Hindawi
Complexity
Volume 2018, Article ID 2857594, 11 pages
https://doi.org/10.1155/2018/2857594

Research Article

Manifold Adaptive Kernelized Low-Rank Representation for Semisupervised Image Classification

Yong Peng,[1,2] **Wanzeng Kong**,[1] **Feiwei Qin**,[1] **and Feiping Nie**[3]

[1] *School of Computer Science, Hangzhou Dianzi University, Hangzhou 310018, China*
[2] *Jiangsu Key Laboratory of Big Data Security & Intelligent Processing, Nanjing University of Posts and Telecommunications, Nanjing 210023, China*
[3] *Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China*

Correspondence should be addressed to Yong Peng; yongpeng@hdu.edu.cn

Received 19 December 2017; Revised 22 March 2018; Accepted 2 April 2018; Published 15 May 2018

Academic Editor: Eulalia Martínez

Constructing a powerful graph that can effectively depict the intrinsic connection of data points is the critical step to make the graph-based semisupervised learning algorithms achieve promising performance. Among popular graph construction algorithms, low-rank representation (LRR) is a very competitive one that can simultaneously explore the global structure of data and recover the data from noisy environments. Therefore, the learned low-rank coefficient matrix in LRR can be used to construct the data affinity matrix. Consider the existing problems such as the following: (1) the essentially linear property of LRR makes it not appropriate to process the possible nonlinear structure of data and (2) learning performance can be greatly enhanced by exploring the structure information of data; we propose a new manifold kernelized low-rank representation (MKLRR) model that can perform LRR in the data manifold adaptive kernel space. Specifically, the manifold structure can be incorporated into the kernel space by using graph Laplacian and thus the underlying geometry of data is reflected by the wrapped kernel space. Experimental results of semisupervised image classification tasks show the effectiveness of MKLRR. For example, MKLRR can, respectively, obtain 96.13%, 98.09%, and 96.08% accuracies on ORL, Extended Yale B, and PIE data sets when given 5, 20, and 20 labeled face images per subject.

1. Introduction

Since it is usually not easy to collect a large number of labeled samples to train learning models, the semisupervised learning (SSL) paradigm, which can harness both labeled and unlabeled samples to improve the learning performance, draws a lot of attention in recent studies [1–7]. Among existing SSL algorithms, graph-based algorithms are a class of the most popular approaches in which the label propagation can be performed on the graph [8–11]. The underlying idea for graph-based algorithms is to characterize the relationship between data pairs by an affinity matrix. Although researchers have pointed that sparsity, high discriminative power, and adaptive neighborhood are desirable properties for a good graph [12], how to learn a good graph that can accurately uncover the latent relationship in data is still a challenging problem.

Among existing graph construction methods, the k-nearest neighbors and ε-neighborhood are the two most widely used algorithms. However, they are usually sensitive to noisy environments, especially when data points contain outliers. To construct more effective graph, many new algorithms were proposed. The sparse graph [8] is parameter-free and insensitive to outliers, which is derived by encoding each datum as a sparse representation of the remaining samples. Sparse graph can automatically select the most informative neighbors for each datum. However, since sparse representation encodes each datum individually, the resultant sparse graph only emphasizes the local structure of data, while it neglects considering the global structure of data. This property will deteriorate its performance, especially when data are grossly corrupted [13]. Different from sparse representation that enforces the representation coefficient to be sparse [14], low-rank representation aims to learn the data affinities jointly, which can reveal the global structure of data and preserve the membership of samples that belong to the same class in noisy environments [15, 16]. The learned LRR graph can capture the global mixture of subspaces structure via the low rankness property and thus it is generative and discriminative for semisupervised learning tasks [9].

Apart from the conventional LRR model, many advanced variants were proposed recently. To efficiently explore the structure information of data, Zheng et al. imposed the local constraint characteristic on the representation coefficient and thus formulated the low-rank representation with local constraint (LRRLC) model [10]. Lu et al. proposed the graph regularized LRR (GLRR) that introduces the graph regularizer to enforce the local consistency of data [17]. Zhuang et al. proposed incorporating the sparse and nonnegative constraints into low-rank representation and formulated the NNLRS model [9]. The manifold low-rank representation (MLRR) [18] first uses a sparse learning objective to identify the data manifold and then incorporates the manifold information into low-rank representation as a regularizer. Additionally, [19] proposed preserving the structure information of data from two aspects: local affinity and distant repulsion. Li and Fu proposed constructing graph based on low-rank coding and b-matching constraint for obtaining a sparse and balanced graph [20]. All the above-mentioned low-rank models are linear; therefore, they inevitably have limitations in modeling complex data distribution, which does not strictly follow a linear model but a nonlinear one. To make the low-rank model effectively deal with the nonlinear structure of data, [11] proposed the kernel low-rank representation (KLRR) graph for semisupervised classification by using kernel trick. As a nonlinear extension of LRR, KLRR also showed excellent performance in face recognition [21].

Recent studies [22–26] have shown that learning performance can be greatly enhanced by considering the geometrical structure and local invariant idea [27]. It is obvious that this idea should be considered in both original data space and the reproducing kernel Hilbert space (RKHS). However, there is no existing LRR variant that takes into account the intrinsic manifold structure in RKHS. In this paper, we propose a novel manifold adaptive kernelized LRR for semisupervised classification. By using a data-dependent norm on RKHS proposed by [28], we can warp the structure of the RKHS to reflect the underlying geometry of the data. Then, the conventional low-rank representation can be performed in the manifold adaptive kernel space. The main contributions of this paper can be briefly summarized as follows:

(1) We construct the manifold adaptive kernel space, where the underlying geometry of data can be reflected by the graph Laplacian.

(2) We give the model formulation, the optimization method, and the complexity analysis of MKLRR in detail.

(3) We conduct extensive experiments on semisupervised image classification tasks to evaluate the effectiveness of MKLRR and the experimental results show that MKLRR can get pretty promising performance.

The remainder of this paper is organized as follows. In Section 2, we give a brief review on the conventional LRR model and the semisupervised learning framework to be used in our work. Section 3 describes the model formulation, optimization method, and complexity analysis of the manifold adaptive kernelized LRR model in detail. Experimental studies of MKLRR on semisupervised image classification task will be introduced in Section 4. Section 5 concludes the whole paper and presents an extension of MKLRR as our future work.

2. Related Work

In this section, we give a brief review of the conventional low-rank representation model [15] and the semisupervised classification framework based on Gaussian Fields and Harmonic Functions (GHF) [1].

2.1. LRR. Given a set of samples $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, LRR aims to represent each sample as a linear combination of the bases in $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m] \in \mathbb{R}^{d\times m}$ by $\mathbf{X} = \mathbf{AZ}$, where $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n]$ is the matrix in which each $\mathbf{z}_i$ is the representation coefficient corresponding to sample $\mathbf{x}_i$. Therefore, each entry in $\mathbf{z}_i$ can be viewed as the contribution to the reconstruction of $\mathbf{x}_i$ with $\mathbf{A}$ as the dictionary. LRR seeks to find the lowest rank solution by solving the following optimization problem [15]:

$$\begin{aligned} \min_{\mathbf{Z}} \quad & \operatorname{rank}(\mathbf{Z}), \\ \text{s.t.} \quad & \mathbf{X} = \mathbf{AZ}. \end{aligned} \tag{1}$$

It is NP-hard to directly optimize the rank function. Therefore, we usually use the trace norm (also called nuclear norm) as the closest convex surrogate to the rank norm, which leads to the following objective [29]:

$$\begin{aligned} \min_{\mathbf{Z}} \quad & \|\mathbf{Z}\|_*, \\ \text{s.t.} \quad & \mathbf{X} = \mathbf{AZ}, \end{aligned} \tag{2}$$

where $\|\cdot\|_*$ is the sum of its singular values of a certain matrix [30]. Considering the fact that samples are usually noisy or even grossly corrupted, a more reasonable objective for LRR can be expressed as

$$\begin{aligned} \min_{\mathbf{Z},\mathbf{E}} \quad & \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_{2,1}, \\ \text{s.t.} \quad & \mathbf{X} = \mathbf{AZ} + \mathbf{E}, \end{aligned} \tag{3}$$

where $\mathbf{E} \in \mathbb{R}^{d\times n}$ and $\|\mathbf{E}\|_{2,1} = \sum_{j=1}^{n} \sqrt{\sum_{i=1}^{d} e_{ij}^2}$. The second term in (3) is to characterize the error term by modeling the sample-specific corruptions. Also, some existing studies employed the ℓ_1-norm to measure the error term [31, 32]. The optimal solution $\mathbf{Z}^*$ can be obtained via the inexact augmented Lagrange multiplier method [31].

2.2. GHF. Assume that we have a data set $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_l, \mathbf{x}_{l+1}, \dots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ from c classes, where $\mathbf{x}_i$, $i = 1, \dots, l$, and

$\mathbf{x}_i$, $i = l+1,\dots,n$, are the labeled and unlabeled samples, respectively. The label indicator matrix $\mathbf{Y} \in \mathbb{R}^{n\times c}$ is defined as follows: for each sample $\mathbf{x}_i$ ($i = 1,\dots,n$), $\mathbf{y}_i \in \mathbb{R}^c$ is its label vector. If $\mathbf{x}_i$ is from the kth ($k = 1,2,\dots,c$) class, then only the kth entry of $\mathbf{y}_i$ is one and all the other entries are zeros. If $\mathbf{x}_i$ is an unlabeled data, then $\mathbf{y}_i = \mathbf{0}$.

GHF is a well-known graph-based semisupervised learning framework in which the predicted label matrix $\mathbf{F} \in \mathbb{R}^{n\times c}$ is estimated on the graph with respect to the label fitness and the manifold smoothness. Let $\mathbf{f}_i$ and $\mathbf{y}_i$, respectively, denote the ith rows of $\mathbf{F}$ and $\mathbf{Y}$. GHF tries to minimize the following objective:

$$\min_{\mathbf{F}} \quad \frac{1}{2}\sum_{i,j=1}^{n} \left\|\mathbf{f}_i - \mathbf{f}_j\right\|^2 s_{ij} + \lambda_\infty \sum_{i=1}^{l} \left\|\mathbf{f}_i - \mathbf{y}_i\right\|^2, \tag{4}$$

where λ_∞ is a very large value such that $\sum_{i=1}^{l}\|\mathbf{f}_i - \mathbf{y}_i\|^2$ can be approximately satisfied. $\mathbf{S} \in \mathbb{R}^{n\times n}$ is an affinity matrix to depict the pairwise similarity of samples. Obviously, (4) can be rewritten in the compact matrix form as

$$\min_{\mathbf{F}} \quad \frac{1}{2}\operatorname{Tr}\left(\mathbf{F}^T\mathbf{L}_\mathbf{s}\mathbf{F}\right) + \operatorname{Tr}\left((\mathbf{F}-\mathbf{Y})^T\mathbf{U}(\mathbf{F}-\mathbf{Y})\right), \tag{5}$$

where the graph Laplacian matrix $\mathbf{L}_\mathbf{S} \in \mathbb{R}^{n\times n}$ can be calculated as $\mathbf{L}_\mathbf{S} = \mathbf{D} - \mathbf{S}$; $d_{ii} = \sum_j s_{ij}$ (or $\sum_i s_{ij}$ since $\mathbf{S}$ is usually a symmetric matrix) is a diagonal degree matrix. $\mathbf{U}$ is also a diagonal matrix with the first l and the remaining $n-l$ diagonal entries as λ_∞ and 0, respectively.

3. Manifold Adaptive Low-Rank Representation

3.1. Manifold Adaptive Kernel. In this section, we show how to incorporate the manifold structure into the reproducing kernel Hilbert space (RKHS), which leads to manifold adaptive kernel space.

Kernel trick is usually applied with the hope of discovering the nonlinear structure in data by mapping the original nonlinear observations into a higher dimensional linear space [33]. The most commonly used kernels are Gaussian and Polynomial kernels. However, the nonlinear structure captured by the data-independent kernels may not be consistent with intrinsic manifold structure, such geodesic distance, curvature, and homology [34, 35].

In this work, we adopt the manifold adaptive kernel proposed by [28]. Let $\mathscr{V}$ be a linear space with a positive semidefinite inner product (quadratic form) and let $S : \mathscr{H} \to \mathscr{V}$ be a bounded linear operator. We define $\widetilde{\mathscr{H}}$ to be the space of functions from $\mathscr{H}$ with manifold inner product:

$$\langle f, g\rangle_{\widetilde{\mathscr{H}}} = \langle f, g\rangle_{\mathscr{H}} + \langle Sf, Sg\rangle_{\mathscr{V}}; \tag{6}$$

$\widetilde{\mathscr{H}}$ is still a RKHS [28].

Given samples $\mathbf{x}_1,\dots,\mathbf{x}_m$, let $S : \mathscr{H} \to \mathbb{R}^m$ be the evaluation map.

$$S(f) = \left(f(\mathbf{x}_1),\dots,f(\mathbf{x}_m)\right)^T. \tag{7}$$

Denote $\mathbf{f} = (f(\mathbf{x}_1),\dots,f(\mathbf{x}_m))^T$ and $\mathbf{g} = (g(\mathbf{x}_1),\dots,g(\mathbf{x}_m))^T$. Note that $\mathbf{f},\mathbf{g} \in \mathscr{V}$; thus we have

$$\langle Sf, Sg\rangle_{\mathscr{V}} = \langle \mathbf{f}, \mathbf{g}\rangle = \mathbf{f}^T\mathbf{M}\mathbf{g}, \tag{8}$$

where $\mathbf{M}$ is a positive semidefinite matrix. For a data vector $\mathbf{x}$, we define

$$\boldsymbol{\kappa}_\mathbf{x} = \left(\kappa(\mathbf{x},\mathbf{x}_1),\dots,\kappa(\mathbf{x},\mathbf{x}_m)\right). \tag{9}$$

It can be shown that the reproducing kernel in $\widetilde{\mathscr{H}}$ is

$$\widetilde{\kappa}(\mathbf{x},\mathbf{z}) = \kappa(\mathbf{x},\mathbf{z}) - \gamma\boldsymbol{\kappa}_\mathbf{x}^T(\mathbf{I}+\mathbf{M}\mathbf{K})^{-1}\mathbf{M}\boldsymbol{\kappa}_\mathbf{z}, \tag{10}$$

where $\mathbf{I}$ is an identity matrix, $\mathbf{K}$ is the kernel matrix in $\mathscr{H}$, and $\gamma \geq 0$ is a constant controlling the smoothness of the functions. The key issue now is the choice of $\mathbf{M}$, so that the deformation of the kernel induced by the data-dependent norm is motivated with respect to the intrinsic geometry of the data.

Without loss of generality, we assume that there are n_q data points to be utilized to derive the linear space $\mathscr{V}$. It is easy to rewrite formulation (10) in compact matrix form as

$$\widetilde{\mathbf{K}}_{\widetilde{\mathscr{H}}} = \mathbf{K}_{\mathscr{H}} - \gamma\mathbf{K}_{\mathscr{H}_{qn}}^T\left(\mathbf{I}_{\mathscr{H}_q} + \mathbf{M}_{\mathscr{H}_q}\mathbf{K}_{\mathscr{H}_q}\right)^{-1}\mathbf{M}_{\mathscr{H}_q}\mathbf{K}_{\mathscr{H}_{qn}}, \tag{11}$$

where the matrices $\mathbf{K}_{\mathscr{H}} \in \mathbb{R}^{n\times n}$, $\mathbf{K}_{\mathscr{H}_{qn}} \in \mathbb{R}^{n_q\times n}$, and $\mathbf{K}_{\mathscr{H}_q} \in \mathbb{R}^{n_q\times n_q}$ are all in $\mathscr{H}$. Here, $\mathbf{I}$ is an identity matrix with the same size as $\mathbf{K}_{\mathscr{H}_q}$. $\widetilde{\mathbf{K}}_{\widetilde{\mathscr{H}}}$ was referred to as the kernel matrix in the warped RKHS.

The key issue now is the choice of $\mathbf{M}$. As mentioned above, manifold structure can be discovered by the graph Laplacian associated with the data points.

3.2. The Objective Function. From [11], the objective of kernel low-rank representation was formulated as

$$\begin{aligned} \min_{\overline{\mathbf{Z}},\overline{\mathbf{E}}} \quad & \operatorname{rank}\left(\overline{\mathbf{Z}}\right) + \lambda\left\|\overline{\mathbf{E}}\right\|_{2,1}, \\ \text{s.t.} \quad & \mathbf{K} = \mathbf{K}\overline{\mathbf{Z}} + \overline{\mathbf{E}}. \end{aligned} \tag{12}$$

In order to learn the low-rank representation that is consistent with the manifold geometry, it is natural to take advantage of the manifold adaptive kernel in KLRR.

In order to model the manifold structure, we construct a nearest-neighbor graph G. For each data point $\mathbf{x}_i$, we find its p nearest neighbors denoted by $\mathscr{N}(\mathbf{x}_i)$ and put an edge between $\mathbf{x}_i$ and its neighbors. There are many choices for the weight matrix on the graph and we use the "0-1" form defined as follows:

$$w_{ij} = \begin{cases} 1, & \text{if } \mathbf{x}_i \in \mathscr{N}_p\left(\mathbf{x}_j\right) \text{ or } \mathbf{x}_j \in \mathscr{N}_p\left(\mathbf{x}_i\right) \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

The graph Laplacian [36] is defined as $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D}$ is a diagonal degree matrix given by $d_{ii} = \sum_j w_{ij}$ (or $\sum_i w_{ij}$ since

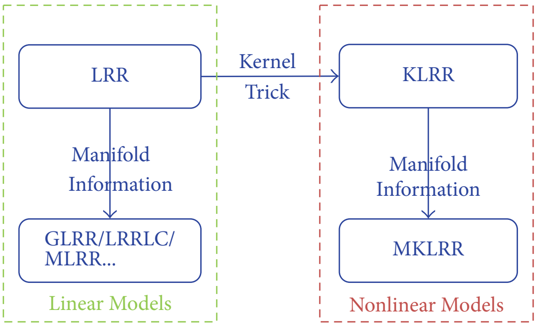


Figure 1: Connections between several LRR models.

$\mathbf{W}$ is symmetric). The graph Laplacian provides the following smoothness penalty on the graph:

$$\mathbf{f}^T\mathbf{L}\mathbf{f} = \sum_{i=1}^{n}\left(f\left(\mathbf{x}_i\right) - f\left(\mathbf{x}_j\right)\right)^2 w_{ij}. \tag{14}$$

Therefore, it is natural to substitute $\mathbf{M}$ with the graph Laplacian $\mathbf{L}$. For convenience, we make use of all the available data points to derive the linear space $\mathscr{V}$ in the warped RKHS (i.e., $n_q = n$); then (11) can be rewritten as

$$\widetilde{\mathbf{K}}_{\mathcal{M}} = \mathbf{K} - \gamma\mathbf{K}^T\left(\mathbf{I} + \mathbf{L}\mathbf{K}\right)^{-1}\mathbf{M}\mathbf{K}, \tag{15}$$

where $\widetilde{\mathbf{K}}_{\mathcal{M}}$ indicates that this kernel matrix is in a manifold RKHS.

Using the nuclear norm to replace the rank function, we arrive at the following objective of manifold adaptive kernelized LRR as

$$\begin{aligned} \min_{\overline{\mathbf{Z}},\overline{\mathbf{E}}} \quad & \left\|\overline{\mathbf{Z}}\right\|_* + \lambda\left\|\overline{\mathbf{E}}\right\|_{2,1}, \\ \text{s.t.} \quad & \widetilde{\mathbf{K}}_{\mathcal{M}} = \widetilde{\mathbf{K}}_{\mathcal{M}}\overline{\mathbf{Z}} + \overline{\mathbf{E}}. \end{aligned} \tag{16}$$

Figure 1 shows the connection between MKLRR and LRR as well as its variants. As we can see, LRR variants such as GLRR, LRRLC, and MLRR can be reached by incorporating manifold information. By using the kernel trick, the KLRR model can find the lowest rank representation in RKHS. Further, by considering the geometric structure of data in RKHS, we can formulate the MKLRR model. Both KLRR and MKLRR are nonlinear models, since an implicit nonlinear mapping is employed.

3.3. Optimization. To make objective (16) separable, we introduce an auxiliary variable $\mathbf{J}$ with respect to $\overline{\mathbf{Z}}$ and then we have the following objective:

$$\begin{aligned} \min_{\overline{\mathbf{Z}},\overline{\mathbf{E}},\mathbf{J}} \quad & \|\mathbf{J}\|_* + \lambda\left\|\overline{\mathbf{E}}\right\|_{2,1}, \\ \text{s.t.} \quad & \widetilde{\mathbf{K}}_{\mathcal{M}} = \widetilde{\mathbf{K}}_{\mathcal{M}}\overline{\mathbf{Z}} + \overline{\mathbf{E}}, \quad \overline{\mathbf{Z}} = \mathbf{J}. \end{aligned} \tag{17}$$

The corresponding augmented Lagrangian function is

$$\begin{aligned} \min_{\overline{\mathbf{Z}},\mathbf{J},\overline{\mathbf{E}},\mathbf{Y}_1,\mathbf{Y}_2} \quad & \|\mathbf{J}\|_* + \lambda\left\|\overline{\mathbf{E}}\right\|_{2,1} + \left\langle \mathbf{Y}_1, \widetilde{\mathbf{K}}_{\mathcal{M}} - \widetilde{\mathbf{K}}_{\mathcal{M}}\overline{\mathbf{Z}} - \overline{\mathbf{E}}\right\rangle \\ & + \left\langle \mathbf{Y}_2, \overline{\mathbf{Z}} - \mathbf{J}\right\rangle \\ & + \frac{\mu}{2}\left(\left\|\widetilde{\mathbf{K}}_{\mathcal{M}} - \widetilde{\mathbf{K}}_{\mathcal{M}}\overline{\mathbf{Z}} - \overline{\mathbf{E}}\right\|_F^2 + \left\|\overline{\mathbf{Z}} - \mathbf{J}\right\|_F^2\right), \end{aligned} \tag{18}$$

where $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are Lagrange multipliers and $\mu > 0$ is a penalty parameter. The inexact augmented Lagrange multiplier (ALM) algorithm is employed to optimize objective (18) [31]. The detailed optimization process is summarized in Algorithm 1.

The updating rule for $\mathbf{J}$ is based on singular value thresholding operator which is given by the following theorem [30].

Theorem 1. *Let $\mathbf{C} \in \mathbb{R}^{m\times n}$ and $\mathbf{C} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ be the SVD of $\mathbf{C}$, where $\mathbf{U} \in \mathbb{R}^{m\times r}$ and $\mathbf{V} \in \mathbf{R}^{n\times r}$ have orthonormal columns, $\mathbf{\Sigma} \in \mathbb{R}^{r\times r}$ is diagonal, and $r = \operatorname{rank}(\mathbf{C})$. Then*

$$\mathcal{T}_\lambda\left(\mathbf{C}\right) = \arg\min_{\mathbf{W}}\left\{\frac{1}{2}\left\|\mathbf{W} - \mathbf{C}\right\|_F^2 + \lambda\left\|\mathbf{W}\right\|_*\right\} \tag{19}$$

is given by $\mathcal{T}_\lambda(\mathbf{C}) = \mathbf{U}\mathbf{\Sigma}_\lambda\mathbf{V}^T$, where $\mathbf{\Sigma}_\lambda$ is diagonal with $(\Sigma_\lambda)_{ii} = \max\{0, \Sigma_{ii} - \lambda\}$.

The updating rule for $\overline{\mathbf{E}}$ can be obtained by soft-shrinkage operator [15], which is given as below.

Theorem 2. *Let $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_i, \ldots]$ be a given matrix and let $\|\cdot\|_F$ be the Frobenius norm. If the optimal solution to*

$$\min_{\mathbf{W}} \quad \lambda\left\|\mathbf{W}\right\|_{2,1} + \frac{1}{2}\left\|\mathbf{W} - \mathbf{Q}\right\|_F^2 \tag{20}$$

is $\mathbf{W}^*$, then the ith column of $\mathbf{W}^*$ is

$$\mathbf{W}^*\left(:, i\right) = \begin{cases} \dfrac{\left\|\mathbf{q}_i\right\| - \lambda}{\left\|\mathbf{q}_i\right\|}\mathbf{q}_i, & \textit{if } \lambda < \left\|\mathbf{q}_i\right\| \\ 0, & \text{otherwise.} \end{cases} \tag{21}$$

3.4. Algorithm Workflow and Complexity Analysis. As a whole, we summarize the manifold adaptive kernelized low-rank representation-based semisupervised classification algorithm as follows:

(i) *Construct the graph Laplacian*: construct a p nearest-neighbor graph G with weight matrix defined in (13) and then calculate the graph Laplacian by $\mathbf{L} = \mathbf{D} - \mathbf{W}$

(ii) *Calculate the manifold adaptive kernel*: assume that the kernel matrix $\mathbf{K}$ in $\mathscr{H}$ can be induced from any data-independent kernel (e.g., Gaussian kernel or linear kernel). Then calculate the manifold adaptive kernel $\widetilde{\mathbf{K}}_{\mathcal{M}}$ in the warped RKHS according to (15)

(iii) *Manifold kernel low-rank representation*: optimize the MKLRR model and obtain the low-rank representation coefficient matrix $\overline{\mathbf{Z}}$ via Algorithm 1. Shrink some small values in $\overline{\mathbf{Z}}$ and then make it symmetric and nonnegative as $\overline{\mathbf{Z}} = (|\overline{\mathbf{Z}}| + |\overline{\mathbf{Z}}^T|)/2$

```
Input: data points $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{l} \bigcup \{\mathbf{x}_i\}_{i=l+1}^{n}$, regularization parameters $\lambda$,
    $\overline{\mathbf{Z}} = \mathbf{J} = \mathbf{0}, \overline{\mathbf{E}} = \mathbf{0}, \mathbf{Y}_1 = \mathbf{Y}_2 = \mathbf{0}, \mu = 1e-6, \mu_{\max} = 1e6, \rho = 1.1$, and $\varepsilon = 1e-8$;
Output: the low-rank representation coefficient matrix $\overline{\mathbf{Z}}$.
while not converged do
    Fix the other variables and update $\mathbf{J}$ by
        $\mathbf{J} = \arg\min_{\mathbf{J}} \frac{1}{\mu} \|\mathbf{J}\|_* + \frac{1}{2} \left\| \mathbf{J} - \left( \overline{\mathbf{Z}} + \frac{\mathbf{Y}_2}{\mu} \right) \right\|_F^2 \quad (*)$
    Fix the others and update $\overline{\mathbf{Z}}$ by
        $\overline{\mathbf{Z}} = \left( \mathbf{I} + \widetilde{\mathbf{K}}_{\mathcal{M}}^T \widetilde{\mathbf{K}}_{\mathcal{M}} \right)^{-1} \left( \widetilde{\mathbf{K}}_{\mathcal{M}}^T \widetilde{\mathbf{K}}_{\mathcal{M}} - \widetilde{\mathbf{K}}_{\mathcal{M}}^T \overline{\mathbf{E}} + \mathbf{J} + \frac{\left( \widetilde{\mathbf{K}}_{\mathcal{M}}^T \mathbf{Y}_1 - \mathbf{Y}_2 \right)}{\mu} \right)$
    Fix the others and update $\overline{\mathbf{E}}$ by
        $\overline{\mathbf{E}} = \arg\min_{\overline{\mathbf{E}}} \frac{\lambda}{\mu} \left\| \overline{\mathbf{E}} \right\|_{2,1} + \frac{1}{2} \left\| \overline{\mathbf{E}} - \left( \widetilde{\mathbf{K}}_{\mathcal{M}} - \widetilde{\mathbf{K}}_{\mathcal{M}} \overline{\mathbf{Z}} + \frac{\mathbf{Y}_1}{\mu} \right) \right\|_F^2$
    Update the multipliers
        $\mathbf{Y}_1 = \mathbf{Y}_1 + \mu \left( \widetilde{\mathbf{K}}_{\mathcal{M}} - \widetilde{\mathbf{K}}_{\mathcal{M}} \overline{\mathbf{Z}} - \overline{\mathbf{E}} \right)$
        $\mathbf{Y}_2 = \mathbf{Y}_2 + \mu \left( \overline{\mathbf{Z}} - \mathbf{J} \right)$
    Update the parameter $\mu$ by $\mu = \min(\rho\mu, \max_{\mu})$
    Check the convergence conditions
        $\|\widetilde{\mathbf{K}}_{\mathcal{M}} - \widetilde{\mathbf{K}}_{\mathcal{M}} \overline{\mathbf{Z}} - \overline{\mathbf{E}}\|_\infty < \varepsilon,$
        $\left\| \overline{\mathbf{Z}} - \mathbf{J} \right\|_\infty < \varepsilon$
end while
```

Algorithm 1: Optimization to (18).

(iv) *Semisupervised classification*: calculate the Laplacian matrix $\mathbf{L}_s = \mathbf{D} - \overline{\mathbf{Z}}$ and do semisupervised classification based on (5)

Below we give a brief analysis on the computational complexity of MKLRR. Constructing the p nearest-neighbor graph in the first step of MKLRR needs $O(pn^2)$. In the second step, computing the data-independent kernel matrix $\mathbf{K}$ needs $O(n^2)$ and computing the manifold adaptive kernel matrix needs $O(n^3)$. In the fourth step, the complexity of semisupervised learning based on GHF is $O(n^3)$ $(c \ll n)$. Below we give a detailed description on the complexity of Algorithm 1. Obviously, the main computation burden of MKLRR lies in the updating of $\mathbf{J}$, since it involves the singular value decomposition (SVD). Specifically, in equation $(*)$ in Algorithm 1, the SVD is operated on an $n \times n$ matrix, which is time-consuming if the number of samples (i.e., n) is large. As referred to in [37], by substituting $\mathbf{A}$ with the orthogonal basis of the dictionary, the computation can be reduced to $O(r^2 n)$, where r is the rank of dictionary $\mathbf{A}$. The computational complexity of updating $\overline{\mathbf{Z}}$ is trivial owing to its simple closed form solution. The complexity of updating $\overline{\mathbf{E}}$ is $O(n^2 r)$. Thus, the computation complexity of MKLRR-based semisupervised learning is $O(t(r^2 n + n^2 r) + pn^2 + n^3)$ in general, where t is the number of iterations of loop in Algorithm 1.

4. Experiments

This section evaluates the effectiveness of the proposed MKLRR algorithm on semisupervised classification task. Specifically, we will compare the performance of MKLRR with some state-of-the-art graph construction methods on four representative image data sets. All experiments are conducted on platform *Intel(R) Core(TM) i7-4700MQ CPU @2.40 GHz 16.0 GB RAM Windows 8.1 System* and *Matlab 2013a*.

4.1. Experimental Settings. For the comparison methods, several baseline methods are compared including some state-of-the-art graph-based semisupervised learning methods:

(i) kNN: if one sample is among the k nearest neighbors of the other, then these two samples are viewed as connected. In kNN-1, k is set to 5; and in kNN-2, k is set to 8. The distance information is measured by the "Heatkernel" function, where the variance δ is the average of squared Euclidean distances for all edged pairs on graph

(ii) ℓ_1 graph [8, 38]: the ℓ_1-norm regularized least squares problem is optimized by the ℓ_1-*ls* package [39]. The regularization parameter to enforce the sparsity is searched from $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2\}$

(iii) LNP (linear neighborhood propagation): we follow the pipeline in [40] to construct the graph. The neighborhood size in LNP is set to 40

(iv) SPG (sparse probability graph) [41]: we implement the SPG algorithm by setting n_{knn} as one-quarter of the size of data set and λ is set to 0.001 as suggested by [41]

(v) LRR (low-rank representation) [15]: for all data sets, we tune the parameter in the range $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2\}$ to achieve the best performance

Table 1: Classification accuracies (%) of different graphs on synthetic data set.

# labeled	kNN-1	kNN-2	LNP	ℓ_1-graph	SPG	LRR	NNLRS	GLRR	LRCB	KLRR	MKLRR
1	93.67 ± 0.74	93.81 ± 2.32	86.22 ± 4.72	75.12 ± 8.73	85.30 ± 4.70	86.83 ± 5.35	90.32 ± 4.58	89.44 ± 5.82	90.51 ± 2.37	**92.59 ± 2.00**	92.33 ± 1.87
2	93.98 ± 0.34	94.61 ± 0.20	89.99 ± 1.49	85.57 ± 5.66	87.91 ± 2.62	92.34 ± 1.00	93.81 ± 2.17	94.10 ± 1.25	94.10 ± 1.98	94.69 ± 0.58	**94.98 ± 0.55**
3	93.93 ± 0.13	94.65 ± 0.24	90.80 ± 0.92	90.32 ± 2.95	91.59 ± 1.97	92.86 ± 0.62	94.21 ± 1.34	94.84 ± 0.51	94.67 ± 1.17	95.13 ± 0.47	**95.67 ± 0.43**
4	93.98 ± 0.20	94.59 ± 0.26	90.91 ± 0.72	91.11 ± 2.33	93.74 ± 1.75	93.11 ± 0.56	94.79 ± 0.57	94.92 ± 0.49	94.98 ± 0.62	95.32 ± 0.34	**96.04 ± 0.37**
5	93.95 ± 0.24	94.67 ± 0.24	91.22 ± 0.33	92.83 ± 1.26	95.52 ± 1.45	93.20 ± 0.41	95.26 ± 0.35	95.12 ± 0.51	95.43 ± 0.41	95.30 ± 0.31	**96.48 ± 0.28**

(vi) GLRR (graph regularized low-rank representation) [17]: in [17], the accelerated gradient method [42] was employed to optimize GLRR by updating **J**, which is the corresponding auxiliary variable with respect to **Z**, while in our implementation, the GLRR objective function was relaxed as described in [10] and **J** was updated by using the SVT operator [31]

(vii) NNLRS (nonnegative low-rank and sparse graph) [9]: we construct the LRR graph with nonnegative and sparse properties. The weighted parameters are set as guided in [9]

(viii) LRCB (low-rank representation with b-matching constraint) [20]: as suggested in [20], we set the parameters λ_1 and λ_2 as 2 and 0.03 for all the four data sets. The parameter b is set as 5, 5, 10, and 5 in the ORL, Extended Yale B, PIE, and USPS data sets, respectively.

(ix) KLRR (kernel low-rank representation) [11]

For both KLRR and MKLRR, we use the Gaussian kernel function defined as $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2)$ and the band width parameter is set as the mean value of all the distances of each data pair. When constructing the weight matrix, the number of nearest neighbors p is set as 5. The regularization parameter λ in conventional LRR model is searched from the candidate values of $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^{2}\}$. Similar to the usage in [43], we fix the parameter $\gamma = 1$ in all experiments below.

4.2. Experiment on Synthetic Data. Similar to studies [11, 15], a synthetic data set is constructed as follows. We construct 5 independent subspaces $\{\mathcal{S}_i\}_{i=1}^{5} \subset \mathbb{R}^{100}$ whose bases $\{\mathbf{U}_i\}_{i=1}^{5}$ are computed by $\mathbf{U}_{i+1} = \mathbf{T}\mathbf{U}_i$, $i \leq i \leq 4$, where $\mathbf{T}$ is a random rotation and $\mathbf{U}_1$ is a random orthogonal matrix with dimension 100 × 100. Therefore, each subspace has a dimension of 100. We sample 200 data vectors from each subspace by $\mathbf{X}_i = \mathbf{U}_i\mathbf{Q}_i$, $1 \leq i \leq 5$, with $\mathbf{Q}_i$ being a 100 × 200 i.i.d. $\mathcal{N}(0, 1)$ matrix. We randomly choose 30% of the total samples to corrupt. For example, if data vector $\mathbf{x}$ is chosen, its observed vector is computed by adding Gaussian noise with zero mean and variance $0.3\|\mathbf{x}\|_2$.

We select different numbers of labeled samples to evaluate the performance of different graph construction methods. Table 1 shows the classification accuracies of different graphs on the synthetic data set. The results are obtained from ten independent runs. From the results, we can find that all LRR variants can achieve good performance even when given only a few labeled samples. KLRR is slightly better than MKLRR by 0.26% when there is only one labeled sample per class. MKLRR obtains the best results in all the remaining cases.

Since GLRR is also related to incorporating the structure information of data into LRR, we show the learned block diagonal structure, respectively, by LRR, GLRR, KLRR, and MKLRR in Figure 2. Generally, although the visual discrepancies between MKLRR and its counterparts are minor, the block diagonal structure obtained by MKLRR is clearer than the others. Most of the values within each block of MKLRR graph are obviously larger than those of KLRR graph.

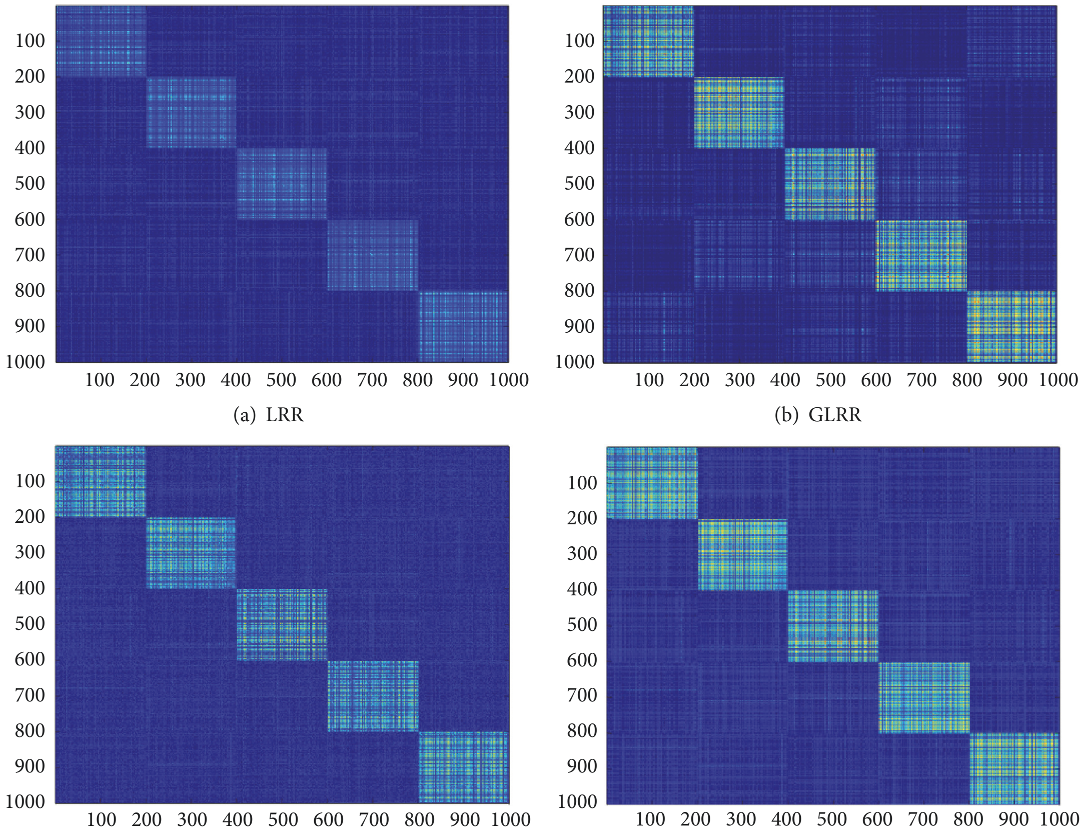
4.3. Experiment on ORL Data Set. The ORL data set contains ten different images of each of 40 distinct subjects. The images were taken at different times, varying the lighting, facial expressions, and facial details. Each image is manually cropped and normalized to size of 32 × 32 pixels. Figure 3 shows some example images of two subjects from the ORL data set.

We repeat all the experiments ten times. In each time, we randomly select a subset of images from each subject to create a labeled sample set. In this experiment, 1, 2, 3, 4, and 5 images per subject are randomly selected as labeled samples and the remaining images are regarded as unlabeled samples. The random indices are kept the same for all compared algorithms. The classification accuracies of different algorithms with different numbers of labeled samples on the ORL data set are shown in Table 2, in which MKLRR outperforms all the compared algorithms. For example, when we select 1, 2, 3, 4, and 5 images per person as labeled samples, the accuracies of MKLRR are higher than those of the second best algorithm by 1.88% (LCRB), 3.18% (KLRR), 2.18% (SPG), 0.85% (SPG), and 1.28% (SPG), respectively.

4.4. Experiment on Extended Yale B Data Set. The Extended Yale B data set consists of 2414 human face images of 38

Table 2: Classification accuracies (%) of different algorithms with different number of labeled samples on ORL data set.

# labeled	*k*NN-1	*k*NN-2	LNP	ℓ_1-graph	SPG
1	64.11 ± 2.74	56.69 ± 3.76	71.50 ± 2.25	65.58 ± 2.36	74.03 ± 3.14
2	72.53 ± 2.23	65.47 ± 2.28	82.44 ± 2.20	77.47 ± 2.03	85.41 ± 1.25
3	76.75 ± 2.52	69.93 ± 2.65	87.79 ± 2.53	84.71 ± 2.67	89.39 ± 2.55
4	79.96 ± 2.61	73.42 ± 1.49	91.17 ± 2.32	88.87 ± 2.55	93.04 ± 1.63
5	82.20 ± 1.87	75.70 ± 1.70	93.70 ± 1.93	92.10 ± 2.25	94.85 ± 1.33
LRR	NNLRS	GLRR	LRCB	KLRR	MKLRR
67.28 ± 3.23	71.58 ± 3.64	67.33 ± 2.95	75.98 ± 2.39	74.67 ± 2.27	**77.86 ± 2.19**
81.34 ± 2.23	83.63 ± 2.56	82.56 ± 2.10	83.11 ± 1.83	85.97 ± 1.68	**89.15 ± 1.81**
86.71 ± 2.25	88.46 ± 2.39	88.29 ± 2.19	87.91 ± 1.42	88.25 ± 1.25	**91.57 ± 1.31**
90.58 ± 1.52	91.50 ± 1.80	91.88 ± 0.97	91.07 ± 1.35	91.92 ± 2.21	**93.89 ± 1.19**
92.60 ± 1.65	93.20 ± 1.78	94.35 ± 2.15	94.26 ± 1.12	94.10 ± 1.56	**96.13 ± 0.88**



(a) LRR

(b) GLRR

(c) KLRR

(d) MKLRR

Figure 2: Block diagonal structures, respectively, learned by LRR, GLRR, KLRR, and MKLRR.

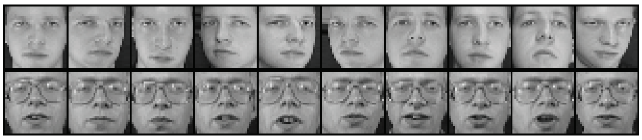


Figure 3: Example face images of two subjects from the ORL data set.

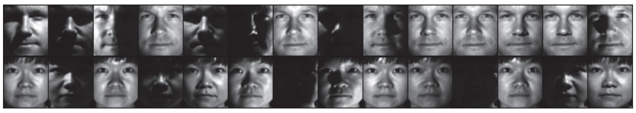


Figure 4: Example face images of two subjects from the Extended Yale B data set.

subjects. Each subject has about 64 images taken under different illuminations. Half of the images are corrupted by shadows or reflection. Each image is cropped to 32 × 32 pixels. Figure 4 shows some images of two subjects from the Extended Yale B data set.

We use the first 20 subjects and get 1262 images in total in the Extended Yale B data set to evaluate different methods.

Table 3: Classification accuracies (%) of different algorithms with different number of labeled samples on the Extended Yale B data set.

# labeled	kNN-1	kNN-2	LNP	ℓ_1-graph	SPG	LRR	NNLRS	GLRR	LRCB	KLRR	MKLRR
4	58.15 ± 2.41	44.77 ± 2.24	78.10 ± 2.09	82.75 ± 1.26	82.20 ± 1.17	77.86 ± 1.89	92.11 ± 1.02	82.24 ± 1.38	91.03 ± 1.07	91.12 ± 0.57	**94.08 ± 1.12**
8	66.04 ± 2.14	52.67 ± 2.16	86.74 ± 1.48	90.05 ± 1.60	89.80 ± 1.42	87.28 ± 1.27	94.12 ± 0.79	89.85 ± 1.33	91.72 ± 0.83	94.44 ± 1.00	**96.12 ± 0.86**
12	70.10 ± 2.45	57.92 ± 2.44	89.68 ± 1.38	92.47 ± 1.05	92.46 ± 0.97	90.97 ± 1.08	94.92 ± 0.85	92.53 ± 1.06	92.43 ± 0.79	95.69 ± 0.74	**96.84 ± 0.77**
16	72.62 ± 1.88	60.62 ± 2.62	91.09 ± 1.34	93.62 ± 0.93	93.98 ± 0.92	92.90 ± 1.15	95.49 ± 0.75	93.90 ± 0.98	95.11 ± 0.81	96.39 ± 0.68	**97.47 ± 0.59**
20	74.48 ± 1.84	63.93 ± 2.62	92.30 ± 1.09	94.74 ± 0.79	95.20 ± 0.86	94.35 ± 1.24	95.97 ± 0.71	95.58 ± 1.01	95.87 ± 0.75	96.96 ± 0.63	**98.09 ± 0.64**

Figure 5: Example face images of two subjects from the PIE data set.

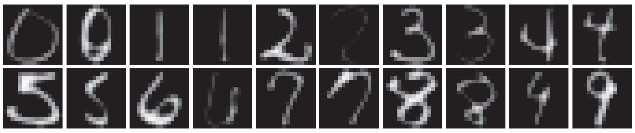


Figure 6: Example digit images from the USPS data set.

In this experiment, 4, 8, 12, 16, and 20 images per subject are randomly selected as labeled samples and the remaining images are regarded as unlabeled samples. The random indices are kept the same for all compared algorithms. Table 3 shows the classification accuracies of different algorithms with different numbers of labeled samples on the Extended Yale B data set. We can easily find that, with increasing number of labeled samples, all algorithms can obtain better classification results. Although the results are close to being saturated, MKLRR still can make some improvements. For example, it gets the accuracy of 98.09% when given 20 labeled images per person, which is 1.13% higher than that of KLRR. In particular, when given small number of labeled samples, MKLRR shows great superiority to the remaining algorithms. There is about 3% improvement when comparing MKLRR with KLRR when the number of labeled samples of each person is only 4. Since there are some noises in this data set, the performance of the basic KNN algorithm greatly decreases.

4.5. Experiment on PIE Data Set. The CMU PIE data set contains 41368 images of 68 subjects with different poses, illumination, and expression. We only use their images in five near frontal poses (C05, C07, C09, C27, and C29) and under different illumination and expressions. The first 15 subjects are selected and there are 2550 face images in total. Each image is manually cropped and resized to 32 × 32 pixels. Figure 5 shows some images of two subjects from the PIE data set.

Identical to the Extended Yale B data set, we also select 4, 8, 12, 16, and 20 images per subject as labeled samples and let the remaining images be unlabeled samples. The random indices are kept the same for all compared algorithms. Table 4 shows the classification accuracies of different algorithms with different numbers of labeled samples on the PIE data set. It is obvious that MKLRR outperforms the other algorithms in all cases. Particularly, MKLRR performs much better than the others when given 4 labeled samples per person.

4.6. Experiment on USPS Data Set. The USPS digit database [44] consists of 9298 handwritten digit images of 10 numbers (0–9). The size of each image is 16 × 16 pixels. We select 200 samples from each class and thus the resultant data set has 2000 images in total. Figure 6 shows some images of the 10 numbers from the USPS data set.

In this experiment, we randomly select 10%, 20%, 30%, 40%, and 50% samples per digit as labeled samples and let the remaining images be unlabeled samples. The random indices are kept the same for all compared algorithms. Table 5 shows the classification accuracies of different algorithms with different numbers of labeled samples on the USPS data set. All algorithms can obtain excellent performance on this data set including the simple KNN algorithm. The classification accuracies of MKLRR are higher than other algorithms in most cases.

4.7. Parameter Sensitivity Analysis. There are two important parameters in MKLRR, which are the regularization parameter γ to construct the manifold adaptive kernel and λ to control the impact of the noise term. It is obvious that MKLRR will boil down to KLRR when we set the parameter γ as zero. In our previous experiments, the usage we employ is to empirically fix the value of γ to one that follows the similar ideas in [43, 45]. In this section, we will analyze the parameter sensitivity of γ and λ by the way of investigating one while fixing the other.

Table 4: Classification accuracies (%) of different algorithms with different number of labeled samples on the PIE data set.

# labeled	kNN-1	kNN-2	LNP	ℓ_1-graph	SPG
4	56.84 ± 3.03	55.42 ± 2.63	78.10 ± 2.09	78.79 ± 2.55	84.71 ± 3.15
8	71.01 ± 2.99	67.18 ± 2.72	86.74 ± 1.48	88.85 ± 1.52	91.31 ± 1.63
12	76.93 ± 2.60	71.15 ± 1.46	89.68 ± 1.38	91.96 ± 1.59	93.82 ± 1.21
16	81.25 ± 1.41	74.67 ± 1.57	91.09 ± 1.34	93.58 ± 1.21	95.03 ± 1.06
20	83.89 ± 1.77	77.18 ± 2.36	92.30 ± 1.09	94.61 ± 1.11	95.84 ± 0.73
LRR	NNLRS	GLRR	LRCB	KLRR	MKLRR
73.00 ± 2.45	86.82 ± 2.30	84.59 ± 1.69	84.81 ± 2.29	86.01 ± 2.43	**88.31 ± 2.28**
86.83 ± 1.54	92.25 ± 1.20	92.46 ± 0.80	90.79 ± 1.16	91.32 ± 1.01	**93.68 ± 0.93**
91.69 ± 0.83	94.44 ± 0.95	94.37 ± 0.39	92.94 ± 0.76	93.62 ± 1.13	**95.81 ± 0.64**
94.21 ± 0.72	95.53 ± 0.73	95.69 ± 0.52	94.72 ± 0.58	94.91 ± 0.95	**96.41 ± 0.42**
95.39 ± 0.51	96.13 ± 0.50	96.08 ± 0.48	96.08 ± 0.41	95.54 ± 0.65	**96.98 ± 0.57**

Table 5: Classification accuracies (%) of different algorithms with different number of labeled samples on the USPS data set.

# labeled	kNN-1	kNN-2	LNP	ℓ_1-graph	SPG
20	**94.33 ± 0.35**	93.16 ± 0.33	94.31 ± 0.61	92.95 ± 0.79	92.89 ± 0.45
40	94.64 ± 0.35	93.69 ± 0.43	95.26 ± 0.54	94.54 ± 0.69	94.76 ± 0.60
60	94.70 ± 0.45	93.74 ± 0.40	95.64 ± 0.44	95.37 ± 0.51	95.54 ± 0.60
80	94.92 ± 0.45	93.99 ± 0.43	96.02 ± 0.43	95.90 ± 0.53	96.18 ± 0.72
100	94.98 ± 0.52	93.99 ± 0.39	96.10 ± 0.43	96.35 ± 0.63	96.43 ± 0.68
LRR	NNLRS	GLRR	LRCB	KLRR	MKLRR
86.96 ± 0.78	92.09 ± 0.67	90.49 ± 0.77	90.23 ± 0.81	90.50 ± 0.90	93.61 ± 0.58
88.64 ± 0.55	93.58 ± 0.45	91.53 ± 0.97	92.38 ± 0.57	93.17 ± 0.83	**95.60 ± 0.49**
88.96 ± 0.72	94.04 ± 0.57	91.84 ± 0.83	93.67 ± 0.61	94.12 ± 0.60	**96.13 ± 0.51**
90.08 ± 0.41	94.34 ± 0.56	92.34 ± 0.77	94.51 ± 0.59	94.90 ± 0.59	**96.51 ± 0.57**
90.42 ± 0.83	94.67 ± 0.63	92.49 ± 0.89	94.77 ± 0.43	95.19 ± 0.68	**96.87 ± 0.37**

Figure 7 shows how the performance of MKLRR varies with the change of γ on the Extended Yale B and PIE data sets, respectively, where we fix $\lambda = 1$. Here, four images per person are labeled and the remaining are unlabeled. For making it easier to do comparison, we also include the results of KLRR in the figure. We can find that $[10^{-2}, 10]$ is a reasonable interval for the selection of values of γ.

Figure 8 shows how the performance of MKLRR varies with the change of λ on the Extended Yale B and PIE data sets, respectively, where we fix $\gamma = 1$. There are also four labeled samples for each person. Generally, MKLRR is insensitive to the variation of λ if it is set as a slightly large value.

For the remaining data sets, the parameter sensitivities of MKLRR with respect to γ and λ have similar tendencies as shown in Figures 7 and 8.

5. Conclusion and Future Work

In this paper, we have proposed a new low-rank representation model for semisupervised image classification, which is called manifold adaptive kernel low-rank representation (MKLRR). Different from most existing LRR variants that consider the structure information in the original data space, our proposed model explicitly takes the intrinsic manifold structure depicted by nearest-neighbor graph into consideration. The graph Laplacian corresponding to the local geometry of the data is incorporated into the manifold adaptive kernel space, in which the low rank representation model is then calculated. Extensive experiments performed on both synthetic and benchmark data sets have shown excellent performance of MKLRR-based graph for semisupervised classification when given limited labeled samples.

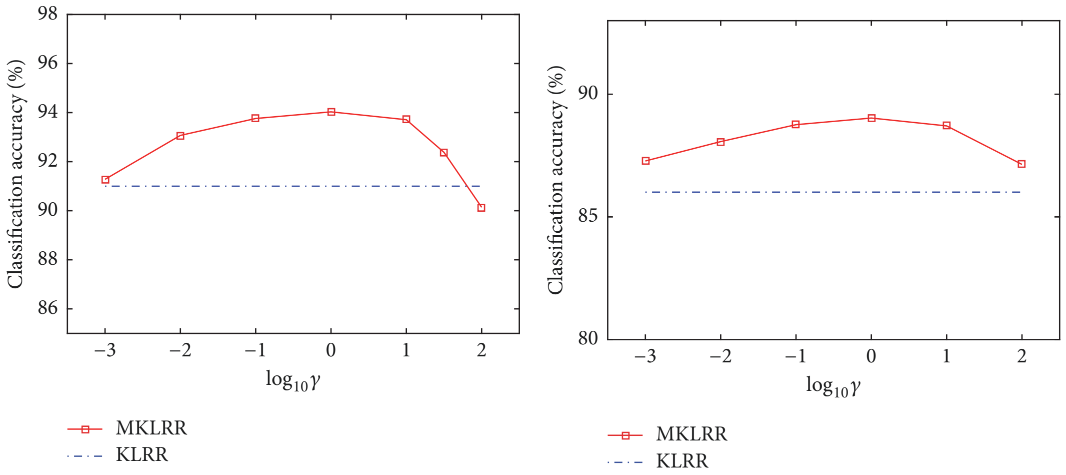
As a limitation of general two-stage graph-based semisupervised learning methods, the information of labeled samples is neglected in graph construction stage. Therefore, it is necessary to take this point into consideration in order to construct more discriminative graph. This will be our future work and one of possible approaches is to introduce a constraint matrix that can depict the partial label information of data into LRR model.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

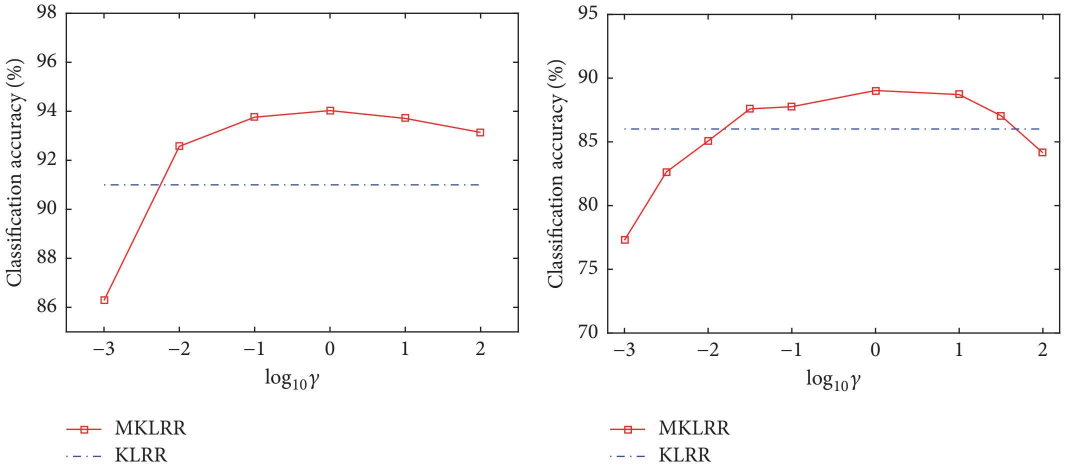
This work was partially supported by National Natural Science Foundation of China (61602140, 61971193, 61633010,



(a) Extended Yale B

(b) PIE

Figure 7: The performance of MKLRR versus parameter γ by fixing $\lambda = 1$.



(a) Extended Yale B

(b) PIE

Figure 8: The performance of MKLRR versus parameter λ by fixing $\gamma = 1$.

61473110, and 61502129), Zhejiang Science and Technology Program (2017C33049, 2018C04012, and LQ16F020004), China Postdoctoral Science Foundation (2017M620470), Jiangsu Key Laboratory of Big Data Security & Intelligent Processing, NJUPT (BDSIP201804), and Co-Innovation Center for Information Supply & Assurance Technology, Anhui University (ADXXBZ201704).

References

[1] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proceedings of the 20th International Conference on Machine Learning (ICML '03)*, vol. 3, pp. 912–919, Washington, DC, USA, August 2003.

[2] X. Zhu, "Semi-supervised learning literature survey," Computer Sciences Technical Report, University of Wisconsin, Madison, WI, USA.

[3] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," in *Proceedings of the 11th IEEE International Conference on Computer Vision (ICCV '07)*, pp. 1–7, Rio de Janeiro, Brazil, October 2007.

[4] Z.-H. Zhou and M. Li, "Semi-supervised learning by disagreement," *Knowledge and Information Systems*, vol. 24, no. 3, pp. 415–439, 2010.

[5] Y. Wang and S. Chen, "Safety-aware semi-supervised classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 11, pp. 1763–1772, 2013.

[6] Z. Jiang, S. Zhang, and J. Zeng, "A hybrid generative/discriminative method for semi-supervised classification," *Knowledge-Based Systems*, vol. 37, pp. 137–145, 2013.

[7] G. Forestier and C. Wemmert, "Semi-supervised learning using multiple clusterings with limited labeled data," *Information Sciences*, vol. 361-362, pp. 48–65, 2016.

[8] S. Yan and H. Wang, "Semi-supervised learning by sparse representation," in *Proceedings of the SIAM International Conference on Data Mining*, pp. 792–801, 2009.

[9] L. Zhuang, H. Gao, Z. Lin, Y. Ma, X. Zhang, and N. Yu, "Nonnegative low rank and sparse graph for semi-supervised learning," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2328–2335, June 2012.

[10] Y. Zheng, X. Zhang, S. Yang, and L. Jiao, "Low-rank representation with local constraint for graph construction," *Neurocomputing*, vol. 122, pp. 398–405, 2013.

[11] S. Yang, Z. Feng, Y. Ren, H. Liu, and L. Jiao, "Semi-supervised classification via kernel low-rank representation graph," *Knowledge-Based Systems*, vol. 69, no. 1, pp. 150–158, 2014.

[12] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, 2010.

[13] Y. Peng and B.-L. Lu, "Robust structured sparse representation via half-quadratic optimization for face recognition," *Multimedia Tools and Applications*, vol. 76, no. 6, pp. 8859–8880, 2017.

[14] Y. Peng and B.-L. Lu, "Discriminative extreme learning machine with supervised sparsity preserving for image classification," *Neurocomputing*, vol. 261, pp. 242–252, 2017.

[15] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proceedings of the 27th International Conference on Machine Learning (ICML '10)*, pp. 663–670, June 2010.

[16] W. Jiang, J. Liu, H. Qi, and Q. Dai, "Robust subspace segmentation via nonconvex low rank representation," *Information Sciences*, vol. 340/341, pp. 144–158, 2016.

[17] X. Lu, Y. Wang, and Y. Yuan, "Graph-regularized low-rank representation for destriping of hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 7, pp. 4009–4018, 2013.

[18] Y. Peng, B.-L. Lu, and S. Wang, "Enhanced low-rank representation via sparse manifold adaption for semi-supervised learning," *Neural Networks*, vol. 65, pp. 1–17, 2015.

[19] Y. Peng, X. Long, and B.-L. Lu, "Graph Based Semi-Supervised Learning via Structure Preserving Low-Rank Representation," *Neural Processing Letters*, vol. 41, no. 3, pp. 389–406, 2015.

[20] S. Li and Y. Fu, "Low-rank coding with b-matching constraint for semi-supervised classification," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI '13)*, pp. 1472–1478, Beijing, China, August 2013.

[21] H. Nguyen, W. Yang, F. Shen, and C. Sun, "Kernel Low-Rank Representation for face recognition," *Neurocomputing*, vol. 155, pp. 32–42, 2015.

[22] M. Zheng, J. Bu, C. Chen et al., "Graph regularized sparse coding for image representation," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1327–1336, 2011.

[23] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, 2011.

[24] D. Cai, X. He, and J. Han, "Locally consistent concept factorization for document clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 6, pp. 902–913, 2011.

[25] X. F. He, D. Cai, Y. L. Shao, H. Bao, and J. Han, "Laplacian regularized Gaussian mixture model for data clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 9, pp. 1406–1418, 2011.

[26] Y. Peng, S. Wang, X. Long, and B.-L. Lu, "Discriminative graph regularized extreme learning machine and its application to face recognition," *Neurocomputing*, pp. 340–353, 2015.

[27] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, pp. 1735–1742, June 2006.

[28] V. Sindhwani, P. Niyogi, and M. Belkin, "Beyond the point cloud: from transductive to semi-supervised learning," in *Proceedings of the 22nd International Conference on Machine Learning (ICML '05)*, pp. 825–832, August 2005.

[29] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, 2011.

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

[31] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," https://arxiv.org/abs/1009.5055.

[32] Y. Peng, S. Wang, and B.-L. Lu, "Structure preserving low-rank representation for semi-supervised face recognition," in *Proceedings of the International Conference on Neural Information Processing*, pp. 148–155, 2013.

[33] B. Schölkopf and A. J. Smola, *Learning with kernels*, The MIT Press, 2002.

[34] M. Belkin, V. Sindhwani, and P. Niyogi, "Manifold regularization: a geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.

[35] P. Niyogi, S. Smale, and S. Weinberger, "Finding the homology of submanifolds with high confidence from random samples," *Discrete & Computational Geometry*, vol. 39, no. 1-3, pp. 419–441, 2008.

[36] F. R. Chung, *Spectral graph theory*, vol. 92, AMS Bookstore, 1997.

[37] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2013.

[38] B. Cheng, J. Yang, S. Yan, Y. Fu, and T. S. Huang, "Learning with-graph for image analysis," *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 858–866, 2010.

[39] K. Koh, S. Kim, and S. Boyd, *l1_ls: A Matlab Solver for Large-Scale l1-Regularized Least Squares Problems*, Stanford University.

[40] F. Wang and C. Zhang, "Label propagation through linear neighborhoods," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 1, pp. 55–67, 2008.

[41] R. He, W.-S. Zheng, B.-G. Hu, and X.-W. Kong, "Nonnegative sparse coding for discriminative semi-supervised learning," in *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011*, pp. 2849–2856, USA, June 2011.

[42] S. Ji and J. Ye, "An accelerated gradient method for trace norm minimization," in *Proceedings of the 26th International Conference On Machine Learning (ICML '09)*, pp. 457–464, ACM, June 2009.

[43] D. Cai and X. He, "Manifold adaptive experimental design for text categorization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 4, pp. 707–719, 2012.

[44] J. J. Hull, "A database for handwritten text recognition research," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 550–554, 1994.

[45] P. Li, J. Bu, L. Zhang, and C. Chen, "Graph-based local concept coordinate factorization," *Knowledge and Information Systems*, vol. 43, no. 1, pp. 103–126, 2015.