}}$ is the optimal solution. ■

ACKNOWLEDGMENT

The authors would like to thank the associate editors and the anonymous reviewers for their constructive suggestions. The authors would also like to thank Dr. Edward C. Mignot from Shandong University, Jinan, China, for linguistic advice.

REFERENCES

[1] V. N. Vapnik, *Statistical Learning Theory*. New York, NY, USA: Wiley, 1998.

[2] H. Do, A. Kalousis, and M. Hilario, "Feature weighting using margin and radius based error bound optimization in SVMs," in *Proc. ECML PKDD*, 2009, pp. 315–329.

[3] J. Wu and H. Yang, "Linear regression-based efficient SVM learning for large-scale classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2357–2369, Oct. 2015.

[4] V. Vapnik and O. Chapelle, "Bounds on error expectation for support vector machines," *Neural Comput.*, vol. 12, no. 9, pp. 2013–2036, Sep. 2000.

[5] P. K. Shivaswamy and T. Jebara, "Maximum relative margin and data-dependent regularization," *J. Mach. Learn. Res.*, vol. 11, pp. 747–788, Feb. 2010.

[6] X. Zhu, P. Gong, Z. Zhao, and C. Zhang, "Learning similarity metric with SVM," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jun. 2012, pp. 1–8.

[7] H. Do and A. Kalousis, "Convex formulations of radius-margin based support vector machines," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2013, pp. 169–177.

[8] F. Wang, W. Zuo, L. Zhang, D. Meng, and D. Zhang, "A kernel classification framework for metric learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 9, pp. 1950–1962, Sep. 2015.

[9] C. Shen, J. Kim, F. Liu, L. Wang, and A. van den Hengel, "Efficient dual approach to distance metric learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 394–406, Feb. 2014.

[10] Z. Xu, K. Q. Weinberger, and O. Chapelle. (2012). "Distance metric learning for kernel machines." [Online]. Available: https://arxiv.org/abs/1208.3422

[11] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 131–159, 2002.

[12] H. Do, A. Kalousis, A. Woznica, and M. Hilario, "Margin and radius based multiple kernel learning," in *Proc. ECML PKDD*, vol. 5781. 2009, pp. 330–343.

[13] X. Liu, L. Wang, J. Yin, E. Zhu, and J. Zhang, "An efficient approach to integrating radius information into multiple kernel learning," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 557–569, Apr. 2013.

[14] K. Gai, G. Chen, and C.-S. Zhang, "Learning kernels with radiuses of minimum enclosing balls," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2010, pp. 649–657.

[15] J. Nocedal and S. Wright, *Numerical Optimization*. New York, NY, USA: Springer, 2006.

[16] R. A. Horn and C. R. Johnson, *Matrix Analysis*. New York, NY, USA: Cambridge Univ. Press, 1988.

[17] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010.

[18] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications," *Neural Netw.*, vol. 13, nos. 4–5, pp. 411–430, 2000.

[19] X. Shi, Z. Guo, F. Nie, L. Yang, J. You, and D. Tao, "Two-dimensional whitening reconstruction for enhancing robustness of principal component analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2130–2136, Oct. 2016.

[20] R. Girshick and J. Malik, "Training deformable part models with decorrelated features," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 3016–3023.

[21] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. Harandi, "Kernel methods on the riemannian manifold of symmetric positive definite matrices," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2013, pp. 73–80.

[22] Q. Wang, P. Li, L. Zhang, and W. Zuo, "Towards effective codebookless model for image classification," *Pattern Recognit.*, vol. 59, pp. 63–71, Nov. 2016.

[23] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM J. Imag. Sci.*, vol. 6, no. 3, pp. 1758–1789, 2013.

[24] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, "Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka–Łojasiewicz inequality," *Math. Oper. Res.*, vol. 35, no. 2, pp. 438–457, 2010.

[25] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009.

[26] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

[27] J. Hu, J. Lu, and Y.-P. Tan, "Discriminative deep metric learning for face verification in the wild," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 1875–1882.

[28] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 787–796.

[29] C. Ding, J. Choi, D. Tao, and L. S. Davis, "Multi-directional multi-level dual-cross patterns for robust face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 518–531, 2016.

[30] G. B. Huang, V. Jain, and E. Learned-Miller, "Unsupervised joint alignment of complex images," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Oct. 2007, pp. 1–8.

[31] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2009, pp. 365–372.

[32] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. Brit. Mach. Vis. Conf*, 2015, vol. 1. no. 3, pp. 1–12.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.

[35] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2013, pp. 1–9.

[36] C.-Y. Lee, S. Xie, P. W. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *Proc. Artif. Intell. Stat.*, 2015, pp. 562–570.

[37] M. Lin, Q. Chen, and S. Yan. (2013). "Network in network." [Online]. Available: https://arxiv.org/abs/1312.4400

[38] M. F. Stollenga, J. Masci, F. Gomez, and J. Schmidhuber, "Deep networks with internal selective attention through feedback connections," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 3545–3553.

[39] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2015, pp. 2377–2385.

[40] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 818–833.

[41] J. Donahue *et al.*, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2014, pp. 647–655.

[42] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. BMVC*, 2014, pp. 1–12.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 346–361.

[44] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[45] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," Inform. Math. Model., Tech. Univ. Denmark, Kongens Lyngby, Denmark, Tech. Rep., 2008. [Online]. Available: http://www2.imm.dtu.dk/pubdb/p.php?3274

[46] G. A. Watson, "Characterization of the subdifferential of some matrix norms," *Linear Algebra Appl.*, vol. 170, pp. 33–45, Jun. 1992.

Xiaohe Wu received the B.E. degree from the Harbin Institute of Technology, Harbin, China, in 2013, where she is currently pursuing the Ph.D. degree with the School of Computer Science and Technology.

In 2014, she was a Research Assistant with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. Her current research interests include support vector machines and single object visual tracking.

Wangmeng Zuo (M'09–SM'15) received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2007.

He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology. He has authored or co-authored over 70 papers in top-tier academic journals and conferences. His current research interests include image enhancement and restoration, image and face editing, object detection, visual tracking, and image classification.

Dr. Zuo was a Tutorial Organizer in ECCV 2016, an Associate Editor of the *IET Biometrics* and the *Journal of Electronic Imaging*, and the Guest Editor of the *Neurocomputing*, the *Pattern Recognition*, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.

Liang Lin (M'09–SM'15) was a Post-Doctoral Fellow at the University of California at Los Angeles, Los Angeles, CA, USA, from 2008 to 2010. He was a Senior Visiting Scholar with The Hong Kong Polytechnic University, Hong Kong, and The Chinese University of Hong Kong, Shenzhen, China, from 2014 to 2015. He is the Executive Research and Development Director of SenseTime Group Ltd., Shenzhen, and a Full Professor of Sun Yat-sen University, Guangzhou, China. He currently leads the SenseTime Research and Development Teams to develop cutting-edges and deliverable solutions on computer vision, data analysis and mining, and intelligent robotic systems. He has authored or co-authored over 100 papers in top-tier academic journals and conferences.

Dr. Lin is a fellow of IET. He is the Excellent Young Scientist of the National Natural Science Foundation of China. He served as the Area/Session Chair for numerous conferences, such as ICME, ACCV, and ICMR. He was a recipient of the Best Paper Runners-Up Award in ACM NPAR 2010, the Google Faculty Award in 2012, the Best Paper Diamond Award in IEEE ICME 2017, and the Hong Kong Scholars Award in 2014. He has been an Associate Editor of the IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS, *The Visual Computer*, and *Neurocomputing*.

Wei Jia (M'09) received the B.Sc. degree in informatics from Central China Normal University, Wuhan, China, in 1998, the M.Sc. degree in computer science from the Hefei University of Technology, Hefei, China, in 2004, and the Ph.D. degree in pattern recognition and intelligence system from the University of Science and Technology of China, Hefei, in 2008.

He was an Assistant Professor and an Associate Professor at the Hefei Institutes of Physical Science, Chinese Academy of Sciences, Beijing, China, from 2008 to 2016. He is currently an Associate Professor with the School of Computer and Information, Hefei University of Technology. His current research interests include computer vision, biometrics, pattern recognition, image processing, and machine learning.

David Zhang (F'09) received the bachelor's degree in computer science from Peking University, Beijing, China, the M.Sc. and Ph.D. degrees in computer science from the Harbin Institute of Technology (HIT), Harbin, China, in 1982 and 1985, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 1994.

From 1986 to 1988, he was a Post-Doctoral Fellow at Tsinghua University, Beijing, and then an Associate Professor at the Academia Sinica, Beijing. Since 2005, he has been a Chair Professor at The Hong Kong Polytechnic University, Hong Kong, where he is the Founding Director of the Biometrics Research Centre (UGC/CRC) supported by the Hong Kong SAR Government in 1998. He is a Professor at the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, China. He also serves as a Visiting Chair Professor at Tsinghua University, and Adjunct Professor at Peking University, Shanghai Jiao Tong University, Shanghai, China, HIT, and the University of Waterloo. He has authored or co-authored about 20 monographs and over 400 international journal papers. He holds around 40 patents from USA/Japan/Hong Kong/China.

Dr. Zhang is a Croucher Senior Research Fellow, a Distinguished Speaker of the IEEE Computer Society, and a fellow of IAPR. He is the Founder and an Editor-in-Chief of the *International Journal of Image and Graphics*, the Founder and a Series Editor of the *International Series on Biometrics (KISB)* (Springer), an Organizer of the International Conference on Biometrics Authentication, and an associate editor of more than ten international journals including IEEE transactions and so on. He was selected as a Highly Cited Researcher in Engineering by Thomson Reuters from 2014 to 2017.