Distance Metric Learning via Iterated Support Vector Machines

Wangmeng Zuo, *Senior Member, IEEE*, Faqiang Wang, David Zhang, *Fellow, IEEE*, Liang Lin, Yuchi Huang, Deyu Meng, and Lei Zhang, *Senior Member, IEEE*

***Abstract*— Distance metric learning aims to learn from the given training data a valid distance metric, with which the similarity between data samples can be more effectively evaluated for classification. Metric learning is often formulated as a convex or nonconvex optimization problem, while most existing methods are based on customized optimizers and become inefficient for large scale problems. In this paper, we formulate metric learning as a kernel classification problem with the positive semi-definite constraint, and solve it by iterated training of support vector machines (SVMs). The new formulation is easy to implement and efficient in training with the off-the-shelf SVM solvers. Two novel metric learning models, namely positive-semidefinite constrained metric learning (PCML) and nonnegative-coefficient constrained metric learning (NCML), are developed. Both PCML and NCML can guarantee the global optimality of their solutions. Experiments are conducted on general classification, face verification, and person re-identification to evaluate our methods. Compared with the state-of-the-art approaches, our methods can achieve comparable classification accuracy and are efficient in training.**

***Index Terms*— Metric learning, support vector machine, kernel method, Lagrange duality, alternating minimization.**

I. INTRODUCTION

DISTANCE metric learning aims to train a valid distance metric which can enlarge the distances between samples of different classes while reducing the distances between samples of the same class [1]. Metric learning is closely related to other learning problems, including k-Nearest Neighbor (k-NN) classification [2] and clustering [3], and has also been widely applied in many image classification tasks, *e.g.*, face recognition [4] and person re-identification [5], [6]. One popular metric learning approach is Mahalanobis distance metric learning, which is to learn a linear transformation matrix $\mathbf{L}$ or a matrix $\mathbf{M} = \mathbf{L}^T\mathbf{L}$ from the training data. Given two samples $\mathbf{x}_i$ and $\mathbf{x}_j$, their Mahalanobis distance is defined as:

$$d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right) = \left\|\mathbf{L}(\mathbf{x}_i - \mathbf{x}_j)\right\|_2^2 = \left(\mathbf{x}_i - \mathbf{x}_j\right)^T \mathbf{M}\left(\mathbf{x}_i - \mathbf{x}_j\right). \quad (1)$$

To satisfy the nonnegative property of a distance metric, $\mathbf{M}$ should be positive semidefinite (PSD). According to which one of $\mathbf{M}$ and $\mathbf{L}$ is learned, Mahalanobis distance metric learning methods can be grouped into two categories. Methods that learn $\mathbf{L}$, including neighborhood components analysis (NCA) [7], large margin components analysis (LMCA) [8] and neighborhood repulsed metric learning (NRML) [9], are mostly formulated as nonconvex optimization problems, which are solved by gradient descent optimizers. Taking the PSD constraint into account, methods that learn $\mathbf{M}$, including large margin nearest neighbor (LMNN) [10] and maximally collapsing metric learning (MCML) [11], are mostly formulated as convex semidefinite programming (SDP), which can be optimized by standard SDP solvers [10], projected gradient [3], Boosting-like [12], or Frank-Wolfe [13] algorithms. Davis *et al.* [14] proposed an information-theoretic metric learning (ITML) model with an iterative Bregman projection algorithm to avoid the projections onto the PSD cone. Besides, the use of online solvers has been discussed in [5] and [15]–[17].

On the other hand, the Mahalanobis distance in (1) can be equivalently written as:

$$\begin{aligned} d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right) &= \operatorname{tr}\left(\mathbf{M}^T(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T\right) \\ &= \left\langle \mathbf{M}, \mathbf{X}_{ij} \right\rangle, \end{aligned} \quad (2)$$

where $\mathbf{M}$ is a PSD matrix, $\mathbf{X}_{ij} = (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$, $\langle \mathbf{A}, \mathbf{B} \rangle = \operatorname{tr}\left(\mathbf{A}^T\mathbf{B}\right)$ is defined as the Frobenius inner product of two matrices $\mathbf{A}$ and $\mathbf{B}$, and $\operatorname{tr}(\bullet)$ stands for the matrix trace operator. By defining the following kernel function

$$\begin{aligned} K\left(\left(\mathbf{x}_i, \mathbf{x}_j\right), (\mathbf{x}_k, \mathbf{x}_l)\right) &= \left\langle \mathbf{X}_{ij}, \mathbf{X}_{kl} \right\rangle \\ &= \left(\left(\mathbf{x}_i - \mathbf{x}_j\right)^T (\mathbf{x}_k - \mathbf{x}_l)\right)^2, \end{aligned} \quad (3)$$

we can cast the Mahalanobis distance in (2) as a kernel classifier. For convenience, we rewrite $K\left(\left(\mathbf{x}_i, \mathbf{x}_j\right), (\mathbf{x}_k, \mathbf{x}_l)\right)$ as K_{ijkl} in the following sections.

Manuscript received March 21, 2016; revised October 27, 2016 and June 24, 2017; accepted June 26, 2017. Date of publication July 11, 2017; date of current version August 7, 2017. This work was supported in part by the National Key R&D Program of China under Grant 2017YFC0113000 and 2016YFB1001004, in part by NSFC under Grant 61671182, in part by the Guangdong Natural Science Foundation under Grant 2015A030313129, and in part by the Hong Kong Research Grants Council General Research Fund under Grant PolyU 152212/14E. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Shiguang Shan. *(Corresponding author: Wangmeng Zuo.)*

W. Zuo and F. Wang are with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: cswmzuo@gmail.com; tshfqw@163.com).

D. Zhang and L. Zhang are with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong (e-mail: csdzhang@comp.polyu.edu.hk; cslzhang@comp.polyu.edu.hk).

L. Lin is with the Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Sun Yat-sen University, Guangzhou 510275, China (e-mail: linliang@ieee.org).

Y. Huang is with the Research Division, Educational Testing Service, Princeton, NJ 08541 USA (e-mail: yuchi.huang@foxmail.com).

D. Meng is with the Institute of Information and System Sciences, Faculty of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: dymeng@mail.xjtu.edu.cn).

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/TIP.2017.2725578

As kernel methods [18], [19] have been widely studied in many learning tasks, e.g., semi-supervised learning, multiple instance learning, multitask learning, etc. Kernel learning methods, such as support vector machine (SVM), exhibit good generalization performance. There are many open resources on kernel classification methods, and a variety of toolboxes and libraries have been released [20]–[24]. It is thus important to investigate the connections between metric learning and kernel classification and explore how to utilize the kernel classification resources in the research and development of new metric learning methods. Wang *et al.* [25] made an attempt on developing a kernel classification framework for metric learning. However, in their heuristic two-step greedy scheme, the PSD constraint is ignored in the first step, and then they simply project the learned matrix onto the PSD cone to obtain the final valid distance metric.

In this paper, we propose a novel formulation of metric learning by casting it as a kernel classification problem with PSD constraint, which allows us to effectively and efficiently learn valid distance metrics by iterated training of SVM. The off-the-shelf SVM solvers such as LibSVM [20] can be employed to solve the metric learning problem. Specifically, we propose two novel methods to bridge metric learning with the well-developed SVM techniques, and they are easy to implement. First, we propose a Positive-semidefinite Constrained Metric Learning (PCML) model, which can be solved via iterating between PSD projection and dual SVM learning. Second, by re-parameterizing the matrix $\mathbf{M}$, we propose a Nonnegative-coefficient Constrained Metric Learning (NCML) model, which can be solved by iterated learning of two SVMs. Both PCML and NCML have globally optimal solutions. Compared with [25], our PCML and NCML provide principled schemes to exploit SVM solver for metric learning with guarantee on global optimum. The main abbreviations used in this paper are summarized in Table I. Our experiments on general classification, face verification and person re-identification tasks clearly demonstrate the effectiveness of our methods. The contribution of this work is three-fold:

1) Two models, *i.e.*, PCML and NCML, are proposed by formulating metric learning as kernel classification problem with PSD constraint. Both PCML and NCML models are convex, and can guarantee the PSD property of the learned distance metric.
2) An optimization algorithm is developed for solving PCML by iterating between SVM training and PSD projection. It has the computational complexity of $O(d^3)$ per iteration w.r.t the feature dimension d, and can converge to global optimum.
3) An optimization algorithms is developed for NCML by iterating between the training of two SVMs. It has the computational complexity of $O(d)$ per iteration w.r.t d, and can guarantee the global optimality of the solution.

TABLE I
SUMMARY OF MAIN ABBREVIATIONS

Abbreviation	Full Name
PSD	Positive semidefinite (matrix)
SDP	Semidefinite programming
k-NN	k-nearest neighbor (classification)
KKT	Karush-Kuhn-Tucker (condition)
SVM	Support vector machine
LMCA [8]	Large margin components analysis
LMNN [2]	Large margin nearest neighbor
NCA [7]	Neighborhood components analysis
MCML [11]	Maximally collapsing metric learning
ITML [14]	Information-theoretic metric learning
LDML [4]	Logistic discriminant metric learning
DML-eig [13]	Distance metric learning with eigenvalue optimization
PLML [26]	Parametric local metric learning
KISSME [5]	Keep it simple and straightforward metric learning
DDML [27]	Discriminative deep metric learning
TSML [28]	Triangular similarity metric learning
LM3L [29]	Large margin multi-metric learning
LOMO [30]	Local maximal occurrence (representation)
XQDA [30]	Cross-view quadratic discriminant analysis
CNN	Convolutional neural network
PCML	Positive-semidefinite constrained metric learning
NCML	Nonnegative-coefficient constrained metric learning

II. RELATED WORK

Compared with nonconvex metric learning models [7], [8], convex formulation of metric learning [2], [3], [11]–[13] has drawn increasing attentions due to its desired properties such as global optimality. Most convex models can be formulated as SDP or quadratic SDP problems. Standard SDP solvers, however, are inefficient for metric learning, especially when the size of training samples is big or the feature dimension is high. Therefore, customized optimizer is developed for each specific metric learning model. For LMNN, Weinberger and Saul [31] developed an efficient solver based on sub-gradient descent and active set. In ITML, Davis *et al.* [14] suggested an iterative Bregman projection algorithm. Iterative projected gradient descent method [3], [32] has been widely employed for metric learning but it requires an eigenvalue decomposition in each iteration. Other algorithms such as block-coordinate descent [33], smooth optimization [34], and Frank-Wolfe [13] have also been studied for metric learning. Unlike the customized algorithms, we formulate metric learning as a kernel classification problem with PSD constraint and solve it using the off-the-shelf SVM solvers, which can guarantee the global optimality and the PSD property of the learned $\mathbf{M}$, and is easy to implement and efficient in training.

Another line of work aims to develop metric learning algorithms by solving the Lagrange dual problems. Shen *et al.* derived the Lagrange dual of the exponential loss based metric learning model, and proposed a boosting-like approach, namely BoostMetric [12], [35]. MetricBoost [36] and FrobMetric [37], [38] were further proposed to improve BoostMetric. Liu and Vemuri incorporated two regularization terms in the duality for robust metric learning [39]. Note that BoostMetric [12], [35], MetricBoost [36], and FrobMetric [37] are proposed for metric learning with triplet constraints, whereas in many applications such as verification, only pairwise constraints are available in the training stage.

Studies have also given to connect SVM with metric learning [40]–[42]. Using SVM, Nguyen and Guo [40] formulated metric learning as a quadratic SDP, and adopted a projected gradient descent algorithm. They select the farthest neighbors for each sample to construct similar pairs, while we select the nearest neighbors in PCML and NCML. Moreover, the formulations and optimizers of our models are different from the model in [40]. Brunner *et al.* [41] proposed a pairwise SVM to learn a dissimilarity function. Their metric learning pairwise kernel is similar to that used in our models, but the PSD property is not considered in their model. Do *et al.* [42] analyzed the relation of LMNN and SVM, where LMNN is interpreted as the joint learning of multiple local SVM-like models. By studying SVM from a metric learning perspective, they presented an improved SVM for single sample classification. Different with [42], we explain metric learning as a SVM for sample pair classification with the PSD constraint, and propose two novel metric learning methods, *i.e.*, PCML and NCML, together with optimization algorithms.

Besides, deep learning has been exploited for nonlinear metric learning. Based on the max-margin loss, Hu *et al.* [27] suggested a discriminative deep metric learning (DDML) method, and achieved state-of-the-art performance for face verification in the wild. Hoffer and Ailon [43] trained deep convolutional network (CNN) with the triplet loss for learning deep feature representation. Unlike these methods, we study Mahalanobis distance metric learning from a kernel classification perspective, and can guarantee to obtain the global optimal solutions. For person re-identification, subspace learning can also be an alternative of metric learning and recently has achieved the state-of-the-art performance [30], [44].

III. Positive-Semidefinite Constrained Metric Learning (PCML)

In this section, we formulate metric learning as a convex SDP, and propose the PCML model. We then develop a learning algorithm by alternatively iterating between SVM training and PSD projection, and discuss the convergence of PCML.

A. PCML and Its Dual Problem

Denote by $\{(\mathbf{x}_i, y_i) | i = 1, 2, \ldots, N\}$ a training set, where $\mathbf{x}_i \in \mathbb{R}^d$ is the ith training sample, and y_i is the class label of $\mathbf{x}_i$. Let $\mathcal{S} = \{(\mathbf{x}_i, \mathbf{x}_j) : \mathbf{x}_i$ and $\mathbf{x}_j$ have the sameclass label $\}$ be the set of similar pairs, $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{x}_j) : \mathbf{x}_i$ and $\mathbf{x}_j$ have differentclass labels$\}$ be the set of dissimilar pairs, and b is the distance threshold. We hope the Mahalanobis distance of a similar pair should be lower than $b-1$, and that of a dissimilar pair should be higher than $b+1$ (see Fig. 1).

By introducing an indicator variable h_{ij},

$$h_{ij} = \begin{cases} 1, & \text{if } (\mathbf{x}_i,\mathbf{x}_j) \in \mathcal{D} \\ -1, & \text{if } (\mathbf{x}_i,\mathbf{x}_j) \in \mathcal{S}, \end{cases} \quad (4)$$

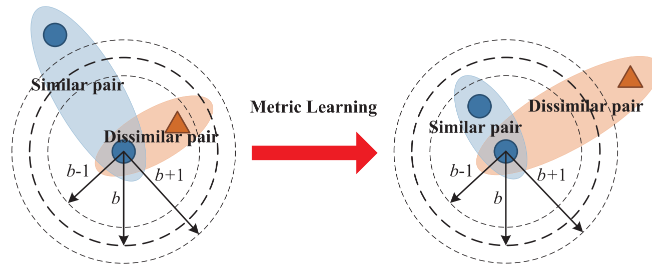


Fig. 1. Schematic illustration of the constraints of similar and dissimilar pairs.

the PCML model can be formulated as:

$$\begin{aligned} &\min_{\mathbf{M},b,\boldsymbol{\xi}} \frac{1}{2} \|\mathbf{M}\|_F^2 + C \sum\nolimits_{i,j} \xi_{ij} \\ &\text{s.t. } h_{ij}\left(\langle \mathbf{M}, \mathbf{X}_{ij} \rangle - b\right) \geq 1 - \xi_{ij}, \quad \xi_{ij} \geq 0, \ \forall i,j \\ &\quad\ \mathbf{M} \succcurlyeq 0, \end{aligned} \quad (5)$$

where ξ_{ij} denotes the slack variables, and $\|\bullet\|_F$ denotes the Frobenius norm.

The PCML model is convex and can be solved by standard SDP solvers. However, the high complexity of general-purpose interior-point SDP solver makes it only suitable for small-scale problems. In order to improve the efficiency, we first analyze the Lagrange duality of the PCML model, and then propose an algorithm to iterate between SVM training and PSD projection to learn the distance metric.

By introducing the Lagrange multipliers $\boldsymbol{\lambda}$ and PSD matrix $\mathbf{Y}$, the Lagrange dual of the problem in (5) can be written as:

$$\begin{aligned} &\max_{\boldsymbol{\lambda},\mathbf{Y}} -\frac{1}{2} \left\| \sum\nolimits_{i,j} \lambda_{ij} h_{ij} \mathbf{X}_{ij} + \mathbf{Y} \right\|_F^2 + \sum\nolimits_{i,j} \lambda_{ij} \\ &\text{s.t. } \sum\nolimits_{i,j} \lambda_{ij} h_{ij} = 0, \quad 0 \leq \lambda_{ij} \leq C, \ \forall i,j, \ \mathbf{Y} \succcurlyeq 0. \end{aligned} \quad (6)$$

Please refer to **Appendix A** for the detailed derivation of the dual problem. Based on the Karush-Kuhn-Tucker (KKT) conditions, the matrix $\mathbf{M}$ can be obtained by

$$\mathbf{M} = \sum\nolimits_{i,j} \lambda_{ij} h_{ij} \mathbf{X}_{ij} + \mathbf{Y}, \quad (7)$$

and the distance threshold b can be obtained by (43) in **Appendix A**. The strong duality allows us to first solve the dual problem in (6) and obtain the matrix $\mathbf{M}$ by (7).

B. Alternating Optimization Algorithm

To solve the dual problem efficiently, we propose an optimization approach by updating $\boldsymbol{\lambda}$ and $\mathbf{Y}$ alternatively. Given $\mathbf{Y}$, we introduce a new variable $\boldsymbol{\eta}$ with $\eta_{ij} = 1 - h_{ij}\langle \mathbf{X}_{ij}, \mathbf{Y} \rangle = 1 - h_{ij}\left(\mathbf{x}_i - \mathbf{x}_j\right)^T \mathbf{Y} \left(\mathbf{x}_i - \mathbf{x}_j\right)$. With the kernel function in (3), the subproblem on $\boldsymbol{\lambda}$ can be formulated as the following QP problem:

$$\begin{aligned} &\max_{\boldsymbol{\lambda}} -\frac{1}{2} \sum\nolimits_{i,j,k,l} \lambda_{ij} \lambda_{kl} h_{ij} h_{kl} K_{ijkl} + \sum\nolimits_{i,j} \eta_{ij} \lambda_{ij} \\ &\text{s.t. } \sum\nolimits_{i,j} \lambda_{ij} h_{ij} = 0, \quad 0 \leq \lambda_{ij} \leq C, \ \forall i,j. \end{aligned} \quad (8)$$

This subproblem on $\boldsymbol{\lambda}$ is a kernel-based classification problem, and can be efficiently solved by using the existing

Algorithm 1 Algorithm of PCML

Input: $\mathcal{S} = \{(\mathbf{x}_i, \mathbf{x}_j)$: the class labels of $\mathbf{x}_i$ and $\mathbf{x}_j$ are the same$\}$, $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{x}_j)$: the class labels of $\mathbf{x}_i$ and $\mathbf{x}_j$ are different$\}$, and h_{ij}.
Output: $\mathbf{M}$.
Initialize $\mathbf{Y}^{(0)}$, $t \leftarrow 0$.
repeat
1. Update $\eta^{(t+1)}$ with $\eta_{ij}^{(t+1)} = 1 - h_{ij} \langle \mathbf{X}_{ij}, \mathbf{Y}^{(t)} \rangle$.
2. Update $\boldsymbol{\lambda}^{(t+1)}$ by solving the subproblem (7) using an SVM solver.
3. Update $\mathbf{Y}_0^{(t+1)} = -\sum_{i,j} \lambda_{ij}^{(t+1)} h_{ij} \mathbf{X}_{ij}$.
4. Update $\mathbf{Y}^{(t+1)} = \mathbf{U}^{(t+1)} \boldsymbol{\Lambda}_+^{(t+1)} \mathbf{U}^{(t+1)T}$, where $\mathbf{Y}_0^{(t+1)} = \mathbf{U}^{(t+1)} \boldsymbol{\Lambda}^{(t+1)} \mathbf{U}^{(t+1)T}$ and $\boldsymbol{\Lambda}_+^{(t+1)} = \max\left(\boldsymbol{\Lambda}^{(t+1)}, \mathbf{0}\right)$.
5. $t \leftarrow t + 1$.

until convergence
$\mathbf{M} = \sum_{i,j} \lambda_{ij}^{(t-1)} h_{ij} \mathbf{X}_{ij} + \mathbf{Y}^{(t-1)}$.
return $\mathbf{M}$

SVM solvers [20]. Given $\boldsymbol{\lambda}$, the subproblem on $\mathbf{Y}$ can be formulated as the projection onto the convex cone of PSD matrices:

$$\min_{\mathbf{Y}} \quad \|\mathbf{Y} - \mathbf{Y}_0\|_F^2, \quad \text{s.t.} \quad \mathbf{Y} \succcurlyeq 0, \tag{9}$$

where $\mathbf{Y}_0 = -\sum_{i,j} \lambda_{ij} h_{ij} \mathbf{X}_{ij}$. Through the eigen-decomposition of $\mathbf{Y}_0$, i.e., $\mathbf{Y}_0 = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$ and $\boldsymbol{\Lambda}$ is the diagonal matrix of eigenvalues, the solution to (9) can be expressed as $\mathbf{Y} = \mathbf{U}\boldsymbol{\Lambda}_+\mathbf{U}^T$, where $\boldsymbol{\Lambda}_+ = \max(\boldsymbol{\Lambda}, \mathbf{0})$. Finally, the PCML algorithm is summarized in **Algorithm 1**.

C. Optimality Condition

Our algorithms can be treated as an implementation of generalized block coordinate descent (GBCD) [45] with two blocks. In our algorithms, the optimal solution to each sub-problem is obtained. As stated in [45], when the objective function is strongly convex, GBCD can converge to the global optimal solution. Therefore, the proposed algorithm can reach the global optimum of the problems in (5) and (6).

Moreover, the optimality condition of our algorithm can be checked by the duality gap in each iteration, which is defined as the difference between the primal and dual objective values:

$$\text{DualGap}_{\text{PCML}}^{(n)} = \frac{1}{2}\left\|\mathbf{M}^{(n)}\right\|_F^2 + C\sum\nolimits_{i,j} \xi_{ij}^{(n)} - \sum\nolimits_{i,j} \lambda_{ij}^{(n)} + \frac{1}{2}\left\|\sum\nolimits_{i,j} \lambda_{ij}^{(n)} h_{ij}\mathbf{X}_{ij} + \mathbf{Y}^{(n)}\right\|_F^2, \tag{10}$$

where $\mathbf{M}^{(n)}$, $\boldsymbol{\xi}^{(n)}$, $\boldsymbol{\lambda}^{(n)}$, and $\mathbf{Y}^{(n)}$ are feasible primal and dual variables, and $\text{DualGap}_{\text{PCML}}^{(n)}$ is the duality gap in the nth iteration. According to (7), we can derive that

$$\mathbf{M}^{(n)} = \sum\nolimits_{i,j} \lambda_{ij}^{(n)} h_{ij}\mathbf{X}_{ij} + \mathbf{Y}^{(n)} = \mathbf{Y}^{(n)} - \mathbf{Y}_0^{(n)}. \tag{11}$$

As shown in Sec. III.B, $\mathbf{Y}_0^{(n)} = \mathbf{U}^{(n)}\boldsymbol{\Lambda}^{(n)}\mathbf{U}^{(n)T}$, $\mathbf{Y}^{(n)} = \mathbf{U}^{(n)}\boldsymbol{\Lambda}_+^{(n)}\mathbf{U}^{(n)T}$, and hence $\mathbf{M}^{(n)} = \mathbf{U}^{(n)}\boldsymbol{\Lambda}_-^{(n)}\mathbf{U}^{(n)T}$, where

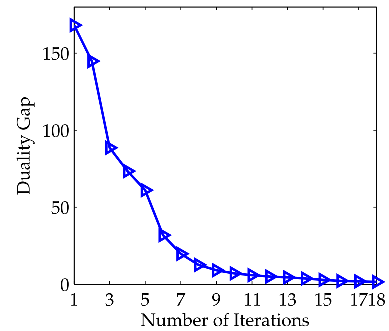


Fig. 2. Duality gap vs. number of iterations on the *PenDigits* database for PCML.

$\boldsymbol{\Lambda}_-^{(n)} = \boldsymbol{\Lambda}_+^{(n)} - \boldsymbol{\Lambda}^{(n)}$. Thus, $\left\|\mathbf{M}^{(n)}\right\|_F^2$ can be computed by

$$\begin{aligned}\left\|\mathbf{M}^{(n)}\right\|_F^2 &= \text{tr}\left(\mathbf{M}^{(n)T}\mathbf{M}^{(n)}\right) \\ &= \text{tr}\left(\mathbf{U}^{(n)}\boldsymbol{\Lambda}_-^{(n)}\mathbf{U}^{(n)T}\mathbf{U}^{(n)}\boldsymbol{\Lambda}_-^{(n)}\mathbf{U}^{(n)T}\right) \\ &= \text{tr}\left(\mathbf{U}^{(n)}\boldsymbol{\Lambda}_-^{(n)2}\mathbf{U}^{(n)T}\right) = \text{tr}\left(\boldsymbol{\Lambda}_-^{(n)2}\right).\end{aligned} \tag{12}$$

Substituting (11) and (12) into (10), the duality gap of PCML can be obtained as follows

$$\text{DualGap}_{\text{PCML}}^{(n)} = C\sum\nolimits_{i,j} \xi_{ij}^{(n)} - \sum\nolimits_{i,j} \lambda_{ij}^{(n)} + \text{tr}\left(\boldsymbol{\Lambda}_-^{(n)^2}\right). \tag{13}$$

Based on the KKT conditions of the PCML dual problem in (6), $\xi_{ij}^{(n)}$ can be obtained by

$$\xi_{ij}^{(n)} = \begin{cases} 0, & \forall \lambda_{ij}^{(n)} < C \\ \left[1 - h_{ij}\left(\left\langle \mathbf{M}^{(n)}, \mathbf{X}_{ij}\right\rangle - b^{(n)}\right)\right]_+, & \forall \lambda_{ij}^{(n)} = C \end{cases} \tag{14}$$

$$b^{(n)} = \left\langle \mathbf{M}^{(n)}, \mathbf{X}_{ij}\right\rangle - 1/h_{ij}, \quad \forall 0 < \lambda_{ij}^{(n)} < C. \tag{15}$$

Please refer to **Appendix A** for the detailed derivation of $\xi_{ij}^{(n)}$ and $b^{(n)}$. The duality gap is always nonnegative and approaches to zero when the primal problem is convex. Thus, it can be used as the termination condition of the algorithm. Fig. 2 plots the curve of duality gap versus the number of iterations on the *PenDigits* database by PCML. The duality gap approaches to zero in less than 20 iterations and our algorithm will reach the global optimum [46, Ch. 5]. In **Algorithm 1**, we adopt the following termination condition:

$$\text{DualGap}_{\text{PCML}}^{(t)} < \varepsilon \cdot \text{DualGap}_{\text{PCML}}^{(1)}, \tag{16}$$

where ε is a small constant and we set $\varepsilon = 0.01$.

D. Remarks

1) Construction of Pairwise Constraints: Based on the training set, N^2 pairwise constraints can be introduced in total. However, in practice we only need to choose a subset of pairwise constraints to reduce the computational cost.

For each sample, we find its k nearest neighbors with the same label to construct similar pairs and its k nearest neighbors with different labels to construct dissimilar pairs. Thus, we only need $2kN$ pairwise constraints, and we can reduce the scale of pairwise constraints from $O\left(N^2\right)$ to $O\left(kN\right)$. Since k is usually small (i.e., $1\sim3$), the computational cost of metric learning is much reduced. Similar strategy for constructing pairwise or triplet constraints can be found in [2] and [47].

2) Computational Complexity: We use the LibSVM library for SVM training. The computational complexity of SMO-type algorithms [48] is $O(k^2N^2d)$. For PSD projection, the complexity of conventional SVD algorithms is $O(d^3)$.

IV. Nonnegative-Coefficient Constrained Metric Learning (NCML)

In PCML, the computational complexity of the PSD projection is $O(d^3)$, which limits the training efficiency for data with high dimension. Therefore, we propose a NCML model, in which we re-parameterize the matrix $\mathbf{M}$ as the linear combination of a series of rank-1 matrices, and let the coefficients to be nonnegative to guarantee the PSD property of the matrix $\mathbf{M}$. NCML does not need any PSD projection in training and has low computational complexity w.r.t. d.

Given a set of rank-1 PSD matrices $\mathbf{M}_t = \mathbf{m}_t\mathbf{m}_t^T$ ($t = 1, \cdots, T$), a linear combination of $\mathbf{M}_t$ is defined as $\mathbf{M} = \sum_t \alpha_t \mathbf{M}_t$, where α_t is the scalar coefficient. One can easily prove the following theorem.

Theorem 1: If the scalar coefficient $\alpha_t \geq 0$, $\forall t$, the matrix $\mathbf{M} = \sum_t \alpha_t \mathbf{M}_t$ is PSD, where $\mathbf{M}_t = \mathbf{m}_t\mathbf{m}_t^T$ is a rank-1 PSD matrix.

Proof: Denote by $\mathbf{u} \in \mathbb{R}^d$ a random vector. Based on the expression of $\mathbf{M}$, we have:

$$\mathbf{u}^T\mathbf{M}\mathbf{u} = \mathbf{u}^T\left(\sum\nolimits_t \alpha_t \mathbf{m}_t\mathbf{m}_t^T\right)\mathbf{u} = \sum\nolimits_t \alpha_t \mathbf{u}^T\mathbf{m}_t\mathbf{m}_t^T\mathbf{u} = \sum\nolimits_t \alpha_t\left(\mathbf{u}^T\mathbf{m}_t\right)^2.$$

Since $\left(\mathbf{u}^T\mathbf{m}_t\right)^2 \geq 0$ and $\alpha_t \geq 0$, $\forall t$, we have $\mathbf{u}^T\mathbf{M}\mathbf{u} \geq 0$. Therefore, $\mathbf{M}$ is a PSD matrix. ■

A. NCML and Its Dual Problem

Motivated by **Theorem 1**, we impose the PSD constraint by re-parameterizing the distance metric $\mathbf{M}$, and develop a nonnegative-coefficient constrained metric learning (NCML) method to learn the PSD matrix $\mathbf{M}$. Given the training data $\mathcal{S}$ and $\mathcal{D}$, a rank-1 PSD matrix $\mathbf{X}_{ij}$ can be constructed for each pair $\left(\mathbf{x}_i, \mathbf{x}_j\right)$. By assuming that the learned matrix should be the linear combination of $\mathbf{X}_{ij}$ with the nonnegative coefficient constraint, the NCML model is formulated as:

$$\begin{aligned}\min_{\mathbf{M},b,\boldsymbol{\alpha},\boldsymbol{\xi}} \ & \frac{1}{2}\|\mathbf{M}\|_F^2 + C\sum\nolimits_{i,j}\xi_{ij}\\ \text{s.t. } & h_{ij}\left(\left\langle\mathbf{M},\mathbf{X}_{ij}\right\rangle - b\right) \geq 1-\xi_{ij}, \alpha_{ij}\geq 0, \xi_{ij}\geq 0, \quad \forall i,j\\ & \mathbf{M} = \sum\nolimits_{i,j}\alpha_{ij}\mathbf{X}_{ij}. \end{aligned} \tag{17}$$

By substituting $\mathbf{M}$ with $\sum_{i,j}\alpha_{ij}\mathbf{X}_{ij}$, we reformulate NCML as:

$$\begin{aligned}\min_{\boldsymbol{\alpha},b,\boldsymbol{\xi}} \ & \frac{1}{2}\sum\nolimits_{i,j}\sum\nolimits_{k,l}\alpha_{ij}\alpha_{kl}K_{ijkl} + C\sum\nolimits_{i,j}\xi_{ij}\\ \text{s.t. } & h_{ij}\left(\sum\nolimits_{k,l}\alpha_{kl}K_{ijkl} - b\right) \geq 1-\xi_{ij}\\ & \alpha_{ij}\geq 0, \ \xi_{ij}\geq 0, \quad \forall i,j. \end{aligned} \tag{18}$$

By introducing the Lagrange multipliers $\boldsymbol{\eta}$ and $\boldsymbol{\beta}$, and the kernel function in (3), the Lagrange dual of the primal problem in (18) can be formulated as:

$$\begin{aligned}\max_{\boldsymbol{\eta},\boldsymbol{\beta}} \ & -\frac{1}{2}\sum_{i,j,k,l}\left(\beta_{ij}h_{ij}+\eta_{ij}\right)\left(\beta_{kl}h_{kl}+\eta_{kl}\right)K_{ijkl} + \sum_{i,j}\beta_{ij}\\ \text{s.t. } & \sum_{k,l}\eta_{kl}K_{ijkl}\geq 0, \ 0\leq\beta_{ij}\leq C, \quad \forall i,j\\ & \sum_{i,j}\beta_{ij}h_{ij} = 0. \end{aligned} \tag{19}$$

Please refer to **Appendix B** for the detailed derivation of the dual problem. Based on the KKT conditions, the coefficient α_{ij} can be obtained by:

$$\alpha_{ij} = \beta_{ij}h_{ij} + \eta_{ij}. \tag{20}$$

Thus, we can first solve the above dual problem, and then obtain the matrix $\mathbf{M}$ by

$$\mathbf{M} = \sum\nolimits_{i,j}\left(\beta_{ij}h_{ij}+\eta_{ij}\right)\mathbf{X}_{ij}, \tag{21}$$

and the distance threshold b can be obtained by (55) in **Appendix B**.

B. Optimization Algorithm

There are two groups of variables, $\boldsymbol{\eta}$ and $\boldsymbol{\beta}$, in problem (19). We adopt an alternating minimization approach to solve them. First, given $\boldsymbol{\eta}$, the variables β_{ij} can be obtained by:

$$\begin{aligned}\max_{\boldsymbol{\beta}} \ & -\frac{1}{2}\sum_{i,j,k,l}\beta_{ij}\beta_{kl}h_{ij}h_{kl}K_{ijkl} + \sum_{i,j}\delta_{ij}\beta_{ij}\\ \text{s.t. } & 0\leq\beta_{ij}\leq C, \quad \forall i, \ j, \ \sum_{i,j}\beta_{ij}h_{ij} = 0, \end{aligned} \tag{22}$$

where $\boldsymbol{\delta}$ is the variable with $\delta_{ij} = \left(1 - h_{ij}\sum_{kl}\eta_{kl}\left\langle\mathbf{X}_{ij},\mathbf{X}_{kl}\right\rangle\right)$. Clearly, the subproblem on $\boldsymbol{\beta}$ is similar to the dual of SVM, and it can be solved by LibSVM [20].

Given $\boldsymbol{\beta}$, the subproblem on $\boldsymbol{\eta}$ can be formulated as follows:

$$\begin{aligned}\min_{\boldsymbol{\eta}} \ & \frac{1}{2}\sum\nolimits_{i,j}\sum\nolimits_{k,l}\eta_{ij}\eta_{kl}K_{ijkl} + \sum\nolimits_{i,j}\eta_{ij}\gamma_{ij}\\ \text{s.t. } & \sum\nolimits_{k,l}\eta_{ij}K_{ijkl}\geq 0, \quad \forall i,j, \end{aligned} \tag{23}$$

where $\gamma_{ij} = \sum_{kl}\beta_{kl}h_{kl}\left\langle\mathbf{X}_{ij},\mathbf{X}_{kl}\right\rangle$. To simplify the subproblem on $\boldsymbol{\eta}$, we derive its Lagrange dual based on the KKT condition:

$$\eta_{ij} = \mu_{ij} - h_{ij}\beta_{ij}, \quad \forall i,j, \tag{24}$$

where $\boldsymbol{\mu}$ is the Lagrange dual multiplier. The Lagrange dual problem of (23) is formulated as follows:

$$\begin{aligned}\max_{\boldsymbol{\mu}} \ & -\frac{1}{2}\sum\nolimits_{i,j}\sum\nolimits_{k,l}\mu_{ij}\mu_{kl}K_{ijkl} + \sum\nolimits_{i,j}\gamma_{ij}\mu_{ij}\\ \text{s.t. } & \mu_{ij}\geq 0, \quad \forall i,j. \end{aligned} \tag{25}$$

Please refer to **Appendix C** for the detailed derivation. Clearly, problem (25) is more simple and can be efficiently solved by the SVM solvers.

Algorithm 2 Algorithm of NCML

Input: Training set $\{(\mathbf{x}_i, \mathbf{x}_j), h_{ij}\}$.
Output: The matrix $\mathbf{M}$.
Initialize $\boldsymbol{\eta}^{(0)}$ with small random values, $t \leftarrow 0$.
repeat
1. Update $\boldsymbol{\delta}^{(t+1)}$ with $\boldsymbol{\delta}_{ij}^{(t+1)} = \left(1 - h_{ij} \sum_{kl} \eta_{kl}^{(t)} K_{ijkl}\right)$.
2. Update $\boldsymbol{\beta}^{(t+1)}$ by solving the subproblem (22) using an SVM solver.
3. Update $\boldsymbol{\gamma}^{(t+1)}$ with $\gamma_{ij}^{(t+1)} = \sum_{kl} \beta_{kl}^{(t+1)} h_{kl} K_{ijkl}$.
4. Update $\boldsymbol{\mu}^{(t+1)}$ by solving the subproblem (25) using an SVM solver.
5. Update $\boldsymbol{\eta}^{(t+1)}$ with $\eta_{ij}^{(t+1)} \leftarrow \mu_{ij}^{(t+1)} - h_{ij}\beta_{ij}^{(t+1)}$.
6. $t \leftarrow t + 1$.

until convergence
$\mathbf{M} = \sum_{ij} \mu_{ij}^{(t)} \mathbf{X}_{ij}$.
return $\mathbf{M}$

After obtaining $\boldsymbol{\mu}$ and $\boldsymbol{\beta}$, the solution of $\boldsymbol{\alpha}$ in problem (18) can be obtained by

$$\alpha_{ij} = \mu_{ij}, \quad \forall i, j. \tag{26}$$

We then have $\mathbf{M} = \sum_{ij} \alpha_{ij} \mathbf{X}_{ij}$. The NCML algorithm is summarized in **Algorithm 2**.

C. Optimality Condition

Our NCML training algorithm can reach global optimum. From (18) and (19), the duality gap in the nth iteration is

$$\begin{aligned}\text{DualGap}_{\text{NCML}}^{(n)} &= \frac{1}{2} \sum_{i,j,k,l} \alpha_{ij}^{(n)} \alpha_{kl}^{(n)} K_{ijkl} \\ &\quad + \frac{1}{2} \sum_{i,j,k,l} (\beta_{ij}^{(n)} h_{ij} + \eta_{ij}^{(n)})(\beta_{kl}^{(n)} h_{kl} + \eta_{kl}^{(n)}) K_{ijkl} \\ &\quad - \sum_{i,j} \beta_{ij}^{(n)} + C \sum_{i,j} \xi_{ij}^{(n)},\end{aligned} \tag{27}$$

where $\alpha_{ij}^{(n)}$ and $\xi_{ij}^{(n)}$ are the feasible solutions to the primal problem, $\beta_{ij}^{(n)}$ and $\eta_{ij}^{(n)}$ are the feasible solutions to the dual problem, and $\text{DualGap}_{\text{NCML}}^{(n)}$ is the duality gap in the nth iteration. As $\eta_{ij}^{(n)}$ and $\mu_{ij}^{(n)}$ are the optimal solutions to the primal subproblem on $\boldsymbol{\eta}$ in (23) and its dual problem in (25), respectively, the duality gap of subproblem on $\boldsymbol{\eta}$ is zero, i.e.,

$$\begin{aligned}&\frac{1}{2} \sum_{i,j,k,l} \eta_{ij}^{(n)} \eta_{kl}^{(n)} K_{ijkl} + \sum_{i,j} \eta_{ij}^{(n)} \gamma_{ij}^{(n)} + \frac{1}{2} \sum_{i,j,k,l} \mu_{ij}^{(n)} \mu_{kl}^{(n)} K_{ijkl} \\ &\quad - \sum_{i,j} \gamma_{ij}^{(n)} \mu_{ij}^{(n)} + \frac{1}{2} \sum_{i,j,k,l} \beta_{ij}^{(n)} \beta_{kl}^{(n)} h_{ij} h_{kl} K_{ijkl} = 0.\end{aligned} \tag{28}$$

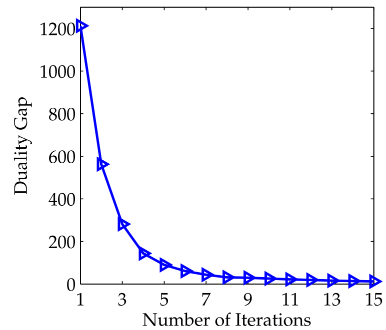


Fig. 3. Duality gap vs. number of iterations on the *PenDigits* database for NCML.

As shown in (26), $\alpha_{ij}^{(n)}$ and $\mu_{ij}^{(n)}$ should be equal. We substitute (28) into (27) as follows:

$$\text{DualGap}_{\text{NCML}}^{(n)} = C \sum_{i,j} \xi_{ij}^{(n)} - \sum_{i,j} \beta_{ij}^{(n)} + \sum_{i,j} \mu_{ij}^{(n)} \gamma_{ij}^{(n)}. \tag{29}$$

Based on the KKT conditions of the NCML dual problem in (19), $\xi_{ij}^{(n)}$ can be obtained by (30), as shown at the bottom of this page, where $[z] = \max(z, 0)$ and $b^{(n)}$ can be obtained by

$$\begin{aligned}b^{(n)} &= \sum_{k,l} \alpha_{kl}^{(n)} K_{ijkl} - 1/h_{ij} \\ &= \gamma_{ij}^{(n)} - \delta_{ij}^{(n+1)}/h_{ij} \quad \text{for all } 0 < \beta_{ij}^{(n)} < C.\end{aligned} \tag{31}$$

Please refer to **Appendix B** for the derivation of $\xi_{ij}^{(n)}$ and $b^{(n)}$.

Fig. 3 plots the curve of duality gap versus the number of iterations on *PenDigits* by NCML. The duality gap is nearly zero within 10∼15 iterations, and NCML reaches the global optimum. In the implementation of **Algorithm 2**, we adopt the following termination condition:

$$\text{DualGap}_{\text{NCML}}^{(t)} < \varepsilon \cdot \text{DualGap}_{\text{NCML}}^{(1)}, \tag{32}$$

where ε is a small constant and we set $\varepsilon = 0.01$.

D. Remarks

1) Computational Complexity: We use the same strategy as that in PCML to construct the pairwise constraints. In each iteration, NCML calls for the SVM solver twice while PCML calls for it only once. When the SMO-type algorithm [48] is adopted for SVM training, the computational complexity of NCML is $O\left(k^2 N^2 d\right)$. One extra advantage of NCML lies in its lower computational cost with respect to d, which involves the computation of $K\left((\mathbf{x}_i, \mathbf{x}_j), (\mathbf{x}_k, \mathbf{x}_l)\right)$ and the construction of matrix $\mathbf{M}$. Since $K\left((\mathbf{x}_i, \mathbf{x}_j), (\mathbf{x}_k, \mathbf{x}_l)\right) = \left((\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_k - \mathbf{x}_l)\right)^2$, the cost of kernel computation is $O(d)$. The cost of constructing the matrix $\mathbf{M}$ is less than $O\left(kNd^2\right)$, and this operation is required only once after obtaining $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$.

$$\xi_{ij}^{(n)} = \begin{cases} 0 & \text{for all } \beta_{ij}^{(n)} < C \\ \left[1 - h_{ij}\left(\sum_{k,l} \alpha_{kl}^{(n)} K_{ijkl} - b^{(n)}\right)\right]_+ = \left[\delta_{ij}^{(n+1)} - h_{ij}\left(\gamma_{ij}^{(n)} - b^{(n)}\right)\right]_+ & \text{for all } \beta_{ij}^{(n)} = C. \end{cases} \tag{30}$$

2) Difference With Doublet-SVM [25]: Our PCML and NCML are related but distinctly different with doublet-SVM [25]. Like doublet-SVM, our PCML and NCML also cast metric learning as kernel classification problems. However, in doublet-SVM, $\mathbf{M}$ is first learned by ignoring the PSD constraint to exploit SVM solver and then projected onto the PSD cone. Thus, doublet-SVM is only a heuristic method and cannot obtain the global solution. In contrast, our PCML iterates between SVM training and PSD projection to learn $\mathbf{M}$, and our NCML iterates between two SVMs to learn $\mathbf{M}$. They provide a principled scheme to exploit SVM solver for metric learning. As analyzed in Sec. III.C and IV.C, our algorithms can ensure the global optimality of $\mathbf{M}$. Moreover, by initializing $\mathbf{Y}$ with $\mathbf{0}$, doublet-SVM actually is a special case of PCML with one iteration.

V. Experimental Results

We evaluate our PCML and NCML methods for k-NN classification ($k = 1$) on general classification, face verification, and person re-identification. We compare PCML and NCML with the baseline Euclidean distance metric and 8 state-of-the-art metric learning models, including NCA [7], ITML [14], MCML [11], LDML [4], LMNN [2], PLML [26], DML-eig [13] and Doublet-SVM [25]. On each database, if the partition of training set and test set is not defined, we evaluate the performance of each method by 10-fold cross-validation, and the classification error rate and training time are obtained by averaging over 10 runs of 10-fold cross-validation (CV). PCML and NCML are implemented using the LibSVM[1] toolbox, and our codes are online available.[2] The source codes of NCA,[3] ITML,[4] MCML,[5] LDML,[6] LMNN,[7] PLML,[8] and DML-eig[9] are online available, and we tune their parameters to get the best results.

A. Evaluation on General Classification Tasks

We use 9 databases from the UCI Machine Learning Repository [49] and 4 handwritten digit databases to evaluate our methods. Table II and Table IV provides a summary of these databases. On the *Satellite*, *SPECTF Heart*, *Letter*, *MNIST*, *PenDigits*, and *USPS* databases, the training set and test set are defined. On the other databases, we use 10-fold CV to evaluate the metric learning models, and the classification error rate and training time are obtained by averaging over 10 runs of 10-fold cross-validation. As the dimensions of images in the *MNIST*, *Semeion* and *USPS* databases are relatively high, we use principal component analysis (PCA) to reduce the feature dimension to 100, and train the metrics in the PCA subspace.

[1]http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[2]https://github.com/csfwang/ISVM

[3]http://www.cs.berkeley.edu/~fowlkes/software/nca/

[4]http://www.cs.utexas.edu/~pjain/itml/

[5]http://homepage.tudelft.nl/19j49/Matlab_Toolbox_for_Dimensionality_Reduction.html

[6]http://lear.inrialpes.fr/people/guillaumin/code.php

[7]http://www.cse.wustl.edu/~kilian/code/code.html

[8]http://cui.unige.ch/~wangjun/

[9]http://empslocal.ex.ac.uk/people/staff/yy267/software.html

TABLE II
The UCI Databases Used in Our Experiments

Database	# of training samples	# of test samples	Feature dimension	# of classes
Breast Tissue	96	10	9	6
Cardiotocography	1,914	212	21	10
ILPD	525	58	10	2
Letter	16,000	4,000	16	26
Parkinsons	176	19	22	2
Satellite	4,435	2,000	36	6
Segmentation	2,079	231	19	7
Sonar	188	20	60	2
SPECTF Heart	80	187	44	2

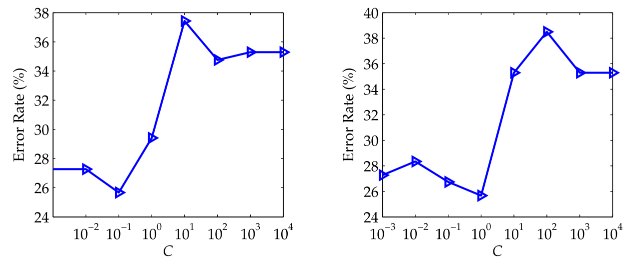


Fig. 4. Classification error rate (%) versus C. (a) PCML; (b) NCML.

Our PCML and NCML involve only one hyper-parameter, i.e., the regularization parameter C. We simply adopt the cross-validation strategy to select C by investigating the influence of C on the classification error rate. Fig. 4 shows the curves of classification error rate versus C for PCML and NCML on the *SPECTF Heart* database. The curves on other databases are similar. We can observe that when $C < 1$, the classification error rates of PCML and NCML will be low and stable. When C is higher than 1, the classification error rates jump dramatically. Thus, we set $C < 1$ in our experiments.

We compare the classification error rates of the competing methods in Tables III and V. On *Cardiotocography*, *Segmentation*, *PenDigits* and *Semeion*, PCML achieves the lowest error rates. On *Segmentation*, *SPECTF Heart* and *PenDigits*, NCML achieves the lowest error rates. According to [50], the average rank can provide a fair comparison of classification methods. Therefore, we provide the average ranks of competing methods in the last rows of Tables III and V. We do not report the error rate and training time of MCML on *MNIST* because MCML requires too large memory space (more than 30 GB) on this database and cannot run in our PC. From Tables III and V, PCML and NCML achieve the first and third best average ranks on the UCI databases, and achieve the best average ranks on the handwritten digit databases, demonstrating their effectiveness for general classification tasks.

We compare the training time of competing methods in Figs. 5 and 6. All the experiments are run in a PC with 4 Intel Core i5-2410 CPUs (2.30 GHz) and 16GB RAM. Clearly, doublet-SVM [25] and the proposed PCML and NCML are

TABLE III
CLASSIFICATION ERROR RATE (%) ON THE UCI DATABASES

Database	Euclidean	NCA	ITML	MCML	LDML	LMNN	PLML	DML-eig	Doublet-SVM	PCML	NCML
Breast Tissue	**31.00**	41.27	35.82	32.09	48.00	34.37	35.18	34.13	38.37	38.00	35.37
Cardiotocography	21.40	21.16	18.67	22.29	22.26	19.21	18.54	29.31	18.67	**18.50**	18.69
ILPD	35.69	34.65	35.35	35.49	35.84	34.12	**31.61**	36.87	34.82	33.96	32.43
Letter	4.33	**2.47**	3.80	4.20	11.05	3.45	3.28	3.85	2.67	2.67	2.72
Parkinsons	**4.08**	6.63	6.13	9.84	7.15	5.26	8.84	7.82	5.63	5.68	7.26
Satellite	10.95	10.40	11.45	15.65	15.90	**10.05**	11.85	10.90	11.05	11.15	11.10
Segmentation	2.86	2.51	2.73	2.60	2.86	2.64	2.68	2.97	2.42	**2.12**	**2.12**
Sonar	12.98	15.40	12.07	24.29	22.86	**11.57**	12.07	15.07	14.00	12.71	13.29
SPECTF Heart	38.50	26.74	34.76	38.50	33.16	34.76	27.27	31.02	27.27	28.88	**25.67**
Average Rank	*6.33*	*5.11*	*6.11*	*8.44*	*9.44*	*4.22*	*4.89*	*7.78*	*4.67*	***3.89***	*4.33*

TABLE IV
THE HANDWRITTEN DIGIT DATABASES USED IN THE EXPERIMENTS

Database	# of training samples	# of test samples	dimension	PCA dimension	# of classes
MNIST	60,000	10,000	784	100	10
PenDigits	7,494	3,498	16	N/A	10
Semeion	1,434	159	256	100	10
USPS	7,291	2,007	256	100	10

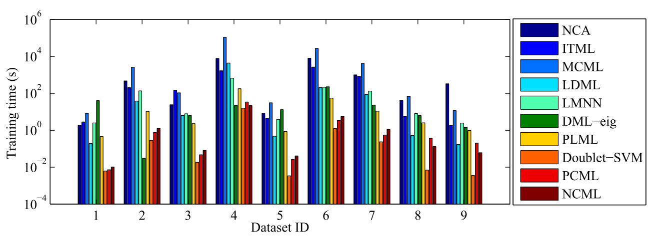


Fig. 5. Training time (s) of NCA, ITML, MCML, LDML, LMNN, DML-eig, PLML, Doublet-SVM, PCML and NCML. From 1 to 9, the Database ID represents *Breast Tissue*, *Cardiotocography*, *ILPD*, *Letter*, *Parkinsons*, *Satellite*, *Segmentation*, *Sonar* and *SPECTF Heart*.

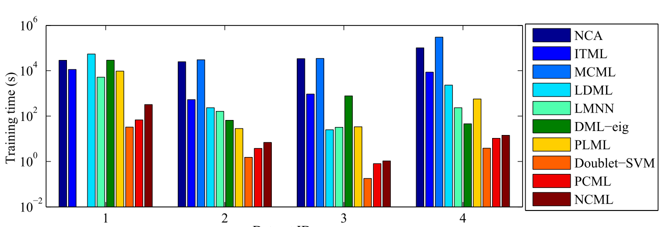


Fig. 6. Training time (s) of NCA, ITML, MCML, LDML, LMNN, DML-eig, PLML, Doublet-SVM, PCML and NCML. From 1 to 4, the Database ID represents *MNIST*, *PenDigits*, *Semeion* and *USPS*.

the fastest in most cases. Although DML-eig is faster than PCML on *Letter*, its classification error rate on this database is much higher than PCML and NCML. On average, PCML and NCML are 35 and 21 times faster than PLML, the fourth fastest algorithm, respectively.

Finally, we compare the running time of PCML and NCML under different feature dimensions d. Fig. 7 shows the training

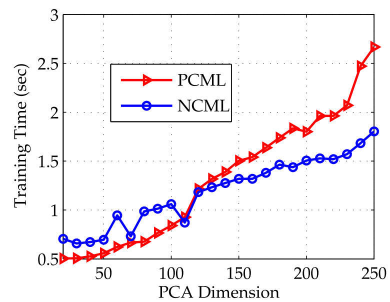


Fig. 7. Training time (s) vs. PCA dimension on the *Semeion* database.

time on *Semeion* with different PCA dimensions. When the dimension is lower than 110, the training time of NCML is longer than PCML. When the dimension is higher than 110, the training time of PCML increases and becomes longer than NCML. The results are consistent with the complexity analysis given in Subsections III.D and IV.D.

1) Discussion: In this subsection, we give a brief discussion on the training efficiency and accuracy of PCML and NCML.

1) Training efficiency: Albeit lower in terms of computational complexity, NCML requires to run the SVM solver twice per iteration while PCML only once. Besides, the number of iterations may also be different for NCML and PCML. As shown in Fig. 7, when the feature dimension is lower, PCML is more efficient in training. As in most of our experiments, the dimensions of training samples are relatively low, making NCML less efficient than PCML.
2) Accuracy: From Theorem 1, the feasible domain of NCML is a subset of that of PCML. Thus, with sufficient training data, PCML has the opportunity to find distance metric in a larger searching domain. But in many practical problems, the training data generally are insufficient. Thus, the restriction of feasible domain by NCML may serve as some kind of regularization on the solution, and sometimes may even benefit classification performance.

B. Face Verification

We evaluate the proposed methods for face verification using the Labeled Faces in the Wild (LFW) [51] database.

TABLE V
COMPARISON OF CLASSIFICATION ERROR RATE (%) ON THE HANDWRITTEN DIGIT DATABASES

Database	Euclidean	NCA	ITML	MCML	LDML	LMNN	DML-eig	PLML	Doublet-SVM	PCML	NCML
MNIST	2.87	3.75	2.89	N/A	6.05	**2.28**	5.06	2.54	3.19	3.85	2.80
PenDigits	2.26	2.23	2.29	2.26	6.20	2.23	3.75	2.46	**2.06**	**2.06**	**2.06**
Semeion	8.54	8.60	5.71	11.23	11.98	6.09	5.72	7.66	5.21	**4.83**	5.53
USPS	**5.08**	5.68	6.33	**5.08**	8.77	5.38	5.43	6.73	5.43	5.33	5.43
Average Rank	*4.75*	*7.00*	*6.50*	*7.00*	*10.75*	*3.75*	*7.25*	*7.00*	*4.50*	***3.25***	***3.00***

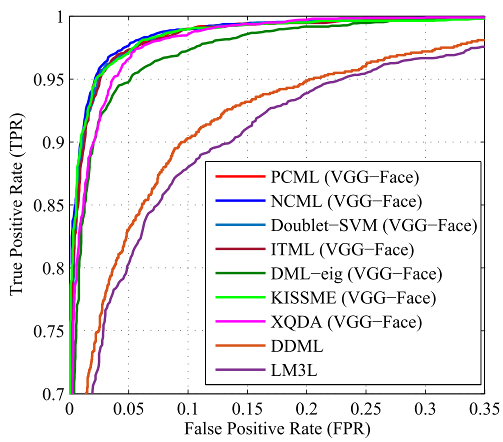


Fig. 8. The ROC curves of different methods on the LFW database.

TABLE VI
VERIFICATION ACCURACIES (%) AND TRAINING TIME (s) OF COMPETING METHODS ON THE LFW-FUNNELED DATABASE

Methods	Verification Accuracy (%)	Training Time (s)
PCML (VGG-Face)	96.43	9.81
NCML (VGG-Face)	**96.63**	10.16
Doublet-SVM [25] (VGG-Face)	96.40	0.43
ITML [14] (VGG-Face)	96.40	194.92
DML-eig [13] (VGG-Face)	94.90	256.24
KISSME [5] (VGG-Face)	96.33	0.01
XQDA [30] (VGG-Face)	95.67	0.02
DDML [27]	90.68	-
TSML [28]	89.80	-
LM3L [29]	89.57	-

The face images in LFW were collected from the Internet and demonstrate large variations of pose, illumination, expression, etc. The database consists of 13,233 face images from 5,749 persons. Under the image restricted setting, the performance of a face verification method is evaluated by 10-fold CV. For each of the 10 runs, the database provides 300 positive pairs and 300 negative pairs for testing, and 5,400 image pairs for training. The verification rate and Receiver Operator Characteristic (ROC) curve of each method are obtained by averaging over the 10 runs.

In our experiments, we use the VGG-Face [52] feature to evaluate the face verification methods. Since the dimension of VGG-Face feature is high (i.e., 4096), PCA is used to reduce the feature dimension to 50. We transform each feature vector $\mathbf{x}$ by $\tilde{\mathbf{x}} = \mathbf{L}_{\mathcal{S}}^{-1}\mathbf{x}$, where $\mathbf{L}_{\mathcal{S}}\mathbf{L}_{\mathcal{S}}^{T} = \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in\mathcal{S}}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$ [53]. Under the restricted setting, we only know whether two images are matched or not for the given pairs. In the training stage, we use the training pairs to train a Mahalanobis distance metric. In the test stage, we compare the distance of the test pair with the distance threshold to decide whether the two images are matched or not.

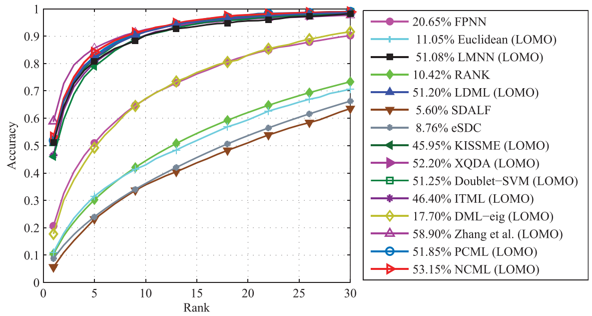
We report the ROC curves and verification accuracies of PCML, NCML, Doublet-SVM [25], ITML [14], DML-eig [13], KISSME [5], XQDA [30], DDML [27], TSML [28][10] and LM3L [29] in Fig. 8 and Table VI. It can be seen that our proposed PCML and NCML methods can achieve satisfactory verification accuracies which are higher or comparable to the competing methods. The training time of PCML and NCML are much shorter than ITML [14] and DML-eig [13], but are longer than Doublet-SVM [25], KISSME [5] and XQDA [30]. We note that Doublet-SVM [25] is a two-stage method and KISSME is a one-pass optimization method. And they cannot guarantee to obtain the global optimum of the model. XQDA [30] is a subspace method, and its closed-form solution can be obtained by eigenvalue decomposition. In contrast, our proposed PCML and NCML methods solve a convex SDP problem and are able to reach the global optimum.

[10]As the ROC curve of TSML [28] hasn't been released, we haven't reported it in this paper.

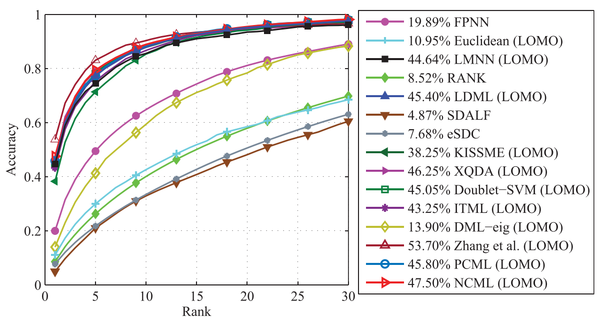
C. Person Re-Identification

In this subsection, we evaluate the performance of our methods for person re-identification, *i.e.*, recognizing a person by the pedestrian image at different locations and at different times [54]. We use the CUHK03 [55] and CUHK01 [56] databases to assess the performance of our methods.

1) CUHK03: CUHK03 database contains 14,096 pedestrian images which are taken from 1,467 persons by two cameras [55]. We randomly select 1,367 persons and use their images as the training set, and use the images from the rest 100 persons as the test set. For each person in the test set, we randomly select the images taken by one camera as the probe images, and use one of the images taken by another camera as the gallery image. 20 partitions of training set and test sets are constructed, and the reported accuracies are averaged over all the partitions. We report the CMC curves, rank-1 accuracies



(a)



(b)

Fig. 9. The CMC curves of different methods on the CUHK03 database with (a) manually labeled bounding box and (b) automatically detected bounding box.

and training time of PCML, NCML and the competing methods, *i.e.* ITML [14], DML-eig [13], LMNN [2], RANK [57], LDML [4], symmetry-driven accumulation of local features (SDALF) [58], eSDC [59], KISSME [5], XQDA [30], filter pairing neural network (FPNN) [55], Doublet-SVM [25] and Zhang *et al.* [44], on CUHK03 database with manually labeled and detected bounding boxes on single-shot setting in Fig. 9 and Table VII. For FPNN [55], RANK [57], SDALF [58] and eSDC [59], we use the results in their original papers. As to the other methods, the results are obtained by using an effective feature representation named Local Maximal Occurrence (LOMO) [30]. One can see that the rank-1 accuracies of PCML and NCML are much higher than most of the competing methods, comparable to XQDA [30], but lower than Zhang *et al.* [44]. Note that Zhang *et al.* [44] learn nonlinear discriminative null space via kernelization, while what the other methods learned are Mahalanobis distance metric or linear subspace. And this might explain the superiority of Zhang *et al.* [44] over the other methods. Analogous to the results on LFW, the training time of PCML and NCML are much shorter than ITML [14], XQDA [30], Zhang *et al.* [44], LDML [4] and LMNN [2], comparable to DML-eig [13], and longer than Doublet-SVM [25] and KISSME [5].

2) CUHK01: CUHK01 database consists of 3,884 images taken from 971 persons [56]. Each person has 4 images taken by two cameras. The training set contains the images from 485 persons, which are randomly selected from this database. The images from the other 486 persons construct the test set. The reported results are based on multi-shot setting. The CMC curves, rank-1 accuracies and training time of PCML, NCML,

TABLE VII

TRAINING TIME (s) ON THE CUHK03 DATABASE WITH LOMO FEATURE [30]

Methods	Training Time (s)
PCML	576.45
NCML	655.77
Doublet-SVM [25]	227.54
ITML [14]	1228.50
DML-eig [13]	523.33
KISSME [5]	0.85
XQDA [30]	902.35
Zhang *et al.* [44]	1954.60
LDML [4]	794.38
LMNN [2]	8383.60

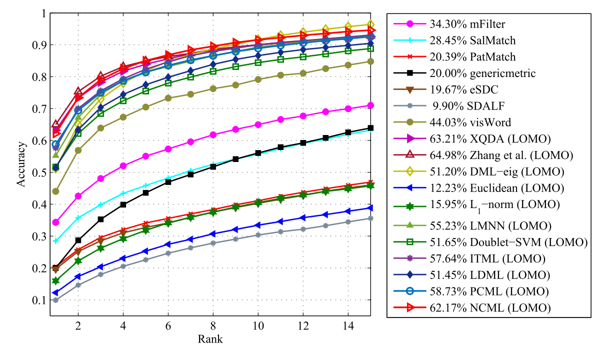


Fig. 10. The CMC curves of different methods on the CUHK01 database.

TABLE VIII

TRAINING TIME (s) ON THE CUHK01 DATABASE WITH LOMO FEATURE [30]

Methods	Training Time (s)
PCML	33.71
NCML	59.61
Doublet-SVM [25]	23.02
ITML [14]	4694.10
DML-eig [13]	3501.50
KISSME [5]	0.59
XQDA [30]	16.47
Zhang *et al.* [44]	4.12
LDML [4]	64.03
LMNN [2]	2665.60

ITML [14], DML-eig [13], Doublet-SVM [25], XQDA [30], Zhang *et al.* [44] and other competing methods [2], [4], [5], [55], [57]–[59] are reported in Fig. 10 and Table VIII, respectively. Except mFilter [60], SalMatch [61], PatMatch [61], generic metric [56], eSDC [59], SDALF [58], and visWord [62], the results of the other methods are obtained by using the LOMO feature [30]. We can see that the rank-1 accuracies of PCML and NCML are also higher than most of the other methods, comparable to XQDA [30], and lower than Zhang *et al.* [44]. For person re-identification, subspace methods such as XQDA [30] and Zhang *et al.* [44] are also very effective and generally outperform most competing methods.

VI. Conclusion

We proposed two distance metric learning models, namely PCML and NCML. The proposed models can guarantee the positive semidefinite property of the learned matrix $\mathbf{M}$, and can be solved efficiently by the existing SVM solvers. Experimental results on general classification tasks showed that, compared with the state-of-the-art metric learning methods, including NCA [7], ITML [14], MCML [11], LDML [4], LMNN [2], PLML [26], DML-eig [13] and Doublet-SVM [25], the proposed PCML and NCML methods can not only achieve favorable classification accuracy, but also are efficient in training. The experimental results on LFW, CUHK01 and CUHK03 databases indicate that the proposed methods also perform well in face verification and person re-identification. For face verification, PCML and NCML achieve higher or comparable accuracies to the competing methods on the LFW database. For person re-identification, our PCML and NCML can obtain better or comparable accuracy to most Mahalanobis distance metric learning or linear subspace methods, but are inferior to the kernelized subspace method by Zhang *et al.* [44].

Appendix A
The Dual of PCML

The original problem of PCML is formulated as

$$\begin{aligned}&\min_{\mathbf{M},b,\boldsymbol{\xi}} \frac{1}{2}\|\mathbf{M}\|_F^2 + C\sum\nolimits_{i,j}\xi_{ij}\\ &\text{s.t. } h_{ij}\left(\langle\mathbf{M},\mathbf{X}_{ij}\rangle - b\right) \ge 1-\xi_{ij}, \xi_{ij}\ge 0, \quad \forall i,j\\ &\quad \mathbf{M}\succcurlyeq 0. \end{aligned} \tag{33}$$

Its Lagrangian is:

$$\begin{aligned}L(\boldsymbol{\lambda},\boldsymbol{\kappa},\mathbf{Y},\mathbf{M},b,\boldsymbol{\xi}) &= \frac{1}{2}\|\mathbf{M}\|_F^2 + C\sum\nolimits_{i,j}\xi_{ij}\\ &-\sum\nolimits_{i,j}\lambda_{ij}\left[h_{ij}\left(\langle\mathbf{M},\mathbf{X}_{ij}\rangle - b\right)-1+\xi_{ij}\right]\\ &-\sum\nolimits_{i,j}\kappa_{ij}\xi_{ij} - \langle\mathbf{Y},\mathbf{M}\rangle,\end{aligned} \tag{34}$$

where $\lambda_{ij}\ge 0$, $\kappa_{ij}\ge 0, \forall i,j$, and $\mathbf{Y}\succcurlyeq 0$ are the Lagrange multipliers. Converting the primal problem to its dual problem needs the following KKT conditions:

$$\frac{\partial L(\boldsymbol{\lambda},\boldsymbol{\kappa},\mathbf{Y},\mathbf{M},b,\boldsymbol{\xi})}{\partial\mathbf{M}} = \mathbf{0} \Rightarrow \mathbf{M} - \sum\nolimits_{i,j}\lambda_{ij}h_{ij}\mathbf{X}_{ij} - \mathbf{Y} = \mathbf{0}, \tag{35}$$

$$\frac{\partial L(\boldsymbol{\lambda},\boldsymbol{\kappa},\mathbf{Y},\mathbf{M},b,\boldsymbol{\xi})}{\partial b} = 0 \Rightarrow \sum\nolimits_{i,j}\lambda_{ij}h_{ij} = 0, \tag{36}$$

$$\frac{\partial L(\boldsymbol{\lambda},\boldsymbol{\kappa},\mathbf{Y},\mathbf{M},b,\boldsymbol{\xi})}{\partial \xi_{ij}} = C-\lambda_{ij}-\kappa_{ij} = 0 \Rightarrow 0\le\lambda_{ij}\le C, \quad \forall i,j, \tag{37}$$

$$h_{ij}\left(\langle\mathbf{M},\mathbf{X}_{ij}\rangle - b\right)-1+\xi_{ij}\ge 0,\ \ \xi_{ij}\ge 0, \tag{38}$$

$$\lambda_{ij}\ge 0,\ \ \kappa_{ij}\ge 0,\ \ \mathbf{Y}\succcurlyeq 0, \tag{39}$$

$$\lambda_{ij}\left[h_{ij}\left(\langle\mathbf{M},\mathbf{X}_{ij}\rangle - b\right)-1+\xi_{ij}\right] = 0,\ \ \kappa_{ij}\xi_{ij} = 0. \tag{40}$$

(35) implies the following relationship:

$$\mathbf{M} = \sum\nolimits_{i,j}\lambda_{ij}h_{ij}\mathbf{X}_{ij} + \mathbf{Y}. \tag{41}$$

Substituting (35)∼(37) back into the Lagrangian, we get the Lagrange dual problem of PCML:

$$\begin{aligned}&\max_{\lambda,\mathbf{Y}} -\frac{1}{2}\left\|\sum\nolimits_{i,j}\lambda_{ij}h_{ij}\mathbf{X}_{ij}+\mathbf{Y}\right\|_F^2 + \sum\nolimits_{i,j}\lambda_{ij}\\ &\text{s.t. } \sum\nolimits_{i,j}\lambda_{ij}h_{ij} = 0, 0\le\lambda_{ij}\le C, \quad \forall i,j, \quad \mathbf{Y}\succcurlyeq 0.\end{aligned} \tag{42}$$

From (41) and (42), $\mathbf{M}$ is explicitly determined by the training procedure, but b is not. Nevertheless, b can be found by using the KKT condition in (37) and (40), and we can take any training point, for which $0<\lambda_{ij}<C$, to compute b by

$$b = \langle\mathbf{M},\mathbf{X}_{ij}\rangle - 1/h_{ij}, \quad \text{for all } 0<\lambda_{ij}<C. \tag{43}$$

After b is computed, we can compute ξ_{ij} by

$$\xi_{ij} = \begin{cases} 0, & \text{for all } \lambda_{ij}<C\\ \left[1-h_{ij}\left(\langle\mathbf{M},\mathbf{X}_{ij}\rangle - b\right)\right]_+, & \text{for all } \lambda_{ij}=C,\end{cases} \tag{44}$$

where $[z]_+ = \max(z,0)$ denotes the hinge loss.

Appendix B
The Dual of NCML

The original problem of NCML is as follows:

$$\begin{aligned}&\min_{\boldsymbol{\alpha},b,\boldsymbol{\xi}} \frac{1}{2}\sum\nolimits_{i,j}\sum\nolimits_{k,l}\alpha_{ij}\alpha_{kl}K_{ijkl} + C\sum\nolimits_{i,j}\xi_{ij}\\ &\text{s.t. } h_{ij}\left(\sum\nolimits_{k,l}\alpha_{kl}K_{ijkl} - b\right)\ge 1-\xi_{ij}\\ &\quad \xi_{ij}\ge 0, \alpha_{ij}\ge 0, \quad \forall i,j.\end{aligned} \tag{45}$$

Its Lagrangian can be defined as:

$$\begin{aligned}&L(\boldsymbol{\beta},\boldsymbol{\sigma},\boldsymbol{\nu},\boldsymbol{\alpha},b,\boldsymbol{\xi})\\ &= \frac{1}{2}\sum\nolimits_{i,j,k,l}\alpha_{ij}\alpha_{kl}K_{ijkl} + C\sum\nolimits_{i,j}\xi_{ij}\\ &-\sum\nolimits_{i,j}\beta_{ij}\left[h_{ij}\left(\sum\nolimits_{kl}\alpha_{kl}K_{ijkl}-b\right)-1+\xi_{ij}\right]\\ &-\sum\nolimits_{i,j}\nu_{ij}\xi_{ij} - \sum\nolimits_{i,j}\sigma_{ij}\alpha_{ij},\end{aligned} \tag{46}$$

where $\beta_{ij}\ge 0$, $\sigma_{ij}\ge 0$ and $\nu_{ij}\ge 0$, $\forall i,j$ are the Lagrange multipliers. Converting the original problem to its dual problem needs the following KKT conditions:

$$\frac{\partial L(\boldsymbol{\beta},\boldsymbol{\sigma},\boldsymbol{\nu},\boldsymbol{\alpha},b,\boldsymbol{\xi})}{\partial\alpha_{ij}} = 0 \Rightarrow \sum\nolimits_{k,l}\alpha_{kl}K_{ijkl} - \sum\nolimits_{k,l}\beta_{kl}h_{kl}K_{ijkl} - \sigma_{ij} = 0, \tag{47}$$

$$\frac{\partial L(\boldsymbol{\beta},\boldsymbol{\sigma},\boldsymbol{\nu},\boldsymbol{\alpha},b,\boldsymbol{\xi})}{\partial b} = 0 \Rightarrow \sum\nolimits_{i,j}\beta_{ij}h_{ij} = 0, \tag{48}$$

$$\frac{\partial L(\boldsymbol{\beta},\boldsymbol{\sigma},\boldsymbol{\nu},\boldsymbol{\alpha},b,\boldsymbol{\xi})}{\partial\xi_{ij}} = 0 \Rightarrow C-\beta_{ij}-\nu_{ij} = 0 \Rightarrow 0\le\beta_{ij}\le C, \tag{49}$$

$$h_{ij}\left(\sum\nolimits_{k,l}\alpha_{kl}K_{ijkl}-b\right)-1+\xi_{ij}\ge 0,\quad \xi_{ij}\ge 0,\ \ \alpha_{ij}\ge 0, \quad \forall i,j, \tag{50}$$

$$\beta_{ij}\ge 0,\ \ \sigma_{ij}\ge 0,\ \ \nu_{ij}\ge 0, \quad \forall i,j, \tag{51}$$

$$\beta_{ij}\left[h_{ij}\left(\sum\nolimits_{k,l}\alpha_{kl}K_{ijkl}-b\right)-1+\xi_{ij}\right] = 0,\quad \nu_{ij}\xi_{ij} = 0,\ \ \sigma_{ij}\alpha_{ij} = 0, \quad \forall i,j. \tag{52}$$

Here we introduce a coefficient vector $\boldsymbol{\eta}$, which satisfies $\sigma_{ij} = \sum_{k,l} \eta_{kl} \langle \mathbf{X}_{ij}, \mathbf{X}_{kl} \rangle$. Note that $\langle \mathbf{X}_{ij}, \mathbf{X}_{kl} \rangle$ is a positive definite kernel. So we can guarantee that every $\boldsymbol{\eta}$ corresponds to a unique $\boldsymbol{\sigma}$, and vice versa. (47) implies the following relationship between $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\boldsymbol{\eta}$:

$$\alpha_{ij} = \beta_{ij} h_{ij} + \eta_{ij}, \quad \forall i, j. \tag{53}$$

Substituting (47)~(49) back into the Lagrangian, we get the Lagrange dual problem of NCML as follows:

$$\begin{aligned} \max_{\boldsymbol{\eta}, \boldsymbol{\beta}} \ & -\frac{1}{2} \sum_{i,j,k,l} \left(\beta_{ij} h_{ij} + \eta_{ij} \right) \left(\beta_{kl} h_{kl} + \eta_{kl} \right) K_{ijkl} \\ & + \sum_{i,j} \beta_{ij} \\ \text{s.t.} \ & \sum\nolimits_{k,l} \eta_{kl} K_{ijkl} \geq 0, 0 \leq \beta_{ij} \leq C, \quad \forall i, j \\ & \sum\nolimits_{i,j} \beta_{ij} h_{ij} = 0. \end{aligned} \tag{54}$$

Analogous to PCML, we can use the KKT condition in (49) and (52) to compute b and ξ_{ij} in NCML. (49) and (52) show that $\xi_{ij} = 0$ if $\beta_{ij} < C$, and $h_{ij} \left(\sum_{kl} \alpha_{kl} \langle \mathbf{X}_{ij}, \mathbf{X}_{kl} \rangle - b \right) - 1 + \xi_{ij} = 0$ if $\beta_{ij} > 0$. Thus we can simply take any training data point, for which $0 < \beta_{ij} < C$, to compute b by

$$b = \sum\nolimits_{k,l} \alpha_{kl} K_{ijkl} - 1/h_{ij}. \tag{55}$$

After obtain b, we can compute β_{ij} by

$$\xi_{ij} = \begin{cases} 0, & \forall \beta_{ij} < C \\ \left[1 - h_{ij} \left(\sum_{k,l} \alpha_{kl} K_{ijkl} - b \right) \right]_+, & \forall \beta_{ij} = C. \end{cases} \tag{56}$$

APPENDIX C
THE DUAL OF THE SUBPROBLEM ON $\boldsymbol{\eta}$ IN NCML

The subproblem on $\boldsymbol{\eta}$ is formulated as follows:

$$\begin{aligned} \min_{\boldsymbol{\eta}} \ & \frac{1}{2} \sum\nolimits_{i,j} \sum\nolimits_{k,l} \eta_{ij} \eta_{kl} K_{ijkl} + \sum\nolimits_{i,j} \eta_{ij} \gamma_{ij} \\ \text{s.t.} \ & \sum\nolimits_{k,l} \eta_{kl} K_{ijkl} \geq 0, \quad \forall i, j, \end{aligned} \tag{57}$$

where $\gamma_{ij} = \sum_{k,l} \beta_{kl} h_{kl} K_{ijkl}$. Its Lagrangian is:

$$\begin{aligned} L(\boldsymbol{\mu}, \boldsymbol{\eta}) = & \frac{1}{2} \sum\nolimits_{i,j} \sum\nolimits_{k,l} \eta_{ij} \eta_{kl} K_{ijkl} \\ & + \sum\nolimits_{i,j} \eta_{ij} \gamma_{ij} - \sum\nolimits_{i,j} \mu_{ij} \sum\nolimits_{k,l} \eta_{kl} K_{ijkl}, \end{aligned} \tag{58}$$

where $\boldsymbol{\mu}$ is the Lagrange multiplier which satisfies $\mu_{ij} \geq 0$, $\forall i, j$. Converting the original problem to its dual problem needs the following KKT condition:

$$\begin{aligned} & \frac{\partial L(\boldsymbol{\mu}, \boldsymbol{\eta})}{\partial \eta_{ij}} = 0 \\ & \Rightarrow \sum\nolimits_{k,l} \eta_{kl} K_{ijkl} + \gamma_{ij} - \sum\nolimits_{k,l} \mu_{kl} K_{ijkl} = 0. \end{aligned} \tag{59}$$

(59) implies the following relationship between $\boldsymbol{\mu}$, $\boldsymbol{\eta}$ and $\boldsymbol{\beta}$:

$$\eta_{ij} = \mu_{ij} - h_{ij} \beta_{ij}, \quad \forall i, j. \tag{60}$$

Substituting (59) and (60) back into the Lagrangian, we get the following Lagrange dual problem of the subproblem on $\boldsymbol{\eta}$:

$$\begin{aligned} \max_{\boldsymbol{\mu}} \ & -\frac{1}{2} \sum\nolimits_{i,j} \sum\nolimits_{k,l} \mu_{ij} \mu_{kl} K_{ijkl} + \sum\nolimits_{i,j} \gamma_{ij} \mu_{ij} \\ & -\frac{1}{2} \sum\nolimits_{i,j} \sum\nolimits_{k,l} \beta_{ij} \beta_{kl} h_{ij} h_{kl} K_{ijkl} \\ \text{s.t.} \ & \mu_{ij} \geq 0, \quad \forall i, j. \end{aligned} \tag{61}$$

Since $\boldsymbol{\beta}$ is fixed in this subproblem, $\sum_{i,j} \sum_{k,l} \beta_{ij} \beta_{kl} h_{ij} h_{kl} K_{ijkl}$ remains constant in (61). Thus we can omit this term and have the following simplified Lagrange dual problem:

$$\begin{aligned} \max_{\boldsymbol{\mu}} \ & -\frac{1}{2} \sum\nolimits_{i,j} \sum\nolimits_{k,l} \mu_{ij} \mu_{kl} K_{ijkl} + \sum\nolimits_{i,j} \gamma_{ij} \mu_{ij} \\ \text{s.t.} \ & \mu_{ij} \geq 0, \quad \forall i, j. \end{aligned} \tag{62}$$

REFERENCES

[1] A. Bellet, A. Habrard, and M. Sebban. (Jun. 2013). "A survey on metric learning for feature vectors and structured data." [Online]. Available: https://arxiv.org/abs/1306.6709

[2] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *J. Mach. Learn. Res.*, vol. 10, pp. 207–244, Feb. 2009.

[3] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell, "Distance metric learning, with application to clustering with side-information," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2002, pp. 521–528.

[4] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? Metric learning approaches for face identification," in *Proc. IEEE 12th Int. Conf. Comput. Vis. (ICCV)*, Sep./Oct. 2009, pp. 498–505.

[5] M. Köstinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 2288–2295.

[6] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, "Learning locally-adaptive decision functions for person verification," in *Proc. 16th IEEE Int. Conf. Comput. Vis. (ICCV)*, Jun. 2013, pp. 3610–3617.

[7] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood components analysis," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2004, pp. 513–520.

[8] L. Torresani and K.-C. Lee, "Large margin component analysis," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2006, pp. 1385–1392.

[9] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, and J. Zhou, "Neighborhood repulsed metric learning for kinship verification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 2, pp. 331–345, Feb. 2014.

[10] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2005, pp. 1473–1480.

[11] A. Globerson and S. Roweis, "Metric learning by collapsing classes," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2005, pp. 451–458.

[12] C. Shen, J. Kim, L. Wang, and A. van den Hengel, "Positive semidefinite metric learning using boosting-like algorithms," *J. Mach. Learn. Res.*, vol. 13, pp. 1007–1036, Apr. 2012.

[13] Y. Ying and P. Li, "Distance metric learning with eigenvalue optimization," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 1–26, Jan. 2012.

[14] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. Dhillon, "Information-theoretic metric learning," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, 2007, pp. 209–216.

[15] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka, "Metric learning for large scale image classification: Generalizing to new classes at near-zero cost," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2012, pp. 488–501.

[16] G. Chechik, V. Sharma, U. Shalit, and S. Bengio, "Large scale online learning of image similarity through ranking," *J. Mach. Learn. Res.*, vol. 11, pp. 1109–1135, Jan. 2010.

[17] C. Shen, J. Kim, and L. Wang, "Scalable large-margin Mahalanobis distance metric learning," *IEEE Trans. Neural Netw.*, vol. 21, no. 9, pp. 1524–1530, Sep. 2010.

[18] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, pp. 2399–2434, Nov. 2006.

[19] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 577–584.

[20] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, 2011.

[21] I. W. Tsang, J. T. Kwok, and P.-M. Cheung, "Core vector machines: Fast SVM training on very large data sets," *J. Mach. Learn. Res.*, vol. 3, pp. 363–392, Apr. 2005.

[22] A. Bordes, L. Bottou, P. Gallinari, and J. Weston, "Solving multiclass support vector machines with LaRank," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, 2007, pp. 89–96.

[23] C. H. Teo, Q. V. Le, A. Smola, and S. V. N. Vishwanathan, "A scalable modular convex solver for regularized risk minimization," in *Proc. 13th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2007, pp. 727–736.

[24] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal estimated sub-gradient solver for SVM," *Math. Program.*, vol. 127, no. 1, pp. 3–30, Mar. 2011.

[25] 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.

[26] J. Wang, A. Woznica, and A. Kalousis, "Parametric local metric learning for nearest neighbor classification," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1601–1609.

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

[28] L. Zheng, K. Idrissi, C. Garcia, S. Duffner, and A. Baskurt, "Triangular similarity metric learning for face verification," in *Proc. 11th Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, May 2015, pp. 1–7.

[29] J. Hu, J. Lu, J. Yuan, and Y.-P. Tan, "Large margin multi-metric learning for face and kinship verification in the wild," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, 2014, pp. 252–267.

[30] S. Liao, Y. Hu, X. Zhu, and S. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2197–2206.

[31] K. Q. Weinberger and L. K. Saul, "Fast solvers and efficient implementations for distance metric learning," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 1160–1167.

[32] R. Jin, S. Wang, and Y. Zhou, "Regularized distance metric learning: Theory and algorithm," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 862–870.

[33] G.-J. Qi, J. Tang, Z.-J. Zha, T.-S. Chua, and H.-J. Zhang, "An efficient sparse metric learning in high-dimensional space via ℓ_1-penalized log-determinant regularization," in *Proc. 26th Int. Conf. Mach. Learn.*, 2009, pp. 841–848.

[34] Y. Ying, K. Huang, and C. Campbell, "Sparse metric learning via smooth optimization," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 2214–2222.

[35] C. Shen, J. Kim, L. Wang, and A. van den Hengel, "Positive semidefinite metric learning with boosting," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 1651–1659.

[36] J. Bi, D. Wu, L. Lu, M. Liu, Y. Tao, and M. Wolf, "Adaboost on low-rank PSD matrices for metric learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 2617–2624.

[37] C. Shen, J. Kim, and L. Wang, "A scalable dual approach to semidefinite metric learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 2601–2608.

[38] 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.

[39] M. Liu and B. C. Vemuri, "A robust and efficient doubly regularized metric learning approach," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2012, pp. 646–659.

[40] N. Nguyen and Y. Guo, "Metric learning: A support vector approach," in *Proc. ECML/PKDD*, 2008, pp. 125–136.

[41] C. Brunner, A. Fischer, K. Luig, and T. Thies, "Pairwise support vector machines and their application to large scale problems," *J. Mach. Learn. Res.*, vol. 13, pp. 2279–2292, Aug. 2012.

[42] H. Do, A. Kalousis, J. Wang, and A. Woznica. (Jan. 2012). "A metric learning perspective of SVM: On the relation of SVM and LMNN." [Online]. Available: https://arxiv.org/abs/1201.4714

[43] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *Proc. ICLR*, 2015, pp. 84–92.

[44] L. Zhang, T. Xiang, and S. Gong, "Learning a discriminative null space for person re-identification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1239–1248.

[45] 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.

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

[47] S. C. H. Hoi, W. Liu, and S.-F. Chang, "Semi-supervised distance metric learning for collaborative image retrieval," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–7.

[48] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods: Support Vector Learning*. Cambridge, MA, USA: MIT Press, 1999, pp. 185–208.

[49] A. Frank and A. Asuncion. (2010). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[50] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.

[51] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Univ. Massachusetts, Boston, MA, USA, Tech. Rep. 07-49, 2007.

[52] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. BMVC*, 2015, pp. 41.1–41.12.

[53] Q. Cao, Y. Ying, and P. Li, "Similarity metric learning for face recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 2408–2415.

[54] S. Gong, M. Cristani, S. Yan, and C. C. Loy, *Person Re-Identification*. London, U.K.: Springer, 2014.

[55] W. Li, R. Zhao, T. Xiao, and X. Wang, "DeepReID: Deep filter pairing neural network for person re-identification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 152–159.

[56] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, 2012, pp. 31–44.

[57] B. McFee and G. Lanckriet, "Metric learning to rank," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 775–782.

[58] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2360–2367.

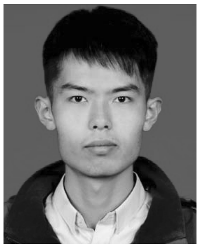
[59] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 3586–3593.

[60] R. Zhao, W. Ouyang, and X. Wang, "Learning mid-level filters for person re-identification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 144–151.

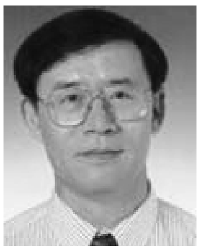
[61] R. Zhao, W. Ouyang, and X. Wang, "Person re-identification by salience matching," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 2528–2535.

[62] Z. Zhang, Y. Chen, and V. Saligrama, "A novel visual word co-occurrence model for person re-identification," in *Proc. ECCV Workshop Vis. Surveill. Re-Identificat.*, 2014, pp. 122–133.

Wangmeng Zuo (M'09–SM'14) received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2007. From 2004 to 2004 and from 2005 to 2006, he was a Research Assistant with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. From 2009 to 2010, he was a Visiting Professor with Microsoft Research Asia. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology. He has published over 60 papers in top tier academic journals and conferences. His current research interests include image enhancement and restoration, object detection, visual tracking, and image classification. He has served as a Tutorial Organizer in ECCV 2016, an Associate Editor of the IET Biometrics, *Journal of Electronic Imaging*, and the Guest Editor of *Neurocomputing*, *Pattern Recognition*, the IEEE T-CSVT, and the IEEE TNNLS.



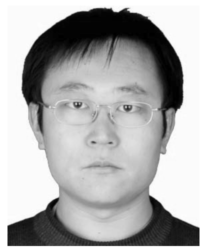
Faqiang Wang received the B.E. and M.E. degrees from the Harbin Institute of Technology, Harbin, China, in 2011 and 2013, respectively, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Technology. He was a Research Assistant with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, from 2012 to 2013 and in 2014. His current research interests include discriminative learning and computer vision.



David Zhang (F'09) received the degree in computer science from Peking University, the M.Sc. and Ph.D. degrees in computer science from the Harbin Institute of Technology (HIT), in 1982 and 1985, respectively, and the second Ph.D. degree in electrical and computer engineering from the University of Waterloo, ON, Canada, in 1994. From 1986 to 1988, he was a Post-Doctoral Fellow with Tsinghua University and then an Associate Professor with Academia Sinica, Beijing. He is currently a Chair Professor with The Hong Kong Polytechnic University, where he is also the Founding Director of the Biometrics Research Centre (UGC/CRC) supported by the Hong Kong SAR Government in 1998. He also serves as a Visiting Chair Professor with Tsinghua University and HIT, and an Adjunct Professor with Shanghai Jiao Tong University, Peking University, the National University of Defense Technology, and the University of Waterloo. He is the Author of over 15 books and over 400 international journal papers. He is a Croucher Senior Research Fellow, a Distinguished Speaker of the IEEE Computer Society, and a fellow of IAPR. He is the Technical Committee Chair of the IEEE SMC. He is the Founder and an Editor-in-Chief of the *International Journal of Image and Graphics*; a Book Editor of the *International Series on Biometrics* (Springer); an Organizer of the first International Conference on Biometrics Authentication; and an Associate Editor of over ten international journals, including the IEEE Transactions. He has been listed as a Highly Cited Researchers in Engineering by Thomos Reters in 2014, 2015, and 2016, respectively.

Liang Lin received the B.S. degree from the Beijing Institute of Technology, Beijing, China, in 2003, and the joint Ph.D. degree from the Beijing Institute of Technology and the Department of Statistics, University of California at Los Angeles (UCLA), Los Angeles, in 2008. From 2008 to 2010, he was a Post-Doctoral Fellow with UCLA. From 2014 to 2015, he was a Senior Visiting Scholar with The Hong Kong Polytechnic University and The Chinese University of Hong Kong. He is currently a Full Professor with Sun Yat-sen University. He has authored or co-authored over 100 papers in top-tier academic journals and conferences. His research interests include computer vision, data analysis and mining, and intelligent robotic systems. He was a recipient of the Best Paper Runners-Up Award in ACM NPAR 2010, the Google Faculty Award in 2012, the Best Student Paper Award in the IEEE ICME 2014, the Hong Kong Scholars Award in 2014, and The World's First 10K Best Paper Diamond Award in the IEEE ICME 2017. He has been serving as an Associate Editor of the IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS.

Yuchi Huang received the B.E. degree in control from the Beijing University of Aeronautics and Astronautics, the M.S. degree in pattern recognition from the Chinese Academy of Sciences in 2004, and the Ph.D. degree in computer science from Rutgers University in 2010. From 2010 to 2015, he was with GE Global Research and NEC Labs as a Computer Vision Scientist. He is currently a Senior Researcher with the Research Division, Educational Testing Service, Princeton, NJ, USA, where he is conducting research in computer vision, affective computing, and multimodal analytics for new forms of learning and educational assessment tools.



Deyu Meng received the B.Sc., M.Sc., and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 2001, 2004, and 2008, respectively. From 2012 to 2014, he took his two-year sabbatical leave with Carnegie Mellon University. He is currently a Professor with the School of Mathematics and Statistics, Institute for Information and System Sciences, Xi'an Jiaotong University. His current research interests include self-paced learning, noise modeling, and tensor sparsity.

Lei Zhang (M'04–SM'14) received the B.Sc. degree from the Shenyang Institute of Aeronautical Engineering, Shenyang, China, in 1995, and the M.Sc. and Ph.D. degrees in control theory and engineering from Northwestern Polytechnical University, Xi'an, China, in 1998 and 2001, respectively. From 2001 to 2002, he was a Research Associate with the Department of Computing, The Hong Kong Polytechnic University. From 2003 to 2006, he was a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, McMaster University, Canada. In 2006, he joined the Department of Computing, The Hong Kong Polytechnic University, as an Assistant Professor, where he has been a Chair Professor since 2017. His research interests include computer vision, pattern recognition, image and video analysis, and biometrics. He has authored over 200 papers in those areas. As of 2017, his publications have been cited more than 26,000 times in the literature. He is an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, the *SIAM Journal of Imaging Sciences*, and *Image and Vision Computing*. He is a Web of Science Highly Cited Researcher selected by Thomson Reuters.