Neural Networks 65 (2015) 1–17

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Enhanced low-rank representation via sparse manifold adaption for semi-supervised learning

Yong Peng [a], Bao-Liang Lu [a,b,*], Suhang Wang [c]

[a] Center for Brain-like Computing and Machine Intelligence, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

[b] Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

[c] Department of Computer Science and Engineering, Arizona State University, Tempe, AZ 85281, USA

ARTICLE INFO

Article history:
Received 3 December 2013
Received in revised form 2 December 2014
Accepted 4 January 2015
Available online 10 January 2015

Keywords:
Low-rank representation
Sparse manifold adaption
Graph construction
Semi-supervised learning
Face recognition

ABSTRACT

Constructing an informative and discriminative graph plays an important role in various pattern recognition tasks such as clustering and classification. Among the existing graph-based learning models, low-rank representation (LRR) is a very competitive one, which has been extensively employed in spectral clustering and semi-supervised learning (SSL). In SSL, the graph is composed of both labeled and unlabeled samples, where the edge weights are calculated based on the LRR coefficients. However, most of existing LRR related approaches fail to consider the geometrical structure of data, which has been shown beneficial for discriminative tasks. In this paper, we propose an enhanced LRR via sparse manifold adaption, termed manifold low-rank representation (MLRR), to learn low-rank data representation. MLRR can explicitly take the data local manifold structure into consideration, which can be identified by the geometric sparsity idea; specifically, the local tangent space of each data point was sought by solving a sparse representation objective. Therefore, the graph to depict the relationship of data points can be built once the manifold information is obtained. We incorporate a regularizer into LRR to make the learned coefficients preserve the geometric constraints revealed in the data space. As a result, MLRR combines both the global information emphasized by low-rank property and the local information emphasized by the identified manifold structure. Extensive experimental results on semi-supervised classification tasks demonstrate that MLRR is an excellent method in comparison with several state-of-the-art graph construction approaches.

1. Introduction

For many machine learning and pattern recognition applications, we often have no sufficient labeled samples, which are usually hard or expensive to acquire. However, unlabeled samples are easier to obtain via the Internet for some applications. For simultaneously utilizing both limited labeled samples and many unlabeled samples, SSL has received increasing attention in learning-based applications. SSL algorithms usually make use of the smoothness, cluster, and manifold assumptions, which can be roughly categorized into four groups: generative models, low-density separation models, heuristic models, and graph-based models. Nie et al. presented a semi-supervised orthogonal discriminant analysis algorithm via label propagation by solving the orthogonal constrained trace ratio optimization problem (Nie, Xiang, Jia, & Zhang, 2009). Yu et al. proposed a two stage method in which an unsupervised basis learning phase was followed by a supervised function learning, for SSL on high dimensional nonlinear manifolds (Yu, Zhang, & Gong, 2009). A unified framework for semi-supervised and unsupervised dimensionality reduction was proposed in Nie, Xu, Tsang, and Zhang (2010). A SSL framework, termed flexible manifold embedding, considers the manifold structure of both labeled and unlabeled samples. Karasuyama et al. designed a parameterized similarity function to define the graph edge weights (Karasuyama & Mamitsuka, 2013), which represent both similarity and local representation weight simultaneously. A detailed review of recent work on SSL can be found in Zhu (2008). In this paper, we focus our work on graph-based SSL due to its excellent performance in practice.

* Corresponding author at: Center for Brain-like Computing and Machine Intelligence, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.
E-mail addresses: stany.peng@gmail.com (Y. Peng), bllu@sjtu.edu.cn (B.-L. Lu).

http://dx.doi.org/10.1016/j.neunet.2015.01.001
0893-6080/

Graph-based SSL relies on using a graph $G = (V, E, W)$ to represent the data structure, where V is a set of vertices and each vertex represents a data point, $E \subseteq V \times V$ is a set of edges connecting related vertices, and W is an adjacency matrix measuring the pairwise weights between vertices. Generally, the graph is constructed by using the relationship of domain knowledge or similarity of samples. Once the graph is constructed, each sample spreads its label information to neighbors over the graph until a global stable state is achieved on the whole data set. Thus, both labeled and unlabeled samples remarkably affect the graph construction. How to construct a good graph for representing data structure is critical for graph-based SSL. Recently, some graphs have been well investigated, such as k nearest neighbors (KNN) graph, local linearly embedding (LLE)-based graph (Roweis & Saul, 2000), graph for label propagation based on linear neighborhoods (LNP) (Wang & Zhang, 2008), sparse representation-based graphs (Lu, Zhou, Tan, Shang, & Zhou, 2012; Yan & Wang, 2009), and sparse probability graph (SPG) (He, Zheng, Hu, & Kong, 2011).

Sparse representation-based graph is motivated by that each datum can be reconstructed by the sparse linear superposition of other data points (Cheng, Yang, Yan, Fu, & Huang, 2010) and the sparse reconstruction coefficients are derived by solving an ℓ_1-norm regularized least square optimization problem. Unlike sparse representation which enforces the representation coefficients to be sparse, the recently proposed LRR can obtain a low-rank coefficient by solving a rank minimization problem. LRR has been widely used for various applications such as subspace segmentation (Liu, Lin, & Yu, 2010; Luo, Nie, Ding, & Huang, 2011), face recognition (Chen, Wei, & Wang, 2012) and multitask learning (Chen, Zhou, & Ye, 2011). The graph constructed by LRR can be used for many learning tasks such as spectral clustering (Liu et al., 2010) and SSL (Yang, Wang, Wang, Han, & Jiao, 2013). Several improved models have been proposed to alleviate the drawbacks of the original LRR algorithm on SSL. Non-negative low-rank and sparse (NNLRS) graph (Zhuang, Gao, Lin, Ma, Zhang and Yu, 2012) was proposed by imposing the non-negative and sparse constraints on the low-rank representation coefficient. Zheng et al. presented an algorithm to construct the graph based on low-rank representation with local constraint (LRRLC) (Zheng, Zhang, Jia, Zhao, Guo, Fu and Yu, 2013) in which the local structure is preserved by a locally constrained regularization and the global structure is preserved by LRR. A graph regularization term was added on the LRR objective and the graph regularized low-rank representation (GLRR) model was formulated for destriping of hyperspectral images in Lu, Wang, and Yuan (2013).

Recently, researchers have considered the case when data is drawn from sampling a probability distribution that has support on or near a submanifold of an ambient space. Here, a d-dimensional submanifold of an Euclidean space $\mathbb{R}^M$ is a subset $\mathcal{M}^d \subset \mathbb{R}^M$, which locally looks like a flat d-dimensional Euclidean space (Lee, 2012). It has been shown that learning performance can be significantly enhanced if the underlying manifold structure can be properly identified (Cai, He, & Han, 2011; Cai, He, Han, & Huang, 2011; Zheng, Bu, Chen, Wang, Zhang, Qiu and Cai, 2011).

Motivated by the recent progress on LRR and manifold learning, we propose a novel manifold low-rank representation model to build graph for semi-supervised classification. The basic motivation behind MLRR is to explicitly combine the global and local geometrical structure of data together in graph construction. In MLRR, the global structure is considered by the low-rank property and the local structure is emphasized by the manifold identification. Different from LRRLC and GLRR, which identify the manifold based on the Euclidean distance between data pairs, MLRR adopts the geometric sparsity idea (Elhamifar & Vidal, 2011) to approximately seek the tangent space of each data point. Here the multiple manifolds underlying the data set are assumed to be composed of many local tangent spaces (Zhang & Zha, 2004). We incorporate a regularizer into the LRR objective, aiming at enforcing the low-rank coefficients to preserve the identified manifold structure of data. Similar to NNLRS (Zhuang et al., 2012), we also constrain the representation coefficients to be sparse and non-negative. By properly identifying the manifold structure, MLRR can obtain excellent experimental results in comparison with several LRR variants and other state-of-the-art approaches.

The remainder of this paper is organized as follows. We review the original LRR, several related LRR variants and optimization method in Section 2. In Section 3, we present the formulation of proposed manifold low-rank representation model and its implementation by linearized alternating direction method with adaptive penalty (LADMAP) method (Lin, Liu, & Su, 2011). Experiments on three widely used face data sets and one voice data set to evaluate the performance of MLRR are illustrated in Section 4. Section 5 concludes the paper.

2. Related work

In this section, we review the following three parts: LRR model as well as its several variants, the LADMAP method (Lin et al., 2011) which is often employed to implement the LRR model, and the semi-supervised classification framework used in this paper.

2.1. Low-rank representation and its several variants

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ be a set of n samples in the d-dimensional space. Low-rank representations aim at representing each sample by a linear combination of the bases in $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m] \in \mathbb{R}^{d\times m}$ as $\mathbf{X} = \mathbf{AZ}$, where $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n]$ is the matrix with each $\mathbf{z}_i$ being the representation coefficient of sample $\mathbf{x}_i$. Each element in $\mathbf{z}_i$ can be seen as the contribution to the reconstruction of $\mathbf{x}_i$ with $\mathbf{A}$ as the basis. LRR seeks the lowest-rank solution by solving the following optimization problem (Liu et al., 2010)

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

Due to the NP-hard nature of the *rank* function, the above optimization problem can be relaxed to the following convex optimization problem (Candès, Li, Ma and Wright, 2011)

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

where $\|\cdot\|_*$ denotes the trace norm of a matrix (Cai, Candès, & Shen, 2010), i.e., the sum of its singular values. Considering the fact that samples are usually noisy or even grossly corrupted, a more reasonable objective for LRR can be expressed as

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

where the $\ell_{2,1}$-norm is defined as $\|\mathbf{E}\|_{2,1} = \sum_{j=1}^{n}\sqrt{\sum_{i=1}^{d} e_{ij}^2}$ and parameter λ is used to balance the effect of low-rank term and error term. Some existing studies also used the ℓ_1-norm to measure the error term (Liu & Yan, 2012; Okutomi, Yan, Sugimoto, Liu and Zheng, 2012; Peng, Wang, Wang, & Lu, 2013) while the $\ell_{2,1}$-norm is used in this paper. The optimal solution $\mathbf{Z}^*$ to problem (3) can be obtained via the inexact augmented Lagrange multiplier (ALM) method (Lin, Chen, & Ma, 2010).

As described in Liu et al. (2010), LRR jointly obtains the representation of all the data under a global low-rank constraint, and thus is good at capturing the global structure. Moreover, since each sample can be used to represent itself, there always exist feasible solutions even when the data sampling is insufficient, which is different from sparse representation. These properties make LRR-graph a good candidate for various learning tasks. Below are several recently proposed LRR variants for graph based SSL.

- GLRR-graph (Lu and Wang et al., 2013). GLRR was formulated by incorporating a graph regularizer into LRR objective, minimizing the following objective

$$\min_{\mathbf{Z},\mathbf{E}} \ \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \beta \mathrm{Tr}(\mathbf{Z}\mathbf{L}\mathbf{Z}^T), \quad s.t.\ \mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{E}, \tag{4}$$

  where $\mathbf{L}$ is the graph Laplacian. This model emphasizes the local consistency involved in data, which intuitively encourages that, if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are close in the intrinsic manifold, the corresponding representations $\mathbf{z}_i$ and $\mathbf{z}_j$ are also similar.
- LRRLC-graph (Zheng, Zhang, Yang, & Jiao, 2013). LRRLC imposes the local constraint on the representation coefficients as

$$\min_{\mathbf{Z},\mathbf{E}} \ \|\mathbf{Z}\|_* + \lambda_1\|\mathbf{E}\|_{2,1} + \lambda_2 \sum_{i,j=1}^{n} \|\mathbf{x}_i - \mathbf{x}_j\|^2 |z_{ij}|, \quad s.t.\ \mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{E}, \tag{5}$$

  where the dictionary $\mathbf{A}$ is the data matrix $\mathbf{X}$ itself. The regularizer is induced based on the locality assumption that similar samples should have similar coefficients. Thus, LRRLC is declared to capture both the global structure by LRR and the local structure by the locally constrained regularizer.
- NNLRS-graph (Zhuang et al., 2012). Based on the assumption that an informative graph should reveal both the true intrinsic complexity and certain global structure of data, NNLRS introduces the non-negativity and sparsity constraints on the representation coefficients as

$$\min_{\mathbf{Z},\mathbf{E}} \ \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \beta\|\mathbf{Z}\|_1, \quad s.t.\ \mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{E},\ \mathbf{Z} \geq 0. \tag{6}$$

  The obtained NNLRS-graph can capture both the global mixture of subspaces structure by the low rankness and the locally linear structure by the sparseness of data.

Though these LRR variants show excellent performance in respective applications (GLRR for remote sensing image denoising, LRRLC and NNLRS for semi-supervised classification), LRR still leaves room for improvement. Here we only focus on the improvement from manifold preserving perspective. GLRR uses graph Laplacian to enforce the learned low-rank representation coefficients to vary smoothly along the data manifold. The geometrical structure in GLRR is reflected by the affinity matrix which is built on the basis of 'HeatKernel' function in Euclidean space. Thus it can be predefined given the data set. LRRLC is also using the graph Laplacian-based formula. Different from GLRR, LRRLC needs not to calculate the affinity matrix beforehand, which uses the low-rank representation coefficient directly. Specifically, z_{ij} is used to measure the closeness of samples $\mathbf{x}_i$ and $\mathbf{x}_j$. However, as described in Liu et al. (2010), Liu et al. (2013) and Zhuang et al. (2012), LRR depicts the global structure of data. Therefore, it is unclear whether z_{ij} can accurately reflect the local structure. NNLRS neglects to explicitly consider the manifold information though the sparseness constraint could capture locally linear structure to some extent.

2.2. Linearized alternating direction method with adaptive penalty

Considering the following linearly constrained convex optimization problem,

$$\min_{\mathbf{x},\mathbf{y}} \ f(\mathbf{x}) + g(\mathbf{y}), \quad s.t.\ \mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}) = \mathbf{c}, \tag{7}$$

where $\mathbf{x}, \mathbf{y}, \mathbf{c}$ are vectors or matrices, f, g are convex functions, and $\mathcal{A}, \mathcal{B}$ are linear mappings.

The alternating direction method (ADM) for problem (7) works on the following augmented Lagrangian function

$$\mathcal{L}(\mathbf{x}, \mathbf{y}, \lambda) = f(\mathbf{x}) + g(\mathbf{y}) + \langle \lambda, \mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}) - \mathbf{c}\rangle + \frac{\beta}{2}\|\mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}) - \mathbf{c}\|^2. \tag{8}$$

ADM decomposes the minimization of $\mathcal{L}$ into two subproblems w.r.t. variables $\mathbf{x}$ and $\mathbf{y}$, respectively. Specifically, the updating rules are as follows,

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}} \ \mathcal{L}(\mathbf{x}, \mathbf{y}_k, \lambda_k) = \arg\min_{\mathbf{x}} \ f(\mathbf{x}) + \frac{\beta}{2}\|\mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c} + \lambda_k/\beta\|^2, \tag{9}$$

$$\mathbf{y}_{k+1} = \arg\min_{\mathbf{y}} \ \mathcal{L}(\mathbf{x}_{k+1}, \mathbf{y}, \lambda_k) = \arg\min_{\mathbf{y}} \ g(\mathbf{y}) + \frac{\beta}{2}\|\mathcal{A}(\mathbf{x}_{k+1}) + \mathcal{B}(\mathbf{y}) - \mathbf{c} + \lambda_k/\beta\|^2, \tag{10}$$

$$\lambda_{k+1} = \lambda_k + \beta[\mathcal{A}(\mathbf{x}_{k+1}) + \mathcal{B}(\mathbf{y}_{k+1}) - \mathbf{c}]. \tag{11}$$

If no closed form solutions to (9) and (10), auxiliary variables are often introduced to iteratively optimize them. To avoid introducing auxiliary variables and solve these two subproblems efficiently, LADMAP was proposed on the basis of the following two techniques: (1) linearization of (9) and (10), and (2) adaptively updating the penalty parameter. In LADMAP, the updating rules are reformulated as

$$\begin{aligned}\mathbf{x}_{k+1} &= \arg\min_{\mathbf{x}} \ f(\mathbf{x}) + \langle \mathcal{A}^*(\lambda_k) + \beta\mathcal{A}^*(\mathcal{A}(\mathbf{x}_k) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c}), \mathbf{x} - \mathbf{x}_k\rangle + \frac{\beta\eta_1}{2}\|\mathbf{x} - \mathbf{x}_k\|^2 \\ &= \arg\min_{\mathbf{x}} \ f(\mathbf{x}) + \frac{\beta\eta_1}{2}\|\mathbf{x} - \mathbf{x}_k + \mathcal{A}^*(\lambda_k + \beta(\mathcal{A}(\mathbf{x}_k) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c}))/(\beta\eta_1)\|^2, \end{aligned} \tag{12}$$

$$\mathbf{y}_{k+1} = \arg\min_{\mathbf{y}} \ g(\mathbf{y}) + \frac{\beta\eta_2}{2}\|\mathbf{y} - \mathbf{y}_k + \mathcal{B}^*(\lambda_k + \beta(\mathcal{A}(\mathbf{x}_{k+1}) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c}))/(\beta\eta_2)\|^2,$$

where $\mathcal{A}^*$ is the adjoint of $\mathcal{A}$ and η_1, η_2 are related parameters. The parameter β, which is fixed in ADM, is adaptively updated as

$$\beta_{k+1} = \min(\beta_{\max}, \rho\beta_k), \tag{13}$$

$$\rho = \begin{cases} \rho_0, & \text{if } \beta_k \cdot \max(\sqrt{\eta_1}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|, \\ & \sqrt{\eta_2}\|\mathbf{y}_{k+1} - \mathbf{y}_k\|)/\|\mathbf{c}\| < \varepsilon_2, \\ 1, & \text{otherwise}, \end{cases} \tag{14}$$

where $\beta_{\max}$ is an upper bound of $\{\beta_k\}$, and $\rho_0 \geq 1$ is a constant. Detailed explanation of η_1, η_2, and the settings of ρ, β can be found in Lin et al. (2011) and Yang and Yuan (2013).

2.3. Semi-supervised classification framework

Denote $\mathbf{Y} = [\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^n]^T \in \mathbb{R}^{n\times c}$ as the initial label matrix. If $\mathbf{x}_i$ is the unlabeled data, then $\mathbf{y}^i = \mathbf{0}$. If $\mathbf{x}_i$ is labeled data in class k, then the kth entry of $\mathbf{y}^i$ is 1 and the other entries of $\mathbf{y}^i$ are 0. Generally, graph-based SSL models aim to solve the following problem (Zhou, Bousquet, Lal, Weston and Schölkopf, 2004),

$$\min_{\mathbf{Q}} \mathrm{Tr}(\mathbf{Q}^T\tilde{\mathbf{L}}\mathbf{Q}) + \mathrm{Tr}((\mathbf{Q} - \mathbf{Y})^T\boldsymbol{\Psi}(\mathbf{Q} - \mathbf{Y})), \tag{15}$$

where $\mathbf{L}$ is the Laplacian matrix, $\tilde{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$ is the normalized graph Laplacian, $\mathbf{D}$ is the corresponding degree matrix w.r.t. the learned affinity matrix $\mathbf{S}$, $\boldsymbol{\Psi}$ is a diagonal matrix with the ith diagonal element ψ_{ii} to control the impact of the initial label $\mathbf{y}^i$ w.r.t. $\mathbf{x}_i$, and $\mathbf{Q} \in \mathbb{R}^{n\times c}$ is the label matrix to be solved.

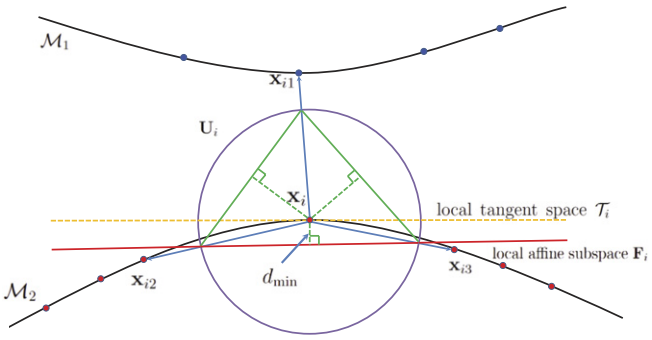


Fig. 1. The illustration of *local tangent space* $\mathcal{T}_i$ and *local affine subspace* $\mathbf{F}_i$ associated with $\mathbf{x}_i$. We aim to find a proper $\mathbf{F}_i$ to approximate the $\mathcal{T}_i$. In this example, there are three local affine subspaces (two of them are shown in green line and one shown in red line) and the best fitted one (shown in red line) is spanned by the two local directions $\frac{\mathbf{x}_{i2}-\mathbf{x}_i}{\|\mathbf{x}_{i2}-\mathbf{x}_i\|}$ and $\frac{\mathbf{x}_{i3}-\mathbf{x}_i}{\|\mathbf{x}_{i3}-\mathbf{x}_i\|}$. Such $\mathbf{F}_i$ holds the minimal distance to $\mathbf{x}_i$ and the minimal angle w.r.t. the local tangent space $\mathcal{T}_i$.
Source: Adapted from Zhang et al. (2013).

Taking the derivative of problem (15) w.r.t. $\mathbf{Q}$ and setting the derivative to zero, we have

$$\mathbf{Q} = (\tilde{\mathbf{L}} + \boldsymbol{\Psi})^{-1}(\boldsymbol{\Psi}\mathbf{Y}). \tag{16}$$

In the following section, we will describe how to learn an informative graph affinity matrix $\mathbf{S}$ by MLRR.

3. Low-rank representation via sparse manifold adaption

In this section, we introduce the newly proposed MLRR model. Firstly, we introduce the principle of identifying the manifold based on spare manifold adaption; secondly, we formulate the objective function of MLRR by incorporating a regularizer into LRR objective function, which aims at enforcing the LRR coefficients to preserve the identified manifold structure of data; finally, we develop the optimization method to MLRR via LADMAP (Lin et al., 2011).

3.1. Sparse manifold adaption

Notations. The affinity matrix used to represent the neighborhood graph is $\mathbf{W} \in \mathbb{R}^{n\times n}$, where w_{ij} represents the edge weight between vertices $\mathbf{x}_i$ and $\mathbf{x}_j$. $\mathcal{M}$ is the manifold underlying the data, $\{\mathcal{T}_i\}_{i=1}^n$ are *local tangent spaces*, $\{\mathbf{U}_i\}_{i=1}^n$ are *local direction basis matrices*, and $\{\mathbf{F}_i\}_{i=1}^n$ are *local affine subspaces*.

For each sample $\mathbf{x}_i, i = 1, 2, \ldots, n$, we want to identify its neighbors on the same manifold rather than the entire Euclidean space. Assuming that samples are sufficient and the manifold is smooth, each data point can be well approximated by a linear combination of a few nearby samples on the same manifold. Based on the idea that the underlying manifolds are equivalent to the local tangent spaces to some extent (Zhang & Zha, 2004), manifold $\mathcal{M}$ can be mathematically written as $\mathcal{M} \triangleq \bigcup_{i=1}^n \mathcal{T}_i$, where each local tangent space $\mathcal{T}_i$ is a small patch of a submanifold from $\mathcal{M}$. Therefore, we need to identify each tangent space, denoted by $\mathcal{T}_i$, which lies around data point $\mathbf{x}_i$. The relationship among the *local manifold*, *local tangent space* $\mathcal{T}_i$, and *local affine space* $\mathbf{F}_i$ is illustrated in Fig. 1.

In this paper, we employ the geometric sparsity idea (Elhamifar & Vidal, 2009, 2011) to seek the local tangent spaces. Similarly, sparse representation was used in Shen and Si (2010) to identify multiple manifolds for non-negative matrix factorization-based spectral clustering (MM-NMF), where each data point $\mathbf{x}_i$ is linearly represented by other data points. Different from MM-NMF, we reconstruct $\mathbf{x}_i$ using sparse bases selected from the local direction basis matrix $\mathbf{U}_i$. The optimal local directions which can determine the optimally fitted local affine subspace $\mathbf{F}_i$ will be selected by minimizing the reconstruction distortion. As a result, we can get a local affine subspace $\mathbf{F}_i$ which can well approximate the local tangent space $\mathcal{T}_i$ and the angle between these two spaces is minimized. Consequently, the submanifold around data point $\mathbf{x}_i$ is identified.

We refer to the above manifold identification method as *sparse manifold adaption* since this process is completed by solving a sparse representation objective. The detailed steps will be given below.

Specifically, we first find N_i neighbors around $\mathbf{x}_i$ based on Euclidean distance (denote the neighbors' indices $[i_1, i_2, \ldots, i_{N_i}]$ of $\mathbf{x}_i$ as $\mathcal{N}_i$) and then select some qualified ones by sparse manifold adaption. To remove the distance variations and preserve only the direction information, we normalize the difference vectors between $\mathbf{x}_i$ and its N_i neighbors. The formed local direction basis matrix $\mathbf{U}_i$ is

$$\mathbf{U}_i = \left[\frac{\mathbf{x}_{i_1}-\mathbf{x}_i}{\|\mathbf{x}_{i_1}-\mathbf{x}_i\|_2}, \frac{\mathbf{x}_{i_2}-\mathbf{x}_i}{\|\mathbf{x}_{i_2}-\mathbf{x}_i\|_2}, \cdots, \frac{\mathbf{x}_{i_{N_i}}-\mathbf{x}_i}{\|\mathbf{x}_{i_{N_i}}-\mathbf{x}_i\|_2}\right]. \tag{17}$$

Then the local affine subspace $\mathbf{F}_i$ can be defined as $\{\mathbf{x}_i + \mathbf{U}_i\mathbf{c}_i | \mathbf{1}^T\mathbf{c}_i = 1, \mathbf{c}_i \in \mathbb{R}^{N_i}\}$. Since sparse manifold adaption aims at finding a proper $\mathbf{F}_i$ which can best fit $\mathbf{x}_i$, we minimize the distortion between $\mathbf{F}_i$ and $\mathbf{x}_i$, which is equivalent to

$$\min_{\mathbf{1}^T\mathbf{c}_i=1} \frac{1}{2}\|\mathbf{U}_i\mathbf{c}_i\|_2^2. \tag{18}$$

Among all solutions which satisfy $\|\mathbf{U}_i\mathbf{c}_i\|_2^2 \leq \theta$ (θ is a small positive value), we impose the ℓ_1-norm sparsity constraint on the coefficient vector $\mathbf{c}_i$ to automatically select local directions from $\mathbf{U}_i$. Therefore, we formulate the sparse representation objective as

$$\min_{\mathbf{c}_i\in\mathbb{R}^{N_i}} \frac{1}{2}\|\mathbf{U}_i\mathbf{c}_i\|_2^2 + \gamma\|\mathbf{c}_i\|_1, \quad s.t.\ \mathbf{1}^T\mathbf{c}_i = 1. \tag{19}$$

As reviewed in Yang, Zhou, Balasubramanian, Sastry and Ma (2013), there are several representative approaches to ℓ_1-norm regularized minimization problem: Gradient Projection, Homotopy, Iterative Shrinkage Thresholding, Proximal Gradient and Augmented Lagrange Multiplier. Problem (19) introduces an extra affine constraint $\mathbf{1}^T\mathbf{c}_i = 1$ on the sparse representation problem (Tibshirani, 1996), which can be efficiently optimized by the ADM method (Elhamifar & Vidal, 2011). The concrete updating rules are shown in Appendix A.

By using sparse manifold adaption, we can obtain the coefficient vectors $\{\mathbf{c}_i\}_{i=1}^n$ and thus a neighborhood graph can be constructed similar to the LLE-graph (Roweis & Saul, 2000). The graph edge weights $\{w_{ij}\}_{i,j=1}^n$ based on $\mathbf{c}_i = [c_{1i}, c_{2i}, \ldots, c_{N_i i}]^T$ are defined as

$$w_{i_j,i} = \frac{c_{ji}}{\|\mathbf{x}_{i_j}-\mathbf{x}_i\|_2} \Big/ \sum_{j'\in\mathcal{N}_i} \frac{c_{j'i}}{\|\mathbf{x}_{i_{j'}}-\mathbf{x}_i\|_2}. \tag{20}$$

Since we only consider the direction information in (17) but neglect the distance information, Eq. (20) can compensate the distance information which makes the neighbors closer to $\mathbf{x}_i$ contribute larger weights.

Now we have obtained the relationship between each point $\mathbf{x}_i$ and its neighbors; thus we can represent $\mathbf{x}_i$ by these neighbors. Neighbors with non-zero coefficients span the local affine subspace $\mathbf{F}_i$, which is a proper approximation to the local tangent space $\mathcal{T}_i$ around $\mathbf{x}_i$, which can be seen as the submanifold where $\mathbf{x}_i$ and these neighbors are drawn from.

Once (19) is solved, we can calculate the graph edge weight w_{ji} using (20). Accordingly, we have the following theorem.

Theorem 1. *For each point $\mathbf{x}_i$, the $w_{i_j,i}$ calculated via (20) satisfies*

$$\|\mathbf{x}_i - \sum_{j\in\mathcal{N}_i} w_{i_j,i}\mathbf{x}_{i_j}\|_2^2 \le \varepsilon, \tag{21}$$

where $\varepsilon = \theta \Big/ \sum_{j\in\mathcal{N}_i} \frac{c_{ji}}{\|\mathbf{x}_{i_j}-\mathbf{x}_i\|_2}$, θ is the resulting error of (19) (e.g., $\|\mathbf{U}_i\mathbf{c}_i\|^2 \le \theta$), and $\mathcal{N}_i$ is the neighbors' index set of $\mathbf{x}_i$.

Proof. According to the definition of $w_{i_j,i}$ in (20), we have Eq. (22) given in Box I. □

3.2. Manifold regularization via sparse manifold adaption

By using the sparse manifold adaption, we obtain the relationship of representing one data point by its neighbors in data space $\mathcal{X}$. Naturally, we hope that this relationship can be preserved in the low-rank coefficient space $\mathcal{Z}$. Now we make an analysis on the connection of data representation between $\mathcal{X}$ and $\mathcal{Z}$.

Theorem 2. *There exists a number $\vartheta \ge 0$, such that the following inequality holds*

$$\begin{aligned}\|\mathbf{x}_i - \sum_{j\in\mathcal{N}_i} w_{ji}\mathbf{x}_j\|_2^2 &= \|(\mathbf{A}\mathbf{z}_i + \mathbf{e}_i) - \sum_{j\in\mathcal{N}_i} w_{ji}(\mathbf{A}\mathbf{z}_j + \mathbf{e}_j)\|_2^2 \\ &\doteq \|\mathbf{A}\mathbf{z}_i - \sum_{j\in\mathcal{N}_i} w_{ji}\mathbf{A}\mathbf{z}_j\|_2^2 \le \vartheta\|\mathbf{z}_i - \sum_{j\in\mathcal{N}_i} w_{ji}\mathbf{z}_j\|_2^2,\end{aligned} \tag{23}$$

where $\sum_{j=1}^{N_i} w_{ji} = 1$, $w_{ji} \ge 0$, $\mathbf{e}_i$ is the ith column of $\mathbf{E}$, and $\mathcal{N}_i$ is the neighbors' index set of $\mathbf{x}_i$.

Proof. We have

$$\begin{aligned}\|\mathbf{A}\mathbf{z}_i - \sum_{j\in\mathcal{N}_i} w_{ji}\mathbf{A}\mathbf{z}_j\|_2^2 &= \|\sum_{j\in\mathcal{N}_i} w_{ji}\mathbf{A}(\mathbf{z}_j - \mathbf{z}_i)\|_2^2 \\ &= \sum_{j,l\in\mathcal{N}_i} w_{ji}w_{li}(\mathbf{z}_j - \mathbf{z}_i)^T\mathbf{A}^T\mathbf{A}(\mathbf{z}_l - \mathbf{z}_i) \\ &\triangleq \sum_{j,l\in\mathcal{N}_i} w_{ji}w_{li}(\mathbf{z}_j - \mathbf{z}_i)^T\mathbf{M}(\mathbf{z}_l - \mathbf{z}_i),\end{aligned} \tag{24}$$

where $\mathbf{M} = \mathbf{A}^T\mathbf{A}$.

Since $\mathbf{M}$ is a real-valued symmetric matrix, its eigenvalue decomposition can be written as $\mathbf{M} = \mathbf{V}\mathbf{\Lambda}_\mathbf{M}\mathbf{V}^T$. Suppose that $\lambda_\mathbf{M}^1$ is the maximal eigenvalue in $\mathbf{\Lambda}_\mathbf{M}$ and thus we have

$$\begin{aligned}\|\mathbf{A}\mathbf{z}_i - \sum_{j\in\mathcal{N}_i} w_{ji}\mathbf{A}\mathbf{z}_j\|_2^2 &= \sum_{j,l\in\mathcal{N}_i} w_{ji}w_{li}(\mathbf{z}_j - \mathbf{z}_i)^T\mathbf{M}(\mathbf{z}_l - \mathbf{z}_i) \\ &\le \lambda_\mathbf{M}^1 \sum_{j,l\in\mathcal{N}_i} w_{ji}w_{li}(\mathbf{z}_j - \mathbf{z}_i)^T(\mathbf{z}_l - \mathbf{z}_i) \\ &= \lambda_\mathbf{M}^1\|\sum_{j\in\mathcal{N}_i} w_{ji}(\mathbf{z}_i - \mathbf{z}_j)\|_2^2 \\ &= \lambda_\mathbf{M}^1\|\mathbf{z}_i - \sum_{j\in\mathcal{N}_i} w_{ij}\mathbf{z}_j\|_2^2.\end{aligned} \tag{25}$$

Let $\vartheta = \lambda_\mathbf{M}^1$, so $\|\mathbf{z}_i - \sum_{j\in\mathcal{N}_i} w_{ji}\mathbf{z}_j\|_2^2$ in the low-rank coefficient space $\mathcal{Z}$ is a tight upper bound of $\|\mathbf{x}_i - \sum_{j\in\mathcal{N}_i} w_{ji}\mathbf{x}_j\|_2^2$ in data space $\mathcal{X}$. □

According to Theorem 2, it is reasonable to use the geometrical structure in data space to constrain the low-rank coefficients. Therefore, we can directly use $\mathbf{W}$, which can be learned in data space $\mathcal{X}$ by sparse manifold adaption, as the neighborhood graph affinity matrix in $\mathcal{Z}$. Specifically, when recovering the data matrix $\mathbf{X}$ within the LRR framework, we expect the learned LRR coefficient matrix $\mathbf{Z}$ to preserve the geometry constraint, which was depicted by the affinity matrix $\mathbf{W}$. To this end, we minimize

$$\begin{aligned}\mathcal{G}(\mathbf{Z}) &= \sum_{i=1}^{n}\|\mathbf{z}_i - \sum_j w_{ji}\mathbf{z}_j\|_2^2 = \|\mathbf{Z} - \mathbf{Z}\mathbf{W}\|_2^2 = \|(\mathbf{Z} - \mathbf{Z}\mathbf{W})^T\|_2^2 \\ &= \mathrm{Tr}(\mathbf{Z}(\mathbf{I} - \mathbf{W})(\mathbf{I} - \mathbf{W})^T\mathbf{Z}^T) = \mathrm{Tr}(\mathbf{Z}\mathbf{G}\mathbf{Z}^T),\end{aligned} \tag{26}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{G} = (\mathbf{I} - \mathbf{W})(\mathbf{I} - \mathbf{W})^T$.

3.3. Manifold low-rank representation model

Existing studies show that an informative graph often satisfies three properties, high discriminative power, sparsity, and adaptive neighborhood (Wright, Ma, Mairal, Sapiro, Huang and Yan, 2010; Zhuang et al., 2012). For inheriting the advantages caused by the sparsity and non-negativity properties, we impose these two constraints on low-rank coefficient matrix as Zhuang et al. (2012). As a result, taking the *manifold information*, *sparsity* and *non-negativity* properties into consideration, we formulate the objective function of MLRR as

$$\begin{aligned}&\min_{\mathbf{Z},\mathbf{E}} \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \alpha\|\mathbf{Z}\|_1 + \beta\mathcal{G}(\mathbf{Z}), \\ &s.t.\ \mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{E},\ \mathbf{Z} \ge 0,\end{aligned} \tag{27}$$

where three parameters, $\lambda > 0, \alpha > 0$ and $\beta > 0$, are respectively used to control the impacts of error term, sparsity and manifold regularizer.

Similar to Liu et al. (2010), we introduce an auxiliary variable $\mathbf{J}$ w.r.t. $\mathbf{Z}$ to make the MLRR objective separable and thus problem (27) can be rewritten as

$$\begin{aligned}&\min_{\mathbf{Z},\mathbf{J},\mathbf{E}} \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \alpha\|\mathbf{J}\|_1 + \beta\mathrm{Tr}(\mathbf{Z}\mathbf{G}\mathbf{Z}^T), \\ &s.t.\ \mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{E},\ \mathbf{Z} = \mathbf{J}, \mathbf{J} \ge 0.\end{aligned} \tag{28}$$

The augmented Lagrangian function $\mathcal{L}$ to problem (28) is

$$\begin{aligned}\mathcal{L}(\mathbf{Z},\mathbf{J},\mathbf{E},\mathbf{Y}_1,\mathbf{Y}_2,\mu) &= \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \alpha\|\mathbf{J}\|_1 + \beta\mathrm{Tr}(\mathbf{Z}\mathbf{G}\mathbf{Z}^T) \\ &\quad + \langle\mathbf{Y}_1, \mathbf{X} - \mathbf{A}\mathbf{Z} - \mathbf{E}\rangle \\ &\quad + \langle\mathbf{Y}_2, \mathbf{Z} - \mathbf{J}\rangle + \frac{\mu}{2}(\|\mathbf{X} - \mathbf{A}\mathbf{Z} - \mathbf{E}\|_F^2 + \|\mathbf{Z} - \mathbf{J}\|_F^2) \\ &= \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \alpha\|\mathbf{J}\|_1 + f(\mathbf{Z},\mathbf{J},\mathbf{E},\mathbf{Y}_1,\mathbf{Y}_2,\mu) \\ &\quad - \frac{1}{2\mu}(\|\mathbf{Y}_1\|_F^2 + \|\mathbf{Y}_2\|_F^2),\end{aligned} \tag{29}$$

where $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are Lagrangian multipliers, $\mu > 0$ is a penalty parameter, $\|\cdot\|_F$ is the Frobenius norm and

$$\begin{aligned}f(\mathbf{Z},\mathbf{J},\mathbf{E},\mathbf{Y}_1,\mathbf{Y}_2,\mu) &= \beta\mathrm{Tr}(\mathbf{Z}\mathbf{G}\mathbf{Z}^T) \\ &\quad + \frac{\mu}{2}\left(\left\|\mathbf{X} - \mathbf{A}\mathbf{Z} - \mathbf{E} + \frac{\mathbf{Y}_1}{\mu}\right\|_F^2 + \left\|\mathbf{Z} - \mathbf{J} + \frac{\mathbf{Y}_2}{\mu}\right\|_F^2\right).\end{aligned} \tag{30}$$

The LADMAP (Lin et al., 2011) is used to obtain the updating rules of variables $\mathbf{Z}$, $\mathbf{J}$ and $\mathbf{E}$, alternately. Specifically, we minimize problem (29) w.r.t. each variable while fixing the others. By some linear algebraic transformations, the updating rules are as follows.

- Update $\mathbf{Z}$ with other variables fixed.

$$\begin{aligned}\mathbf{Z}_{k+1} &= \arg\min_\mathbf{Z} \|\mathbf{Z}\|_* + \langle\nabla_\mathbf{Z} f(\mathbf{Z}_k,\mathbf{J}_k,\mathbf{E}_k,\mathbf{Y}_{1,k},\mathbf{Y}_{2,k},\mu_k), \mathbf{Z} - \mathbf{Z}_k\rangle \\ &\quad + \frac{\eta\mu_k}{2}\|\mathbf{Z} - \mathbf{Z}_k\|_F^2 \\ &= \arg\min_\mathbf{Z} \|\mathbf{Z}\|_* + \left\langle 2\beta\mathbf{Z}_k\mathbf{G} - \mu_k\mathbf{A}^T\left(\mathbf{X} - \mathbf{A}\mathbf{Z}_k - \mathbf{E}_k + \frac{\mathbf{Y}_{1,k}}{\mu_k}\right)\right. \\ &\quad \left. + \mu_k\left(\mathbf{Z}_k - \mathbf{J}_k + \frac{\mathbf{Y}_{2,k}}{\mu_k}\right), \mathbf{Z} - \mathbf{Z}_k\right\rangle + \frac{\eta\mu_k}{2}\|\mathbf{Z} - \mathbf{Z}_k\|_F^2 \\ &= \arg\min_\mathbf{Z} \|\mathbf{Z}\|_* + \frac{\eta\mu_k}{2}\|\mathbf{Z} - \mathbf{Z}_k + \left[\frac{2\beta\mathbf{Z}_k\mathbf{G}}{\mu_k}\right. \\ &\quad \left. - \mathbf{A}^T\left(\mathbf{X} - \mathbf{A}\mathbf{Z}_k - \mathbf{E}_k + \frac{\mathbf{Y}_{1,k}}{\mu_k}\right) + \left(\mathbf{Z}_k - \mathbf{J}_k + \frac{\mathbf{Y}_{2,k}}{\mu_k}\right)\right]/\eta\|_F^2.\end{aligned}$$

Therefore, the updating rule for $\mathbf{Z}$ is as

$$\begin{aligned}\mathbf{Z}_{k+1} \triangleq \Theta_{\frac{1}{\eta\mu_k}}\left[\mathbf{Z}_k + \left[-\frac{2\beta\mathbf{Z}_k\mathbf{G}}{\mu_k} + \mathbf{A}^T\left(\mathbf{X} - \mathbf{A}\mathbf{Z}_k - \mathbf{E}_k + \frac{\mathbf{Y}_{1,k}}{\mu_k}\right)\right.\right. \\ \left.\left. - \left(\mathbf{Z}_k - \mathbf{J}_k + \frac{\mathbf{Y}_{2,k}}{\mu_k}\right)\right]/\eta\right].\end{aligned} \tag{31}$$

$$
\begin{aligned}
\|\mathbf{x}_i - \sum_{j\in\mathcal{N}_i} w_{i_j,i}\mathbf{x}_{i_j}\|_2^2 &= \left\| \sum_{j\in\mathcal{N}_i} (\mathbf{x}_{i_j} - \mathbf{x}_i) \frac{c_{ji}}{\sum_{j'\in\mathcal{N}_i} \frac{c_{j'i}}{\|\mathbf{x}_{i_{j'}} - \mathbf{x}_i\|_2}} \cdot \|\mathbf{x}_{i_j} - \mathbf{x}_i\|_2 \right\|_2^2 \\
&= \left\| \left[\frac{\mathbf{x}_{i_1} - \mathbf{x}_i}{\|\mathbf{x}_{i_1} - \mathbf{x}_i\|_2}, \frac{\mathbf{x}_{i_2} - \mathbf{x}_i}{\|\mathbf{x}_{i_2} - \mathbf{x}_i\|}, \ldots, \frac{\mathbf{x}_{i_{N_i}} - \mathbf{x}_i}{\|\mathbf{x}_{i_{N_i}} - \mathbf{x}_i\|_2} \right] \cdot \begin{bmatrix} c_{1i} / \left(\sum_{j\in\mathcal{N}_i} \frac{c_{ji}}{\|\mathbf{x}_{i_j} - \mathbf{x}_i\|_2} \right) \\ c_{2i} / \left(\sum_{j\in\mathcal{N}_i} \frac{c_{ji}}{\|\mathbf{x}_{i_j} - \mathbf{x}_i\|_2} \right) \\ \cdots \\ c_{N_i i} / \left(\sum_{j\in\mathcal{N}_i} \frac{c_{ji}}{\|\mathbf{x}_{i_j} - \mathbf{x}_i\|_2} \right) \end{bmatrix} \right\|_2^2 \\
&= \left\| \left[\frac{\mathbf{x}_{i_1} - \mathbf{x}_i}{\|\mathbf{x}_{i_1} - \mathbf{x}_i\|_2}, \frac{\mathbf{x}_{i_2} - \mathbf{x}_i}{\|\mathbf{x}_{i_2} - \mathbf{x}_i\|}, \ldots, \frac{\mathbf{x}_{i_{N_i}} - \mathbf{x}_i}{\|\mathbf{x}_{i_{N_i}} - \mathbf{x}_i\|_2} \right] \cdot \frac{1}{\sum_{j\in\mathcal{N}_i} \frac{c_{ji}}{\|\mathbf{x}_{i_j} - \mathbf{x}_i\|_2}} \cdot \begin{bmatrix} c_{1i} \\ c_{2i} \\ \cdots \\ c_{N_i i} \end{bmatrix} \right\|_2^2 \\
&= \|\mathbf{U}_i \mathbf{c}_i\|_2^2 \Big/ \sum_{j\in\mathcal{N}_i} \frac{c_{ji}}{\|\mathbf{x}_{i_j} - \mathbf{x}_i\|_2} \le \theta \Big/ \sum_{j\in\mathcal{N}_i} \frac{c_{ji}}{\|\mathbf{x}_{i_j} - \mathbf{x}_i\|_2} = \varepsilon. \qquad (22)
\end{aligned}
$$

Box I.

- Update $\mathbf{J}$ with other variables fixed.

$$
\begin{aligned}
\mathbf{J}_{k+1} &= \arg\min_{\mathbf{J}} \alpha\|\mathbf{J}\|_1 + \langle \mathbf{Y}_{2,k}, \mathbf{Z}_{k+1} - \mathbf{J}\rangle + \frac{\mu_k}{2}\|\mathbf{Z}_{k+1} - \mathbf{J}\|_F^2 \\
&= \arg\min_{\mathbf{J}} \frac{\alpha}{\mu_k}\|\mathbf{J}\|_1 + \frac{1}{2}\|\mathbf{J} - \left(\mathbf{Z}_{k+1} + \frac{\mathbf{Y}_{2,k}}{\mu_k}\right)\|_F^2.
\end{aligned}
$$

Therefore, the updating rule for $\mathbf{J}$ is as

$$
\mathbf{J}_{k+1} \triangleq \mathcal{S}_{\frac{\alpha}{\mu_k}}\left[\mathbf{Z}_{k+1} + \frac{\mathbf{Y}_{2,k}}{\mu_k}\right].
$$

To enforce the non-negativity on $\mathbf{J}$, we simply set the negative elements in $\mathbf{J}_{k+1}$ to zero as

$$
\mathbf{J}_{k+1} = \max\left[\mathcal{S}_{\frac{\alpha}{\mu_k}}\left(\mathbf{Z}_{k+1} + \frac{\mathbf{Y}_{2,k}}{\mu_k}\right), 0\right]. \qquad (32)
$$

- Update $\mathbf{E}$ with other variables fixed.

$$
\begin{aligned}
\mathbf{E}_{k+1} &= \arg\min_{\mathbf{E}} \lambda\|\mathbf{E}\|_{2,1} + \langle \mathbf{Y}_{1,k}, \mathbf{X} - \mathbf{A}\mathbf{Z}_{k+1} - \mathbf{E}\rangle \\
&\quad + \frac{\mu_k}{2}\|\mathbf{X} - \mathbf{A}\mathbf{Z}_{k+1} - \mathbf{E}\|_F^2 \\
&= \arg\min_{\mathbf{E}} \frac{\lambda}{\mu_k}\|\mathbf{E}\|_{2,1} + \frac{1}{2}\left\|\mathbf{E} - \left(\mathbf{X} - \mathbf{A}\mathbf{Z}_{k+1} + \frac{\mathbf{Y}_{1,k}}{\mu_k}\right)\right\|_F^2.
\end{aligned}
$$

Therefore, the updating rule for $\mathbf{E}$ is as

$$
\mathbf{E}_{k+1} \triangleq \Omega_{\frac{\lambda}{\mu_k}}\left[\mathbf{X} - \mathbf{A}\mathbf{Z}_{k+1} + \frac{\mathbf{Y}_{1,k}}{\mu_k}\right]. \qquad (33)
$$

The notations Θ, $\mathcal{S}$ and Ω are respectively the *singular value thresholding* (Cai et al., 2010), *soft thresholding* (Lin et al., 2010) and $\ell_{2,1}$*-norm minimization* (Liu et al., 2010) operators, which are defined in Appendix B. The complete optimization to MLRR is summarized in Algorithm 1.

3.4. MLRR-based graph construction

Given a data matrix $\mathbf{X}$, we can use itself as the dictionary. $\mathbf{A}$ in (27) can be simply replaced by $\mathbf{X}$ and the learned coefficient $\mathbf{Z}$ measures the self-expressive capacity of data. Once problem (27) is solved, we can obtain an optimal $\mathbf{Z}^*$ in which column $\mathbf{z}_i^*$ of $\mathbf{Z}^*$ depicts how other instances contribute to the representation of $\mathbf{x}_i$. The coefficient matrix $\mathbf{Z}$ has three properties: (1) grouping effect (Li & Fu, 2013; Lu, Feng, Lin, & Yan, 2013) obtained by the low-rankness constraint, which encourages the coefficients of samples from the same manifold to be highly correlated; (2) the sparsity constraint ensures that each sample can only associate with a few samples; (3) geometric structure preserving property obtained by the manifold regularization. Alternatively, MLRR can be seen as a combination of both global information emphasized by the low-rank constraint and local information emphasized by the manifold regularization.

Thus, we can define the affinity matrix of an undirected graph based on $\mathbf{Z}^*$. Similarly, we use the coefficient shrinkage and normalization operators as Zhuang et al. (2012). The whole

Algorithm 1 Efficient LADMAP Algorithm for MLRR

Input: data matrix $\mathbf{X}$, parameters λ, α and β, the manifold identification matrix $\mathbf{W}$;

Output: an optimal solution $\{\mathbf{Z}_k, \mathbf{J}_k, \mathbf{E}_k\}$.

1: Initialization: $\mathbf{Z}_0 = \mathbf{J}_0 = \mathbf{E}_0 = \mathbf{Y}_{1,0} = \mathbf{Y}_{2,0} = \mathbf{0}$, $\mu_0 = 0.1$, $\mu_{\max} = 10^{10}$, $\rho_0 = 1.1$, $\varepsilon_1 = 10^{-6}$, $\varepsilon_2 = 10^{-4}$, $\eta = 1.02\|\mathbf{X}\|_2^2$, $k = 0$.

2: **while** $\mu_k \cdot \max(\sqrt{\eta}\|\mathbf{Z}_k - \mathbf{Z}_{k-1}\|_F, \|\mathbf{J}_k - \mathbf{J}_{k-1}\|_F, \|\mathbf{E}_k - \mathbf{E}_{k-1}\|_F)/\|\mathbf{X}\|_F \ge \varepsilon_2$ or $\|\mathbf{X} - \mathbf{A}\mathbf{Z}_k - \mathbf{E}_k\|_F/\|\mathbf{X}\|_F \ge \varepsilon_1$ **do**

3: Update variable $\mathbf{Z}$ as (31);

4: Update variable $\mathbf{J}$ as (32);

5: Update variable $\mathbf{E}$ as (33);

6: Update Lagrangian multipliers as

$$
\begin{aligned}
\mathbf{Y}_{1,k+1} &= \mathbf{Y}_{1,k} + \mu_k(\mathbf{X} - \mathbf{A}\mathbf{Z}_{k+1} - \mathbf{E}_{k+1}), \\
\mathbf{Y}_{2,k+1} &= \mathbf{Y}_{2,k} + \mu_k(\mathbf{Z}_{k+1} - \mathbf{J}_{k+1});
\end{aligned} \qquad (34)
$$

7: Update μ as

$$
\mu_{k+1} = \min(\mu_{\max}, \rho\mu_k), \qquad (35)
$$

where

$$
\rho = \begin{cases} \rho_0, & \text{if } \mu_k \max(\sqrt{\eta}\|\mathbf{Z}_{k+1} - \mathbf{Z}_k\|_F, \|\mathbf{J}_{k+1} - \mathbf{J}_k\|_F, \\ & \quad \|\mathbf{E}_{k+1} - \mathbf{E}_k\|_F)/\|\mathbf{X}\|_F < \varepsilon_2, \\ 1, & \text{otherwise;} \end{cases}
$$

8: Update k as $k = k + 1$;

9: **end while**

procedure for constructing MLRR-graph is summarized in Algorithm 2.

Algorithm 2 MLRR based Graph Construction

Input: data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$
Output: The affinity matrix $\mathbf{S}$ of MLRR-graph.

1: Normalize each sample to ℓ_2 unit norm via $\hat{\mathbf{x}}_i = \mathbf{x}_i/\|\mathbf{x}_i\|_2$ to obtain $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \cdots, \hat{\mathbf{x}}_n]$;
2: Identify the inner structure of data by solving

$$\min_{\mathbf{c}_i\in\mathbb{R}^{N_i}} \frac{1}{2}\|\mathbf{U}_i\mathbf{c}_i\|_2^2 + \gamma\|\mathbf{c}_i\|_1 \quad s.t.\ \mathbf{1}^T\mathbf{c}_i = 1; \tag{36}$$

3: Formulate the manifold regularizer (26) based on the $\mathbf{W}$ computed by (20);
4: Optimize the following problem using Algorithm 1:

$$\min_{\mathbf{Z},\mathbf{E}} \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \alpha\|\mathbf{Z}\|_1 + \beta\mathcal{G}(\mathbf{Z}) \quad s.t.\ \mathbf{X} = \mathbf{X}\mathbf{Z} + \mathbf{E}, \mathbf{Z} \geq 0$$

and obtain the optimal solution $(\mathbf{Z}^*, \mathbf{E}^*)$.
5: Normalize all column vectors of $\mathbf{Z}^*$ by $\mathbf{z}_i^* = \mathbf{z}_i^*/\|\mathbf{z}_i^*\|_2$ and shrink $\mathbf{Z}^*$ by

$$\hat{z}_{ij}^* = \begin{cases} z_{ij}^*, & \text{if } z_{ij}^* \geq \delta, \\ 0, & \text{otherwise;} \end{cases} \tag{37}$$

and obtain a sparse $\hat{\mathbf{Z}}^*$.
6: Construct the graph affinity matrix $\mathbf{S}$ by

$$\mathbf{S} = (|\hat{\mathbf{Z}}^*| + |\hat{\mathbf{Z}}^*|^T)/2. \tag{38}$$

3.5. Computational complexity analysis

In this section, we give a brief analysis of the computational complexity of the MLRR model based on the O notation. Obviously, computing $\mathbf{c}_i$ involved in problem (19) and $\mathbf{Z}$ involved in problem (27) cause the main complexity for MLRR.

In optimizing problem (19) (see Appendix A), since the $(\mathbf{U}^T\mathbf{U} + \lambda\mathbf{I})^{-1}$ is not related to the variable $\mathbf{r}$ and can be computed beforehand, updating $\mathbf{r}$ is the main computation of ADMM for each data point and its complexity is $O(K^2)$, where K here is equivalent to N_i in (19). As a result, the complexity for solving problem (19) is $O(t_1nK^2)$, where t_1 is the number of iterations for ADMM method.

The main computation of problem (27) is updating $\mathbf{Z}$, which needs to compute the SVD decomposition of an $n \times n$ matrix. Therefore it will be time consuming if n is large, i.e., the number of data samples is large. Similar to Liu et al. (2013), the computational cost can be reduced based on the following theorem.

Theorem 3. *For any optimal solution $(\mathbf{Z}^*, \mathbf{E}^*)$ to problem (27), we have $\mathbf{Z}^* \in \text{span}(\mathbf{A}^T)$ (Liu et al., 2013).*

Theorem 3 shows that the optimal solution $\mathbf{Z}^*$ to problem (27) always lies in the subspace spanned by rows of $\mathbf{A}$. It means that $\mathbf{Z}^*$ can be factorized into $\mathbf{Z}^* = \mathbf{P}^*\tilde{\mathbf{Z}}^*$, where $\mathbf{P}^*$ can be computed beforehand via orthogonalizing the columns of $\mathbf{A}^T$. Therefore, we can solve the following problem instead,

$$\min_{\tilde{\mathbf{Z}},\mathbf{E}} \|\tilde{\mathbf{Z}}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \alpha\|\tilde{\mathbf{Z}}\|_1 + \beta\mathcal{G}(\tilde{\mathbf{Z}}) \quad s.t.\ \mathbf{X} = \mathbf{B}\tilde{\mathbf{Z}} + \mathbf{E},\ \tilde{\mathbf{Z}} \geq 0, \tag{39}$$

where $\mathbf{B} = \mathbf{A}\mathbf{P}^*$. Once a solution $(\tilde{\mathbf{Z}}^*, \mathbf{E}^*)$ to (39) is obtained, the optimal solution to (27) is $(\mathbf{P}^*\tilde{\mathbf{Z}}^*, \mathbf{E}^*)$.

Assuming that the rank of $\mathbf{A}$ is r (i.e., $\tilde{\mathbf{Z}}$ has at most r rows), we get the complexity for updating $\mathbf{Z}$: the SVD decomposition for a $r \times n$ matrix is $O(r^2n)$, and the related multiplication is $O(r^3 + r^2n + rdn)$. Therefore, the complexity is $O(d^2n + t_2(r^2n + r^3 + rdn))$ if we consider the orthogonalization $(O(d^2n))$ and the iterations of running the LADMP algorithm (t_2). As a whole, we can get the complexity of optimizing MLRR is $O(t_1nK^2 + d^2n + t_2(r^2n + r^3 + rdn))$.

Table 1
Statistics of the four data sets.

Dataset	#size(n)	#dimensionality(d)	#classes (c)
ORL	400	1024	40
Extended Yale B	1000	1024	20
CMU PIE	1000	1024	20
ISOLET	1560	617	26

4. Experimental studies

In this section, we evaluate the effectiveness of MLRR-graph on public data sets. The comparison between MLRR and other several graph construction methods includes two parts, (1) comparing MLRR-graph with graphs constructed by LRR variants mentioned in Section 2.1, (2) comparing MLRR-graph with some other state-of-the-art graphs. Source codes to MLRR will be available from http://bcmi.sjtu.edu.cn/~pengyong/.

4.1. Data sets

We select three face data sets and one voice data set, ORL, Extended Yale B, CMU PIE and ISOLET, in our experiments. The statistics of these data sets are summarized below (see also Table 1):

- **ORL**[1]: 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.
- **Extended Yale B**[2]: This face data set has 38 individuals, each subject having around 64 near frontal images under different illuminations. We simply use the first 50 cropped images of the first 20 individuals, and then resize them to 32×32 pixels.
- **CMU PIE**[3]: This face data set contains 41,368 images of 68 subjects with different poses, illumination and expressions. We only use their images in five near frontal poses (C05, C07, C09, C27 and C29) and under different illuminations and expressions. The first 50 images of the first 20 subjects are selected. Each image is manually cropped and resized to size 32×32 pixels.
- **ISOLET**[4]: The ISOLET dataset is used to predict which letter-name was spoken. The features include spectral coefficients, contour features, sonorant features, per-sonorant features, and post-sonorant features. The feature dimension is 617 and the number of samples is 1560.

Several sample images of the three face data sets are shown in Fig. 2.

For efficiently evaluating the performance of all algorithms, 10% to 60% samples in each class are randomly selected, which are treated as labeled samples. On each percentage of labeled samples, we repeat the experiment 50 trials for each algorithm. For fair

[1] http://www.uk.research.att.com/facedatabase.html.
[2] http://vision.ucsd.edu/~leekc/ExtYaleDatabase/ExtYaleB.html.
[3] http://www.ri.cmu.edu/projects/project_418.html.
[4] http://archive.ics.uci.edu/ml/datasets/ISOLET.

(a) Sample images of two subjects in ORL.

(b) Sample images of two subjects in Extended Yale B.

(c) Sample images of two subjects in CMU PIE.

Fig. 2. Sample images from the three face data sets.

comparison, we record the indices of randomly selected samples and use these indices for all algorithms. The average performance of these 50 trials on each percentage of labeled samples will be reported.

We use the semi-supervised classification framework described in Section 2.3 and set $\psi_{ii} = 1$ if $\mathbf{x}_i$ is labeled for all graph-based algorithms in our experiments. To avoid the singularity when calculating the inverse of $(\tilde{\mathbf{L}} + \boldsymbol{\Psi})$, we always add a small value 10^{-6} on its diagonal elements.

4.2. Comparing with sparse manifold adaption

The manifold information used in MLRR is identified by sparse manifold adaption, which also can generate a graph measured by affinity matrix $\mathbf{W}$. We call it SMA-graph for short. Similarly, this graph can be used for semi-supervised classification. Therefore, we compare the classification performance of MLRR-graph with SMA-graph on the four data sets in this section. The parameter γ in sparse manifold adaption is searched from $\{10^{-3}, 10^{-2}, \ldots, 10^{2}\}$ and the number of nearest neighbors is selected from $\{5, 10, 20, 30, 50\}$. The best average results of SMA-graph are used for comparison.

Fig. 3 shows the performance comparison between these two graph construction methods, which offer us two insights: (1) SMA-graph is very competitive in semi-supervised classification since it can effectively explore the data local structure information; and (2) Combining the sparse manifold adaption with LRR can obtain much accuracy improvement. The global structure captured by low rankness can be combined with the local structure induced by sparse manifold adaption, which is more beneficial than emphasizing one of them only. In the following section, we will show MLRR is more effective than several LRR variants in graph-based semi-supervised classification.

Figs. 4 and 5 show the performance of sparse manifold adaption versus parameters N_i and γ, respectively. Obviously, we can see that sparse manifold adaption enjoys the satisfactory performance when parameter γ takes value in $\{1, 10\}$. The performance of sparse manifold adaption is insensitive to the number of nearest neighbors N_i provided that N_i is a relatively large value between 20 and 50.

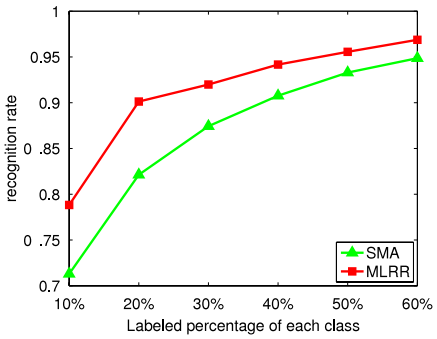
4.3. Comparing with LRR variants

In this section, we compare MLRR with several LRR related graphs including LRR (Liu et al., 2010; Yang and Wang et al., 2013), GLRR (Lu and Wang et al., 2013), LRRLC (Zheng, Zhang, Yang et al., 2013) and NNLRS (Zhuang et al., 2012). GLRR model employs accelerated gradient method (Ji & Ye, 2009) to update $\mathbf{J}$, which is the auxiliary variable w.r.t. $\mathbf{Z}$; while in our experiments, we relax the GLRR objective function as described in Zheng, Zhang, Jia et al. (2013) to solve $\mathbf{J}$ by using the SVT operator (Lin et al., 2010).

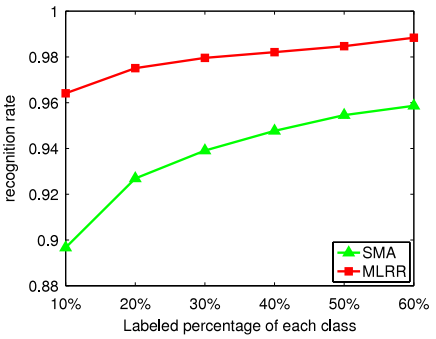
There are several parameters in LRR, LRRLC, GLRR and NNLRS. For each LRR variant, the parameter (usually denoted by λ) to control the impact of error term is searched from $\{10^{-1}, 1, 10, 10^{2}\}$ and parameter (usually denoted by α) to control the impact of locality for LRRLC and GLRR or sparsity for NNLRS is searched from $\{10^{-3}, 10^{-2}, \ldots, 10^{2}\}$. For GLRR, the number of nearest neighbors k is set as 5 and the variance σ in 'HeatKernel' is set as the mean value of the distances between all of the n data points. We will give parameters setting of MLRR in the end of this section. Total 50 trials are run for each candidate value of λ for LRR and each group of (λ, α) for LRRLC, GLRR and NNLRS.

Table 2 reports the mean accuracies as well as standard deviations over these 50 trials. The best results are shown in boldface. From the results, it can be easily observed that:

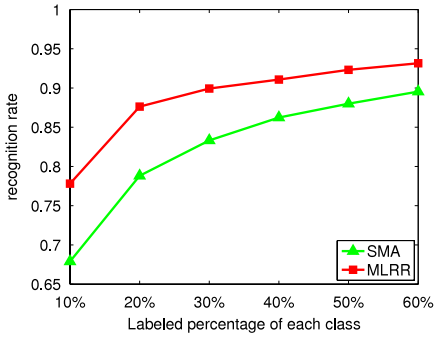
- In most cases, our proposed MLRR model, which employs the sparse manifold adaption to explore the local structure of data, achieves higher recognition rates than the original LRR model and several variants such as LRRLC, GLRR and NNLRS. This demonstrates that enforcing the low-rank coefficient to preserve the geometrical constraints identified in the original data space is reasonable and promising. GLRR uses graph Laplacian to preserve the intrinsic local structure; however, it has been proven to be sensitive to the kernel parameter when calculating the affinity between two data points (Wang & Zhang, 2008; Xiang, Nie, & Zhang, 2010).
- MLRR still can achieve high accuracy when given a small amount of labeled samples. Specifically, on ORL data set, the accuracy obtained with MLRR is 6.9 % higher than the best



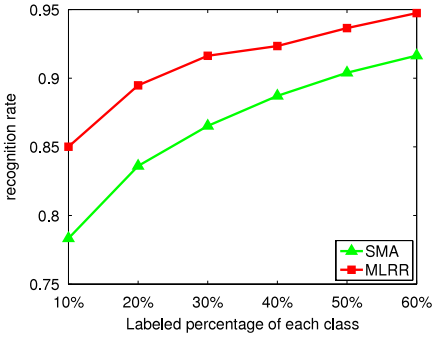
(a) ORL.



(b) Extended Yale B.

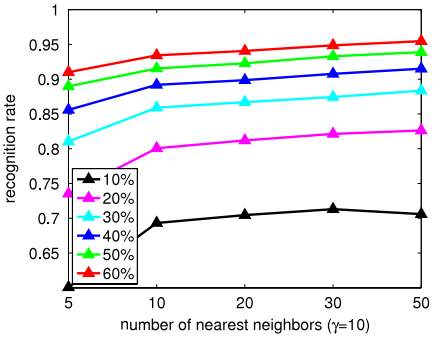


(c) CMU PIE.

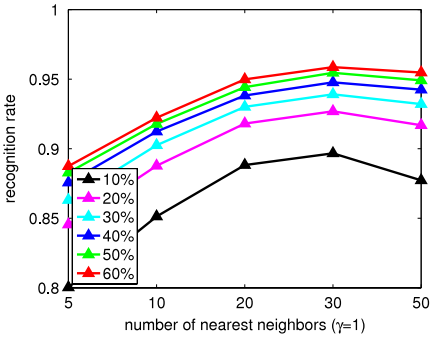


(d) ISOLET.

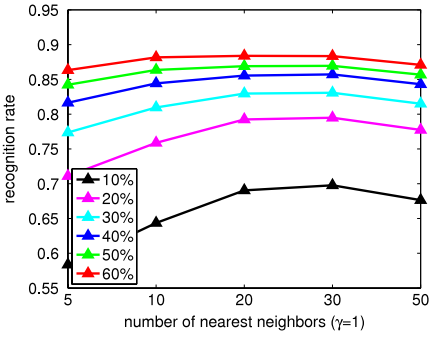
Fig. 3. Performance comparison between MLRR-graph and SMA-graph.



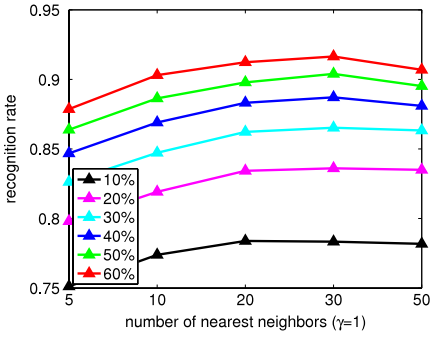
(a) ORL.



(b) Extended Yale B.

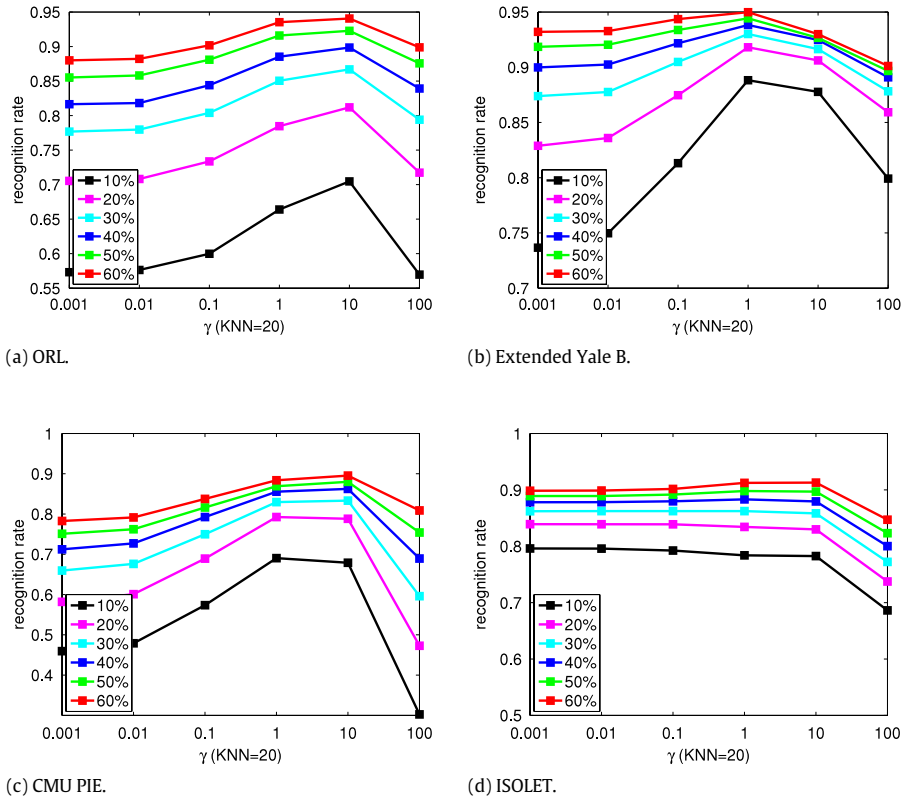


(c) CMU PIE.



(d) ISOLET.

Fig. 4. Performance of sparse manifold adaption with different numbers of nearest neighbors.



(a) ORL. (b) Extended Yale B.

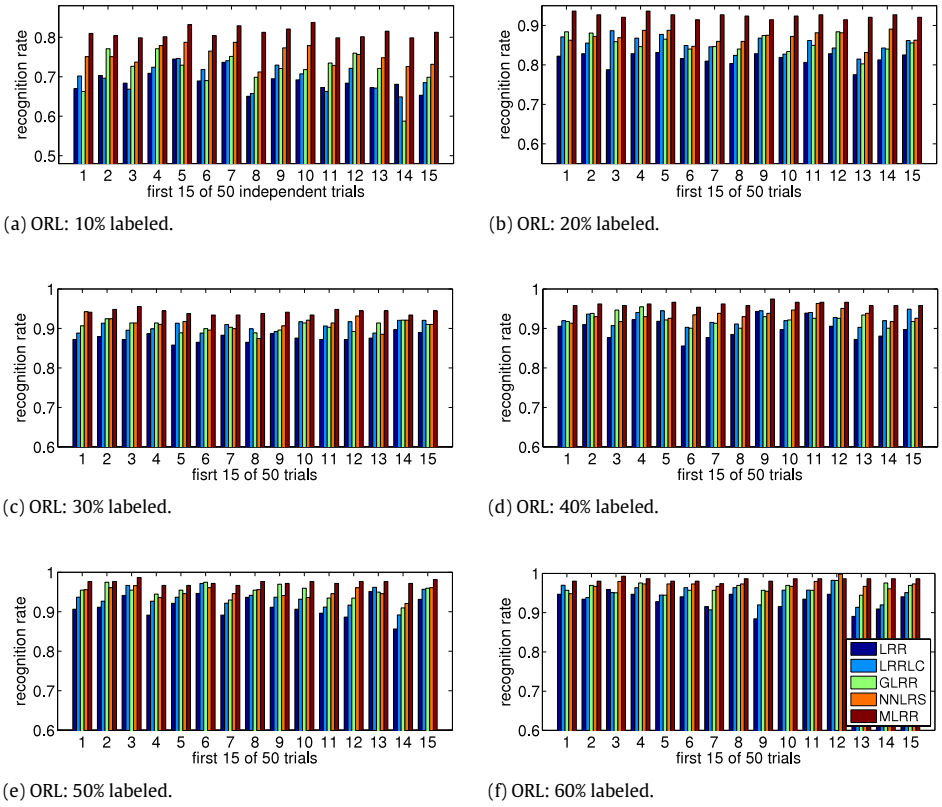
(c) CMU PIE. (d) ISOLET.

Fig. 5. Performance of sparse manifold adaption with different values of parameter γ.

Table 2
Results obtained from LRR variants on the four data sets (mean ± std − dev%).

ORL	LRR	LRRLC	GLRR	NNLRS	MLRR
10%	67.71 ± 2.38	69.12 ± 2.19	70.86 ± 2.78	72.93 ± 2.09	**78.83 ± 1.17**
20%	80.13 ± 2.10	82.79 ± 2.23	83.03 ± 2.53	84.92 ± 2.41	**90.13 ± 1.87**
30%	86.29 ± 1.76	87.93 ± 2.17	88.92 ± 2.37	89.23 ± 2.20	**91.99 ± 1.96**
40%	88.97 ± 1.69	89.83 ± 1.96	91.83 ± 1.63	91.79 ± 1.78	**94.17 ± 1.82**
50%	90.67 ± 1.51	91.34 ± 1.88	94.60 ± 1.54	93.88 ± 1.59	**95.56 ± 1.70**
60%	92.19 ± 1.63	92.87 ± 1.77	95.41 ± 1.47	95.10 ± 1.63	**96.87 ± 1.79**
Yale B	LRR±	LRRLC±	GLRR±	NNLRS±	MLRR±
10%	88.43 ± 1.92	90.12 ± 1.88	90.58 ± 1.76	94.55 ± 0.83	**96.41 ± 0.77**
20%	93.07 ± 1.47	94.79 ± 1.37	93.16 ± 1.38	95.97 ± 0.53	**97.51 ± 0.53**
30%	94.83 ± 1.03	95.96 ± 0.98	95.07 ± 1.09	96.79 ± 0.49	**97.96 ± 0.38**
40%	95.78 ± 1.15	96.80 ± 0.79	95.97 ± 0.93	96.98 ± 0.56	**98.21 ± 0.43**
50%	96.23 ± 0.91	97.56 ± 0.63	96.61 ± 0.91	97.54 ± 0.47	**98.47 ± 0.49**
60%	96.81 ± 0.79	97.86 ± 0.55	97.23 ± 0.76	97.91 ± 0.52	**98.84 ± 0.57**
PIE	LRR±	LRRLC±	GLRR±	NNLRS±	MLRR±
10%	74.21 ± 2.58	74.59 ± 2.71	75.20 ± 2.61	**78.13 ± 2.53**	77.81 ± 2.47
20%	83.91 ± 1.54	84.51 ± 1.49	84.98 ± 1.83	85.17 ± 1.47	**87.63 ± 1.72**
30%	87.83 ± 1.22	88.27 ± 1.37	88.59 ± 1.17	87.99 ± 1.13	**89.93 ± 1.23**
40%	89.67 ± 1.18	89.80 ± 1.19	90.51 ± 1.43	89.77 ± 1.18	**91.07 ± 1.39**
50%	90.33 ± 1.23	90.79 ± 1.06	91.48 ± 1.01	90.62 ± 1.23	**92.32 ± 1.38**
60%	90.89 ± 1.03	91.23 ± 1.36	92.14 ± 1.08	91.46 ± 1.30	**93.15 ± 1.37**
ISOLET	LRR±	LRRLC±	GLRR±	NNLRS±	MLRR±
10%	68.47 ± 1.45	73.22 ± 1.56	75.82 ± 1.29	83.17 ± 1.21	**85.01 ± 1.16**
20%	80.34 ± 1.13	81.51 ± 1.41	83.70 ± 1.13	87.95 ± 0.92	**89.48 ± 0.96**
30%	86.43 ± 1.19	87.16 ± 1.23	88.29 ± 0.97	89.98 ± 0.83	**91.63 ± 0.83**
40%	89.30 ± 0.91	90.11 ± 0.79	90.83 ± 0.84	91.04 ± 0.86	**92.34 ± 0.71**
50%	90.98 ± 0.93	91.28 ± 0.88	91.93 ± 0.90	91.78 ± 0.91	**93.65 ± 0.83**
60%	92.37 ± 0.87	92.46 ± 0.91	92.98 ± 0.81	92.73 ± 0.79	**94.74 ± 0.87**

result obtained from the other LRR variants; on Extended Yale B and ISOLET data sets, the improvements are respectively 1.86 and 1.84. When given increasing number of labeled samples, the performance of all LRR variants will get improved, which shows that low-rank representation is a good approach to graph construction.



(a) ORL: 10% labeled.

(b) ORL: 20% labeled.

(c) ORL: 30% labeled.

(d) ORL: 40% labeled.

(e) ORL: 50% labeled.

(f) ORL: 60% labeled.

Fig. 6. Results obtained from LRR variants of the first 15 trials (of total 50 trials) on ORL.

- NNLRS-graph and MLRR-graph obtain similar performance when given a small amount of labeled samples, which further shows that the non-negativity and sparsity constraints on representation coefficients are beneficial because both graphs share these two common properties. The reason why these two constraints can bring better performance with limited labeled samples will be investigated in detail in our future work.

Figs. 6–9 respectively show the classification results of all LRR variants on the first 15 trials (of total 50 trials) on ORL, Extended Yale B, CMU PIE and ISOLET data sets. It is obvious that MLRR achieves the best results on most of the trials in comparison with the other LRR variants.

For better visualizing the recognition rate distribution of these LRR variants based graph construction models, we depict the box plots of each model in Fig. 10.

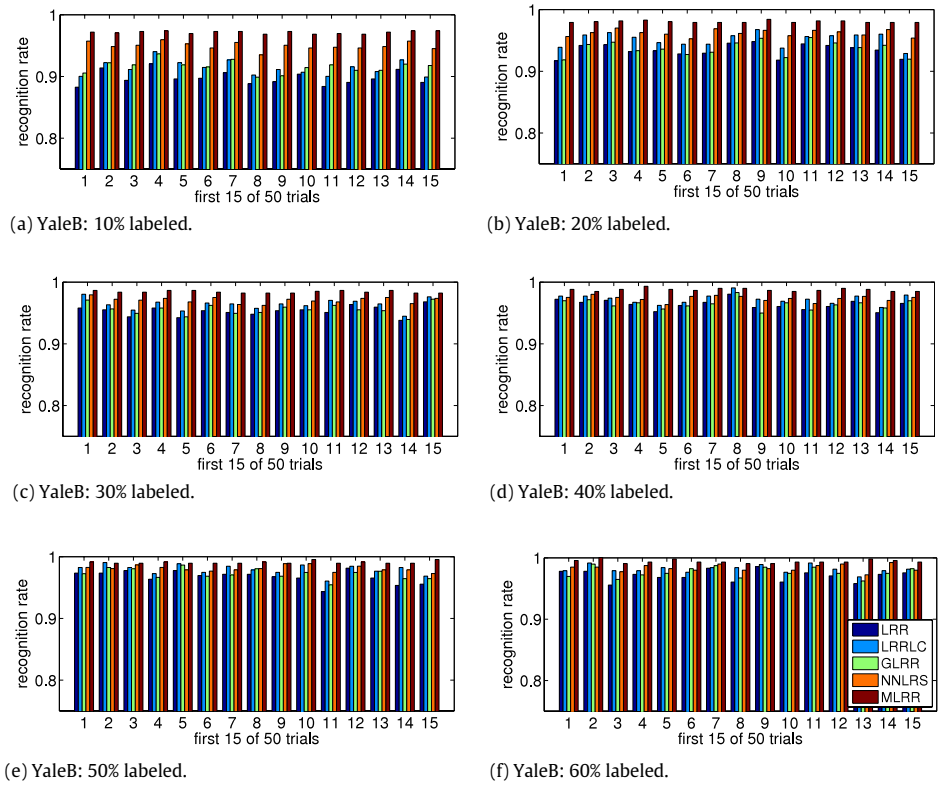
Now we examine the parameter sensitivity of MLRR-graph, which includes three main parameters (λ, α and β) besides the parameters (N_i and γ) involved in sparse manifold adaption. In MLRR, λ is to deal with the corruption in data, α is to control the sparsity of representation coefficients and β is to emphasize the effect of manifold regularization. Based on the results of sparse manifold adaption in Section 4.2, we set (N_i, γ) as a near optimal combination (20, 1) for all data sets. As reported in Zhuang et al. (2012) and Zheng, Zhang, Jia et al. (2013), LRR variants are insensitive to the variation of λ provided that it is given a relatively large value (usually 10). Thus it is reasonable to set λ as a fixed value to alleviate the burden of parameter tuning, which means that the level of corruption in data could be fixed. In all above experiments, we set $\lambda = 10$. Now we vary the parameters α and β and evaluate the performance of MLRR-graph based semi-supervised classification.

Figs. 11 and 12 respectively show the change of accuracy versus parameters α and β. We fix (λ, β) as (10, 0.01) in Fig. 11 and (λ, α) as (10, 0.01) in Fig. 12. From these two figures, we can see that the classification accuracy obtained from MLRR will decrease when α and β are large. Generally, MLRR can achieve promising results when $\alpha < 1$ and $\beta < 0.1$ on all four data sets. MLRR is slightly sensitive to parameters when given a small amount of labeled samples; therefore, when given more labeled samples, the performance of MLRR will become more robust. Obviously, 0.001 and 0.01 are suitable candidate values for both α and β. In all our experiments above, we always set (α, β) as (0.01, 0.01).

4.4. Comparing with other state-of-the-art models

In order to further evaluate the effectiveness of MLRR, we compare it with the following state-of-the-art graph construction models.

- KNN-graph. Samples $\mathbf{x}_i$ and $\mathbf{x}_j$ are considered as neighbors if $\mathbf{x}_i$ is among the k nearest neighbors of $\mathbf{x}_j$ or $\mathbf{x}_j$ is among the k nearest neighbors of $\mathbf{x}_i$. The number of nearest neighbors for KNN1 and KNN2 is respectively 4 and 8. The distance information is measured by 'HeatKernel' where the variance σ is the average of squared Euclidean distances for all edged pairs on graph.
- Local spline regression (LSR)-based graph (Xiang et al., 2010). In the neighborhood of each data point, an optimal spline is estimated via regularized least square regression. The losses in local neighborhoods are accumulated together to measure the global consistency on the labeled and unlabeled data points. We use the learned graph Laplacian matrix, which was denoted by $\mathbf{M}$ in Xiang et al. (2010), for graph-based semi-supervised learning task. The parameter γ in Xiang et al. (2010) is set as 1, which is equivalent to the semi-supervised framework described in Section 2.3. The remaining two parameters: the number of nearest neighbors k is searched from {5, 10, 20, 30} and the regularization parameter λ for local spline regression from $\{10^{-3}, 10^{-2}, \ldots, 10^{2}\}$. The best average accuracies are reported.

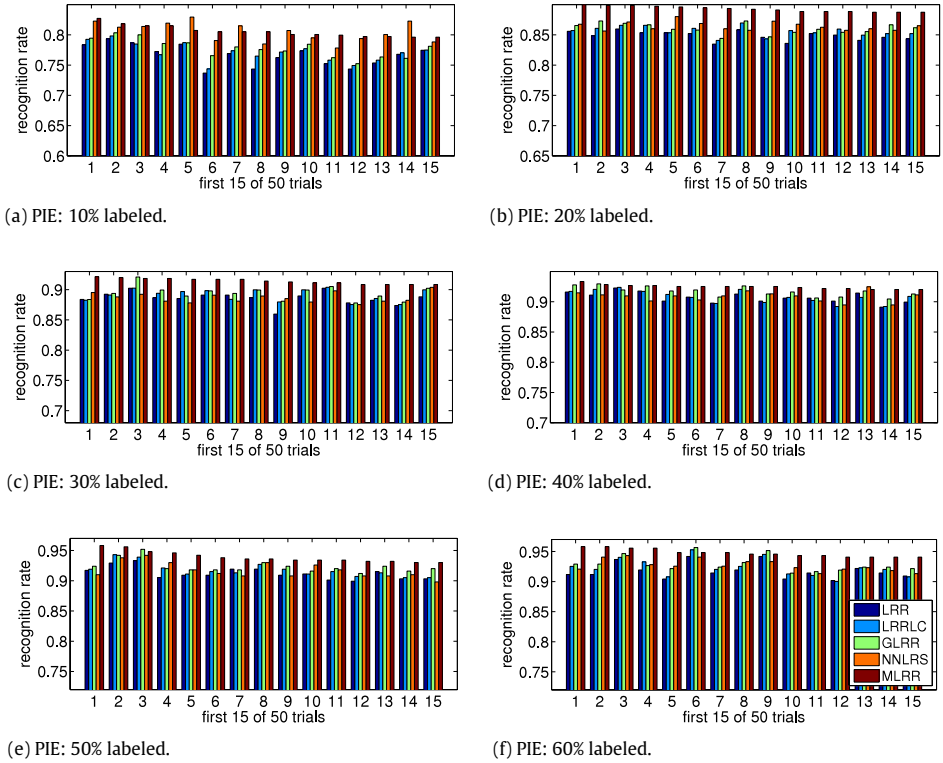


(a) YaleB: 10% labeled. (b) YaleB: 20% labeled.

(c) YaleB: 30% labeled. (d) YaleB: 40% labeled.

(e) YaleB: 50% labeled. (f) YaleB: 60% labeled.

Fig. 7. Results obtained from LRR variants of the first 15 trials (of total 50 trials) on Extended Yale B.



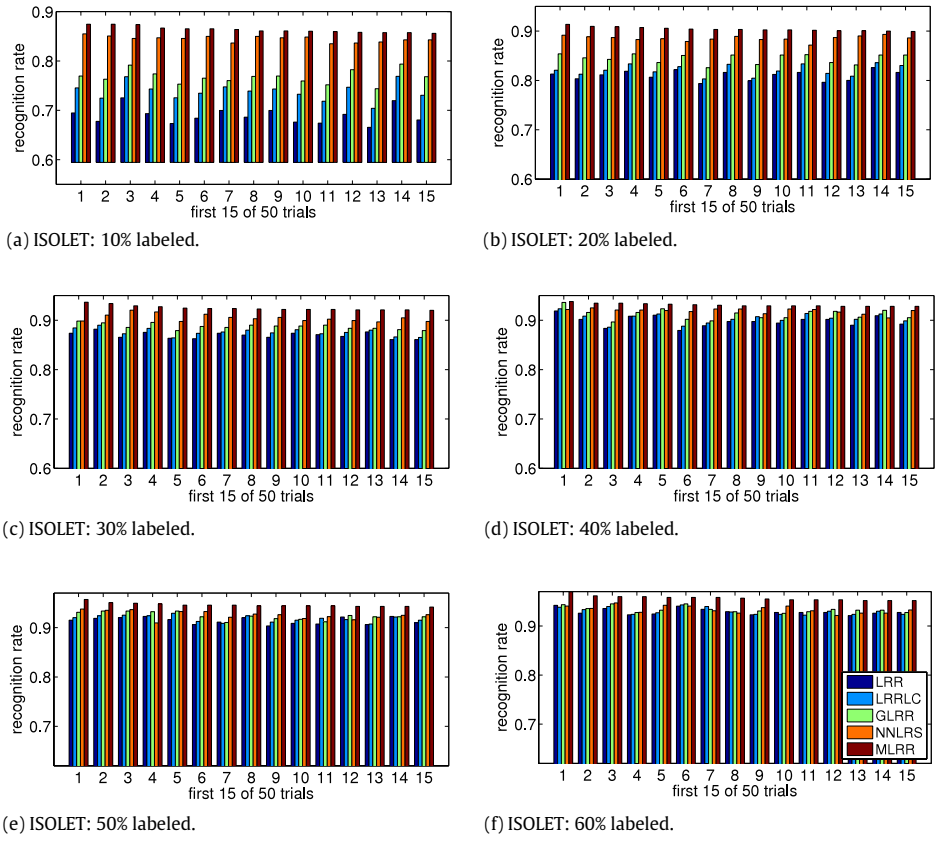
(a) PIE: 10% labeled. (b) PIE: 20% labeled.

(c) PIE: 30% labeled. (d) PIE: 40% labeled.

(e) PIE: 50% labeled. (f) PIE: 60% labeled.

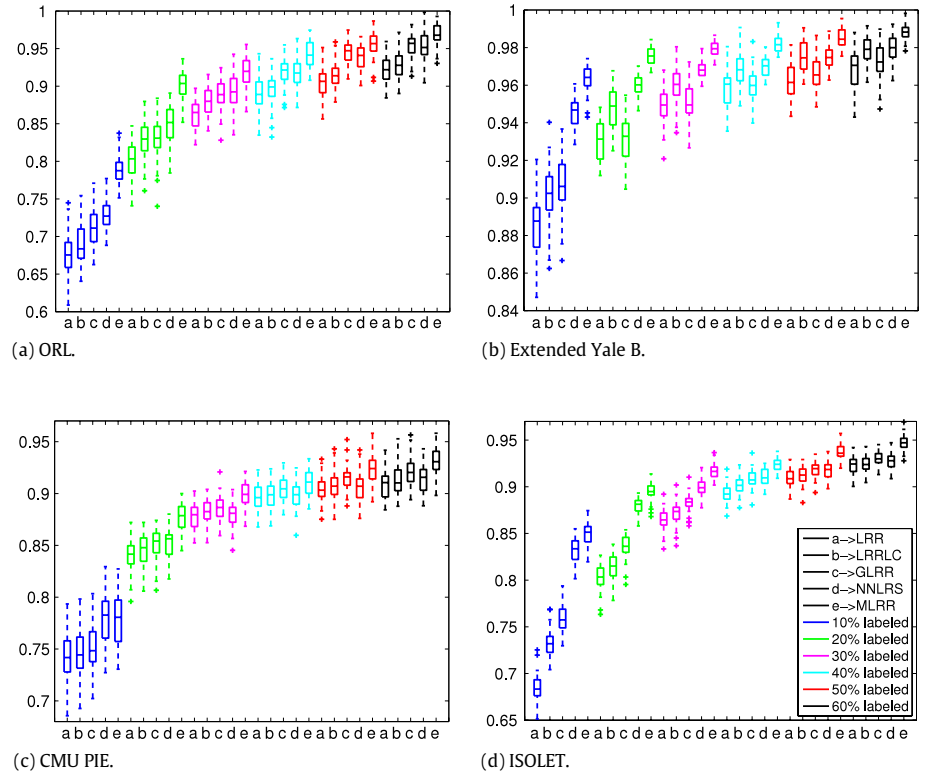
Fig. 8. Results obtained from LRR variants of first 15 trials (of total 50 trials) on CMU PIE.

- Local regression and global alignment (LRGA) (Yang, Nie, Xu, Luo, Zhuang and Pan, 2012). LRGA was originally proposed for multimedia retrieval; meanwhile, the robust Laplacian matrix learned by LRGA is also suitable for semi-supervised learning. In LRGA, for each data point, a local linear regression model is used to explore the local structure; then a unified objective function is proposed to globally align the local models from all the data points. Two related parameters, the number of



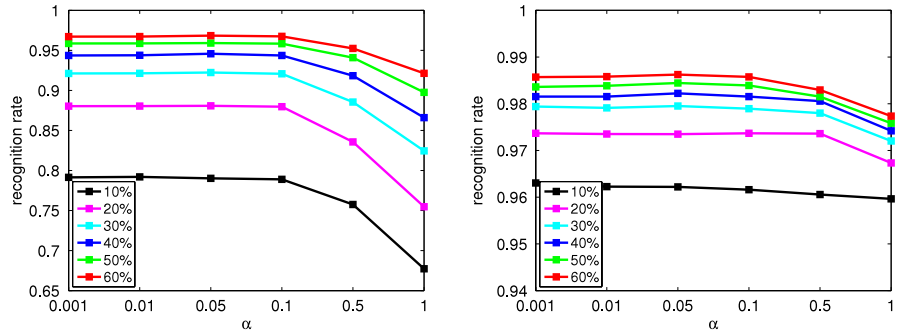
(a) ISOLET: 10% labeled.
(b) ISOLET: 20% labeled.
(c) ISOLET: 30% labeled.
(d) ISOLET: 40% labeled.
(e) ISOLET: 50% labeled.
(f) ISOLET: 60% labeled.

Fig. 9. Results obtained from LRR variants of the first 15 trials (of total 50 trials) on ISOLET.

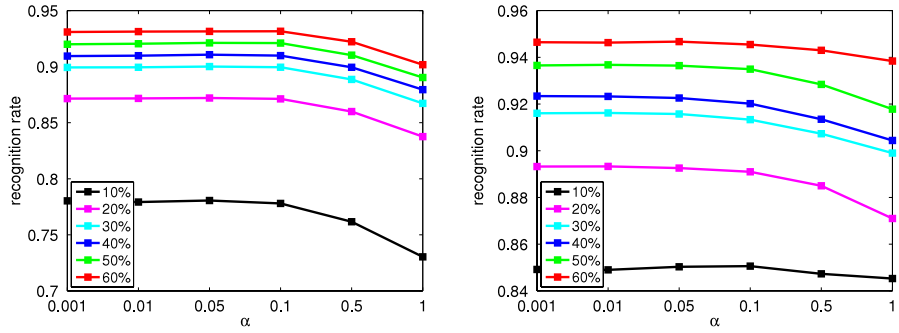


(a) ORL.
(b) Extended Yale B.
(c) CMU PIE.
(d) ISOLET.

Fig. 10. Results obtained from LRR variants on the four data sets.

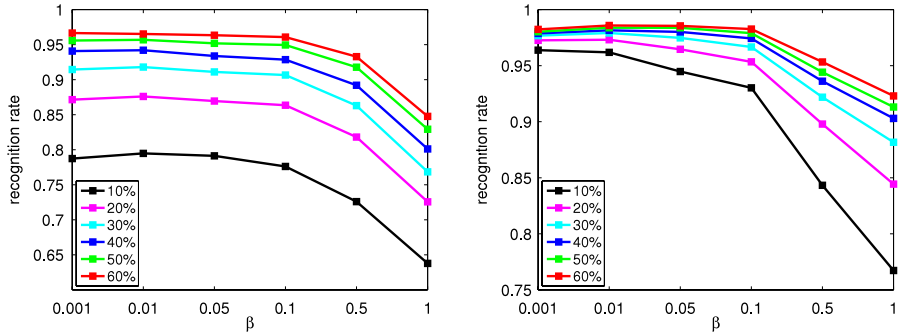


(a) ORL. (b) Extended Yale B.

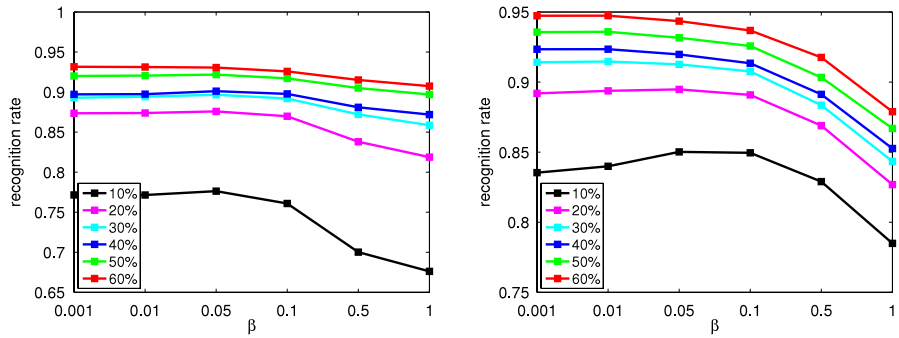


(c) CMU PIE. (d) ISOLET.

Fig. 11. Performance of MLRR with different values of parameter α.



(a) ORL. (b) Extended Yale B.



(c) CMU PIE. (d) ISOLET.

Fig. 12. Performance of MLRR with different values of parameter β.

Table 3
Results obtained from state-of-the-art graph construction models on the four data sets (mean ± std − dev%).

ORL	KNN1	KNN2	LSR	LRGA	LNP	L1-Graph	SPG	MLRR
10%	64.98 ± 2.07	53.47 ± 2.80	57.21 ± 3.13	65.93 ± 2.62	71.21 ± 2.30	61.89 ± 2.69	65.92 ± 2.42	**78.83 ± 1.17**
20%	74.07 ± 2.71	63.97 ± 2.69	71.84 ± 3.08	75.94 ± 2.55	82.01 ± 2.20	76.16 ± 2.73	78.74 ± 1.98	**90.13 ± 1.87**
30%	79.42 ± 2.24	68.93 ± 2.72	79.78 ± 3.09	81.81 ± 2.97	88.06 ± 2.45	84.61 ± 2.32	86.24 ± 2.36	**91.99 ± 1.96**
40%	81.16 ± 2.43	71.64 ± 2.82	85.04 ± 2.78	85.90 ± 2.22	91.43 ± 1.68	89.31 ± 1.90	90.72 ± 1.64	**94.17 ± 1.82**
50%	82.76 ± 2.41	72.83 ± 2.44	88.76 ± 1.76	89.53 ± 1.61	93.15 ± 1.74	92.47 ± 1.69	93.20 ± 1.64	**95.56 ± 1.70**
60%	83.23 ± 2.20	74.75 ± 2.69	91.31 ± 2.37	92.10 ± 2.16	94.69 ± 1.64	94.25 ± 1.59	95.46 ± 1.50	**96.87 ± 1.79**
Yale B	KNN1±	KNN2±	LSR±	LRGA±	LNP±	L1-Graph±	SPG±	MLRR±
10%	73.04 ± 1.62	55.82 ± 2.91	63.03 ± 2.37	72.64 ± 2.45	86.22 ± 1.52	77.69 ± 1.74	82.92 ± 1.62	**96.41 ± 0.77**
20%	77.13 ± 1.31	63.03 ± 2.39	73.27 ± 1.45	77.78 ± 1.49	90.86 ± 1.14	87.58 ± 1.05	89.84 ± 1.16	**97.51 ± 0.53**
30%	80.12 ± 1.38	67.29 ± 1.91	78.25 ± 1.52	82.03 ± 1.38	92.45 ± 0.72	92.13 ± 1.15	92.75 ± 0.99	**97.96 ± 0.38**
40%	81.49 ± 1.35	69.86 ± 2.46	81.57 ± 1.62	85.84 ± 1.89	93.42 ± 0.86	94.39 ± 0.94	94.26 ± 0.77	**98.21 ± 0.43**
50%	83.50 ± 1.43	72.24 ± 2.45	83.80 ± 1.37	88.45 ± 1.57	93.85 ± 0.82	95.90 ± 1.03	95.59 ± 0.72	**98.47 ± 0.49**
60%	84.25 ± 2.03	74.74 ± 2.42	85.26 ± 1.46	90.38 ± 1.64	94.89 ± 0.98	97.29 ± 0.93	96.37 ± 0.85	**98.84 ± 0.57**
PIE	KNN1±	KNN2±	LSR±	LRGA±	LNP±	L1-Graph±	SPG±	MLRR±
10%	49.38 ± 2.83	40.09 ± 2.11	46.55 ± 2.00	54.31 ± 2.17	67.50 ± 2.77	63.60 ± 2.35	65.29 ± 2.16	**77.81 ± 2.47**
20%	59.22 ± 2.24	51.99 ± 1.93	61.53 ± 2.13	67.75 ± 2.23	79.11 ± 1.47	76.43 ± 1.18	77.80 ± 1.70	**87.63 ± 1.72**
30%	64.69 ± 1.73	60.76 ± 1.77	70.32 ± 2.09	76.55 ± 2.18	83.54 ± 1.73	82.93 ± 1.44	83.82 ± 1.26	**89.93 ± 1.23**
40%	67.12 ± 1.91	68.56 ± 1.45	76.65 ± 2.27	82.53 ± 2.14	87.02 ± 1.35	86.99 ± 1.25	87.23 ± 1.25	**91.07 ± 1.39**
50%	69.87 ± 2.09	75.11 ± 1.10	80.20 ± 2.34	85.50 ± 1.87	89.08 ± 1.38	89.34 ± 1.02	89.35 ± 1.41	**92.32 ± 1.38**
60%	71.49 ± 2.04	81.15 ± 1.11	83.45 ± 2.33	88.06 ± 1.02	90.07 ± 1.53	90.96 ± 1.32	91.60 ± 1.63	**93.15 ± 1.37**
ISOLET	KNN1±	KNN2±	LSR±	LRGA±	LNP±	L1-Graph±	SPG±	MLRR±
10%	75.06 ± 1.43	77.35 ± 1.26	76.67 ± 2.03	76.33 ± 2.18	78.58 ± 1.54	71.49 ± 1.37	77.47 ± 1.35	**85.01 ± 1.16**
20%	79.04 ± 1.29	81.17 ± 1.35	82.94 ± 1.22	83.61 ± 1.37	83.69 ± 0.94	75.69 ± 1.36	83.58 ± 0.97	**89.48 ± 0.96**
30%	80.95 ± 1.14	82.93 ± 1.21	85.94 ± 0.88	87.08 ± 1.11	86.60 ± 0.91	82.90 ± 1.13	86.99 ± 0.97	**91.63 ± 0.83**
40%	82.36 ± 1.07	84.41 ± 1.11	87.60 ± 1.11	89.14 ± 1.08	88.83 ± 0.93	87.19 ± 1.01	89.45 ± 1.06	**92.34 ± 0.71**
50%	83.46 ± 1.15	85.21 ± 1.14	88.90 ± 1.04	90.71 ± 0.97	90.54 ± 1.04	90.08 ± 1.00	91.01 ± 0.97	**93.65 ± 0.83**
60%	84.29 ± 1.13	85.93 ± 1.23	89.73 ± 0.95	91.99 ± 0.93	92.19 ± 0.82	91.89 ± 0.98	92.59 ± 0.99	**94.74 ± 0.87**

nearest neighbors k and the parameter λ for local weight decay term are respectively searched from $\{5, 10, 15\}$ and $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$ as suggested by Yang et al. (2012).

- LNP (Wang & Zhang, 2008). We follow the pipeline of linear label propagation in Wang and Zhang (2008) to construct the graph. The neighborhood size in LNP is set to 40 to achieve the best results.
- ℓ_1-graph (Lu et al., 2012; Yan & Wang, 2009). The ℓ_1-norm regularization parameter λ is searched from $\{10^{-3}, 10^{-2}, 10^{-1}, 1\}$ to generate the best results. *l1-ls* package (Koh, Kim, & Boyd, 2007) is used to solve ℓ_1-norm regularized least square problem.
- SPG graph (He et al., 2011). We implement the SPG algorithm by setting n_{knn} as 10% of the size of data set and λ is searched from $\{10^{-3}, 10^{-2}, \ldots, 10\}$.

Similar to previous experiments, we randomly select 10% to 60% samples per class as labeled samples and the rest as unlabeled samples. For each configuration, we conduct 50 trials for each model. Table 3 shows the results obtained from the above mentioned models on the four data sets. The mean accuracies as well as standard deviations are reported in this table in which the best results are shown in boldface.

Roughly, these eight models can be categorized into four groups based on their different characteristics.

- KNN1 and KNN2 graphs are in the first group. They adopt the 'HeatKernel' to calculate the graph edge weights in which the performance may be significantly affected by the kernel parameter (variance σ). If the intrinsic structure of data is not suitable for being measured by Euclidean distance or data is corrupted, the performance of KNN graphs will be decreased;
- LSR, LRGA and LNP graphs are in the second group. Both LSR and LRGA use the 'from-local-to-global' scheme to learn the Laplacian matrix instead of constructing it based on 'HeatKernel' function. The local models from all the data points are accumulated to form a robust Laplacian matrix for semi-supervised classification. The main difference between these two models is that the local model in LSR is based on the spline regression while in LRGA it is a general linear regression. LNP constructs the graph by using the neighborhood information of each point instead of considering the pairwise relationships. These three models need not to tune the variance parameter in the 'HeatKernel' function and obtain great improvements over the KNN graphs.
- ℓ_1-graph and SPG graph are in the third group. These two models are based on sparse representation model. ℓ_1-graph calculates the graph weights based on the sparse coefficients. Recent studies have shown that sparsity is a desirable property for building an informative graph. SPG graph additionally imposes the non-negativity property on the sparse weights of a graph and thus it usually obtains higher accuracies than ℓ_1-graph. According to our experimental results, SPG graph is superior to ℓ_1-graph on ORL, CMU PIE and ISOLET data sets.
- MLRR graph is in the fourth group. The main framework of MLRR graph is based on LRR in which the low rankness constraint has grouping effect for samples from the same class and thus can capture the global structure. Furthermore, the introduced manifold regularizer can preserve the local manifold information. Since each sample can be used to represent itself, there always exist feasible solutions even when data sampling is insufficient. MLRR inherits the non-negativity and sparsity properties, which have been proven to be efficient in constructing a desirable graph. Therefore, MLRR achieves outstanding results in all four data sets, especially when given a small amount of labeled samples.

4.5. Computational complexity analysis

Besides the qualitative computational complexity analysis in Section 3.5, we give quantitative evaluation of the time cost of MLRR in this section. Specifically, we compare MLRR with other

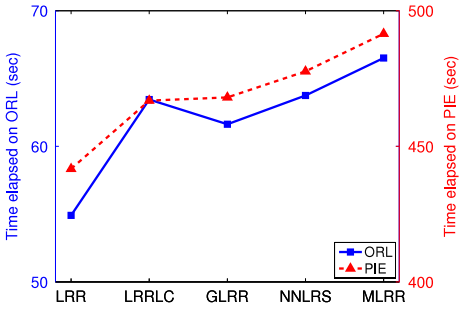


Fig. 13. Comparison of elapsed time of LRR variants on ORL and CMU PIE.

LRR variants on ORL and CMU PIE data sets. The computational platform used in our experiments is *Intel(R) Core(TM) i7-3770 CPU@3.40 GHz 16.0GB RAM* with *Windows 7 systems* and *Matlab 2013a*.

Fig. 13 shows the elapsed time of different LRR variants on ORL and CMU PIE data sets. As shown in this figure, LRR has the least time consuming and the four LRR variants expend approximately the same amount of time. Therefore, MLRR can obtain much accuracy improvement with negligible extra time. Similar results can be found on Extended Yale B and ISOLET data sets.

5. Conclusion

This paper presented a manifold low-rank representation model, which can explicitly consider the data manifold in calculating low-rank representation coefficients, for graph-based semi-supervised learning. The main contribution of MLRR lies in three aspects: (1) Instead of directly calculating the affinity matrix based on some similarity measures, MLRR employed the sparse manifold adaption method to simultaneously do neighborhood selection and edge weight optimization by solving a sparse representation objective; (2) The connection of manifold information between original data space and low-rank representation coefficient space was analyzed and thus it was reasonable to put the learned graph weights in the original data space into the LRR coefficient space; (3) MLRR additionally imposed the sparsity and non-negativity properties on LRR coefficients, which can be efficiently implemented in the LADMAP framework. Experimental results on four popular data sets showed that MLRR is a competitive model for graph-based semi-supervised learning.

MLRR is actually a two-stage method in which the first stage is to identify the manifold by sparse manifold adaption and the second stage is to learn the effective representation coefficient by optimizing a manifold regularized low-rank representation model. The first stage is an unsupervised learning task; therefore, the grid search is used to select the near optimal parameters. In the second stage, a commonly-used paradigm (He et al., 2011; Yan & Wang, 2009; Zhuang et al., 2012), which randomly sampled different percentage of data points as the labeled samples and the left were used as unlabeled data, was followed to quantitatively evaluate different graph-based semi-supervised learning methods. As a shortcoming of MLRR, we lack unified object criteria to tune related parameters, especially in the case of semi-supervised learning in which the number of labeled samples is often limited. Since the labeled samples in different trials are different, the cross-validation bases method is difficult to use. Therefore, we try to minimize the bias when conducting experiments between MLRR and the other methods from two aspects: (1) investigating the mean performance of several independent trials; and (2) searching the near optimal parameters for all comparing methods to make them achieve the best results.

Acknowledgments

This work was partially supported by the National Basic Research Program of China (No. 2013CB329401), the National Natural Science Foundation of China (No. 61272248), the Science and Technology Commission of Shanghai Municipality (No. 13511500200) and the European Union Seventh Framework Program (No. 247619). The first author was supported by China Scholarship Council (No. 201206230012).

Appendix A. Optimization of problem (19)

We introduce an auxiliary variable $\mathbf{r}_i$ w.r.t. $\mathbf{c}_i$ and neglect the subscript i. Therefore, problem (19) can be rewritten as

$$\min_{\mathbf{c}} \frac{1}{2}\|\mathbf{U}\mathbf{r}\|_2^2 + \gamma\|\mathbf{c}\|_1, \quad s.t.\ \mathbf{1}^T\mathbf{c} = 1,\ \mathbf{r} = \mathbf{c}. \tag{A.1}$$

The corresponding augmented Lagrangian function is

$$\mathcal{L}(\mathbf{r}, \mathbf{c}, \lambda, \boldsymbol{\Lambda}, \mu) = \frac{1}{2}\|\mathbf{U}\mathbf{r}\|_2^2 + \gamma\|\mathbf{c}\|_1 + \lambda(1 - \mathbf{1}^T\mathbf{r}) + \langle \Lambda, \boldsymbol{r} - \boldsymbol{c}\rangle + \frac{\mu}{2}\|\boldsymbol{r} - \boldsymbol{c}\|_F^2. \tag{A.2}$$

We need to update $\mathbf{r}$, $\mathbf{c}$ alternately with the other variable fixed and the updating rules are listed below:

- Updating $\mathbf{r}$ as

$$\mathbf{r} = (\mathbf{U}^T\mathbf{U} + \mu\mathbf{I})^{-1}(\mu\mathbf{c} - \Lambda + \lambda\mathbf{1}); \tag{A.3}$$

- Updating $\mathbf{c}$ as

$$\mathbf{c} = \arg\min_{\mathbf{c}} \frac{\gamma}{\mu}\|\mathbf{c}\|_1 + \frac{1}{2}\|\mathbf{c} - \left(\frac{\Lambda}{\mu} + \mathbf{r}\right)\|_F^2, \tag{A.4}$$

  which can be solved via soft shrinkage operator described in Appendix B.

- Updating Lagrangian multipliers and related parameter as

$$\begin{aligned} &\Lambda = \Lambda + \mu(\mathbf{z} - \mathbf{c}),\\ &\lambda = \lambda + \mu(1 - \mathbf{1}^T\mathbf{r}),\\ &\mu = \min(\rho\mu, \mu_{\max}). \end{aligned} \tag{A.5}$$

Appendix B. Three operators

1. Soft thresholding (shrinkage) operator is defined as

$$\mathcal{S}_\varepsilon[x] \doteq \begin{cases} x - \varepsilon, & \text{if } x > \varepsilon,\\ x + \varepsilon, & \text{if } x < -\varepsilon,\\ 0, & \text{otherwise,} \end{cases} \tag{B.1}$$

   where $x \in \mathbb{R}$ and ε is a small positive value. This operator can be extended to vectors and matrices by applying it element-wisely as

$$\mathcal{S}_\varepsilon[\mathbf{W}] = \arg\min_{\mathbf{X}} \varepsilon\|\mathbf{X}\|_1 + \frac{1}{2}\|\mathbf{X} - \mathbf{W}\|_F^2. \tag{B.2}$$

   Therefore, the solutions to (A.4) and (32) are $\mathcal{S}_{\frac{\gamma}{\mu}}\left[\frac{\Lambda}{\mu} + \mathbf{r}\right]$ and $\mathcal{S}_{\frac{\alpha}{\mu_k}}\left[\mathbf{Z}_{k+1} + \frac{\mathbf{Y}_{2,k}}{\mu_k}\right]$.

2. Singular value thresholding (SVT) operator is defined as

$$\mathcal{O}_\varepsilon[\mathbf{W}] \triangleq \mathbf{U}\mathcal{S}_\varepsilon[\boldsymbol{\Sigma}]\mathbf{V}^T = \arg\min_{\mathbf{X}} \varepsilon\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X} - \mathbf{W}\|_F^2, \tag{B.3}$$

   where $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$ is the SVD decomposition of $\mathbf{W}$. SVT shrinks the singular values of $\mathbf{W}$.

3. $\ell_{2,1}$-norm minimization operator (Liu et al., 2010). Given a matrix $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_i, \ldots]$, if the optimal solution of

$$\Omega_\lambda[\mathbf{Q}] \triangleq \arg\min_{\mathbf{W}} \lambda\|\mathbf{W}\|_{2,1} + \frac{1}{2}\|\mathbf{W} - \mathbf{Q}\|_F^2 \tag{B.4}$$

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

$$\mathbf{W}^*(:, i) = \begin{cases} \dfrac{\|\mathbf{q}_i\| - \lambda}{\|\mathbf{q}_i\|}\mathbf{q}_i, & \text{if } \lambda < \|\mathbf{q}_i\|, \\ 0, & \text{otherwise.} \end{cases} \tag{B.5}$$

References

Cai, J. F., Candès, E. J., & Shen, Z. (2010). A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, *20*, 1956–1982.

Cai, D., He, X., & Han, J. (2011). Locally consistent concept factorization for document clustering. *IEEE Transactions on Knowledge and Data Engineering*, *23*, 902–913.

Cai, D., He, X., Han, J., & Huang, T. S. (2011). Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*, 1548–1560.

Candès, E. J., Li, X., Ma, Y., & Wright, J. (2011). Robust principal component analysis? *Journal of the ACM*, *58*, 11.

Chen, C. F., Wei, C. P., & Wang, Y. C. (2012). Low-rank matrix recovery with structural incoherence for robust face recognition. In *Proceedings of IEEE international conference on computer vision and pattern recognition* (pp. 2618–2625).

Chen, J., Zhou, J., & Ye, J. (2011). Integrating low-rank and group-sparse structures for robust multi-task learning. In *Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 42–50).

Cheng, B., Yang, J., Yan, S., Fu, Y., & Huang, T. S. (2010). Learning with ℓ_1-graph for image analysis. *IEEE Transactions on Image Processing*, *19*, 858–866.

Elhamifar, E., & Vidal, R. (2009). Sparse subspace clustering. In *Proceedings of IEEE international conference on computer vision and pattern recognition* (pp.2790–2797).

Elhamifar, E., & Vidal, R. (2011). Sparse manifold clustering and embedding. In *Proceedings of advances in neural information processing systems* (pp. 55–63).

He, R., Zheng, W. S., Hu, B. G., & Kong, X. W. (2011). Nonnegative sparse coding for discriminative semi-supervised learning. In *Proceedings of IEEE international conference on computer vision and pattern recognition* (pp. 2849–2856).

Ji, S., & Ye, J. (2009). An accelerated gradient method for trace norm minimization. In *Proceedings of international conference on machine learning* (pp. 457–464).

Karasuyama, M., & Mamitsuka, H. (2013). Manifold-based similarity adaptation for label propagation. In *Proceedings of advances in neural information processing systems* (pp. 1547–1555).

Koh, K., Kim, S., & Boyd, S. (2007). *ℓ1_ls: A matlab solver for large-scale ℓ1-regularized least squares problems*. Stanford University.

Lee, J. M. (2012). *Introduction to smooth manifolds*: *Vol. 218*. Springer.

Li, S., & Fu, Y. (2013) Low-rank coding with b-matching constraint for semi-supervised classification. In *Proceedings of international joint conference on artificial intelligence* (pp. 1472–1478).

Lin, Z., Chen, M., & Ma, Y. (2010). The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices, arXiv preprint arXiv:1009.5055.

Lin, Z., Liu, R., & Su, Z. (2011). Linearized alternating direction method with adaptive penalty for low-rank representation. In *Proceedings of advances in neural information processing systems* (pp. 612–620).

Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., & Ma, Y. (2013). Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*, 171–184.

Liu, G., Lin, Z., & Yu, Y. (2010). Robust subspace segmentation by low-rank representation. In *Proceedings of international conference on machine learning* (pp. 663–670).

Liu, G., & Yan, S. (2012). Active subspace: toward scalable low-rank learning. *Neural Computation*, *24*, 3371–3394.

Lu, C., Feng, J., Lin, Z., & Yan, S. (2013). Correlation adaptive subspace segmentation by trace Lasso. In *Proceedings of IEEE international conference on computer vision* (pp. 1345–1352).

Lu, X., Wang, Y., & Yuan, Y. (2013). Graph-regularized low-rank representation for destriping of hyperspectral images. *IEEE Transactions on Geoscience and Remote Sensing*, *15*, 4009–4018.

Lu, J., Zhou, X., Tan, Y. P., Shang, Y., & Zhou, J. (2012). Cost-sensitive semi-supervised discriminant analysis for face recognition. *IEEE Transactions on Information Forensics and Security*, *7*, 944–953.

Luo, D., Nie, F., Ding, C., & Huang, H. (2011). Multi-subspace representation and discovery. In *Proceedings of the European conference on machine learning and principles and practice of knowledge discovery in databases* (pp. 405–420).

Nie, F., Xiang, S., Jia, Y., & Zhang, C. (2009). Semi-supervised orthogonal discriminant analysis via label propagation. *Pattern Recognition*, *42*, 2615–2627.

Nie, F., Xu, D., Tsang, I. W. H., & Zhang, C. (2010). Flexible manifold embedding: a framework for semi-supervised and unsupervised dimension reduction. *IEEE Transactions on Image Processing*, *19*, 1921–1932.

Okutomi, M., Yan, S., Sugimoto, S., Liu, G., & Zheng, Y. (2012). Practical low-rank matrix approximation under robust ℓ_1-norm. In *Proceedings of IEEE international conference on computer vision and pattern recognition* (pp. 1410–1417).

Peng, Y., Wang, S., Wang, S., & Lu, B. L. (2013). Structure preserving low-rank representation for semi-supervised face fecognition. In *Proceedings of international conference on neural information processing* (pp. 148–155).

Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290*, 2323–2326.

Shen, B., & Si, L. (2010). Non-negative matrix factorization clustering on multiple manifolds. In *Proceedings of AAAI conference on artificial intelligence* (pp. 575–580).

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267–288.

Wang, F., & Zhang, C. (2008). Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, *20*, 55–67.

Wright, J., Ma, Y., Mairal, J., Sapiro, G., Huang, T. S., & Yan, S. (2010). Sparse representation for computer vision and pattern recognition. *Proceedings of IEEE*, *98*, 1031–1044.

Xiang, S., Nie, F., & Zhang, C. (2010). Semi-supervised classification via local spline regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*, 2039–2053.

Yan, S., & Wang, H. (2009). Semi-supervised learning by sparse representation. In *Proceedings of SIAM international conference on data mining* (pp. 792–801).

Yang, Y., Nie, F., Xu, D., Luo, J., Zhuang, Y., & Pan, Y. (2012). A multimedia retrieval framework based on semi-supervised ranking and relevance feedback. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*, 723–742.

Yang, S., Wang, X., Wang, M., Han, Y., & Jiao, L. (2013). Semi-supervised low-rank representation graph for pattern recognition. *IET Image Processing*, *7*, 131–136.

Yang, J., & Yuan, X. (2013). Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization. *Mathematics of Computation*, *82*, 301–329.

Yang, A., Zhou, Z., Balasubramanian, A., Sastry, S., & Ma, Y. (2013). Fast ℓ1-minimization algorithms for robust face recognition. *IEEE Transactions on Image Processing*, *22*, 3234–3246.

Yu, K., Zhang, T., & Gong, Y. (2009). Nonlinear learning using local coordinate coding. In *Proceedings of advances in neural information processing systems* (pp. 2223–2231).

Zhang, T., Ji, R., Liu, W., Tao, D., & Hua, G. (2013). Semi-supervised learning with manifold fitted graphs. In *Proceedings of international joint conference on artificial intelligence* (pp. 1896–1902).

Zhang, Z. Y., & Zha, H. Y. (2004). Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM Journal on Scientific Computing*, *26*, 313–338.

Zheng, M., Bu, J., Chen, C., Wang, C., Zhang, L., Qiu, G., & Cai, D. (2011). Graph regularized sparse coding for image representation. *IEEE Transactions on Image Processing*, *20*, 1327–1336.

Zheng, Z., Zhang, H., Jia, J., Zhao, J., Guo, L., Fu, F., & Yu, M. (2013). Low-rank matrix recovery with discriminant regularization. In *Proceedings of Pacific-Asia conference on knowledge discovery and data mining* (pp. 437–448).

Zheng, Y., Zhang, X., Yang, S., & Jiao, L. (2013). Low-rank representation with local constraint for graph construction. *Neurocomputing*, *122*, 398–405.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., & Schölkopf, B. (2004). Learning with local and global consistency. In *Proceedings of advances in neural information processing systems* (pp. 321–328).

Zhu, X. (2008). *Semi-supervised learning literature survey, Technical Report*. University of Wisconsin-Madison.

Zhuang, L., Gao, H., Lin, Z., Ma, Y., Zhang, X., & Yu, N. (2012). Non-negative low rank and sparse graph for semi-supervised learning. In *Proceedings of IEEE international conference on computer vision and pattern recognition* (pp. 2328–2335).