Tensor-based Low Rank Graph with Multi-manifold Regularization for Dimensionality Reduction of Hyperspectral Images

Jinliang An, Xiangrong Zhang, *Senior Member, IEEE,* Huiyu Zhou, Licheng Jiao, *Fellow, IEEE*

***Abstract*—Dimensionality reduction is an essential task in hyperspectral image processing. How to preserve the original intrinsic structure information and enhance the discriminant ability is still a challenge in this area. Recently, with the advantage of preserving global intrinsic structure information, low rank representation has been applied to dimensionality reduction and achieved promising performance. By exploiting the sub-manifolds information of the original dataset, multi-manifold learning is effective in enhancing the discriminant ability of the processed dataset. In addition, due to the ability of preserving the spatial neighborhood structure information, tensor analysis has become a popular technique for hyperspectral image processing. Motivated by the above analysis, a novel tensor-based low rank graph with multi-manifold regularization (T-LGMR) for dimensionality reduction of hyperspectral images is proposed in this paper. In T-LGMR, a low rank constraint is employed to preserve the global data structure while multi-manifold information is utilized to enhance the discriminant ability and tensor representation is used to preserve the spatial neighborhood information. Finally, dimensionality reduction is achieved in the graph embedding framework. Experimental results on three real hyperspectral datasets demonstrate the superiority of the proposed method over several state-of-the-art approaches.**

***Index Terms*—Dimensionality reduction, hyperspectral images classification, tensor processing, graph embedding.**

I. Introduction

With the development of advanced imaging techniques, hyperspectral sensors can simultaneously measure hundreds of narrow contiguous bands spanning over the visible-to-infrared spectrum. Hyperspectral images can provide rich spatial and spectral information, and have been used in many fields, such as classification [1], [2], target detection [3], [4], anomaly detection [5], [6], and others [7], [8]. On the other hand, hyperspectral images contain overwhelming spectral bands, which may lead to the curse of dimensionality especially when the training samples are scarce [9]. And the high spectral dimensionality may cause significant increase in the computational time and data storage. Consequently, dimensionality reduction is an essential task in hyperspectral images processing. The goal of dimensionality reduction is to reduce the dimensionality of the feature space while the desired intrinsic information is preserved.

Popularly used dimensionality reduction methods include unsupervised method, such as Principal Component Analysis (PCA) [10], which aims at maximizing the mutual information between the original high-dimensional dataset and the supervised methods, such as Linear Discriminant Analysis (LDA) [11], which targets at finding a projection matrix to minimise the trace of the between-class scatter matrix whilst minimising the trace of the within-class scatter matrix in the projected subspace simultaneously. In addition, numerous dimensionality reduction methods have been proposed and achieved promising performance. But how to preserve the original intrinsic structure information and enhance the discriminant ability is still a challenge in this area.

The low rank property has been proved to be useful in preserving global data structures [12]. Recently, low rank representation (LRR) [13] based methods have been successfully applied to hyperspectral images processing [14]–[16]. LRR aims at finding the lowest rank representation of the data with an appropriate dictionary. Compared with other representation methods, such as sparse representation (SR) [17], LRR is robust against noise and can make full use of the high spectral correlation of hyperspectral images. Nevertheless, LRR fails to exploit the local geometrical structures information and this may degrade the performance in some applications.

Researchers have pointed out that the human brain represents the real world's perceptual stimuli in a manifold way and encodes high dimensional signals in an intrinsically low dimensional structure [18]. Based on this theory, many manifold learning methods which can preserve global or local geometrical properties of the original input data have been proposed for dimensionality reduction [19], classification [20] and other applications [21]. Among these manifold based methods, the most well-known ones are isometric feature mapping (ISOMAP) [22], local linear embedding (LLE) [23] and Laplacian Eigenmap [24]. He et al. [25] proposed the Locality Preserving Projections (LPP), which is a linearization method derived from the Laplacian Eigenmap by preserving the local similarity between data points. But classical LPP is an un-supervised method which does not use the class label information, so the discriminability of the projection matrix derived from LPP is limited.

By presenting each data point as a vertex and regarding the similarity of pairwise data as the corresponding edges which link vertex pairs, graph based methods have been successfully applied in dimensionality reduction [26]. In [27], a Laplacian regularized collaborative graph-based discriminant analysis (LapCGDA) framework was proposed. LapCGDA can offer collaborative representation and exploit the intrinsic geometric information by laplacian regularization. Different from Euclidean distance which is usually used to evaluate the similarity between two vectors, a graph-based discriminant

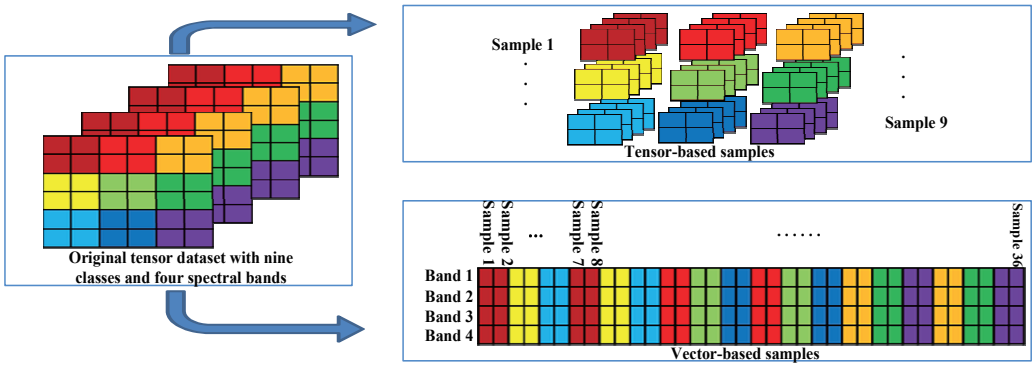


Fig. 1. Vector- and tensor-based representations of the original tensor dataset.

analysis with spectral similarity (GDA-SS) measurement was proposed in [28]. To jointly utilize low rank and sparse properties of the original dataset, a sparse and low rank graph-based discriminant analysis (SLGDA) method was proposed in [29]. Furthermore, two kernel extension methods of SLGDA, the classical kernel SLGDA (cSLGDA) and Nystrom-based kernel SLGDA (nSLGDA) [30] were proposed to enhance the ability of processing complex data with a nonlinear nature.

In general, the methods discussed above assume that the input samples are in the form of vectors. But in many applications, the original data is in a higher order tensor form, such as hyperspectral images and videos. In these cases, we have to convert the high order tensors into vectors firstly. But the spatial information which has been proved to be important for the following processing may be destroyed during the vectorization. For example, as shown in Fig.1, given an original tensor dataset with nine classes and four spectral bands (each color stands for a class and we suppose the four spectral bands are the same for simplicity). For vector-based samples, samples 1, 2, 7 and 8 which belong to the same class may be apart from each other in the sample matrix and the spatial neighborhood assumption is violated. For tensor-based samples, the spatial neighborhood information can be well preserved (see Fig.1).

To address these problems, many tensor-based methods [31]–[33] have been developed. By employing a powerful mathematical framework referred to as multilinear algebra, tensor based techniques have been successfully applied in the field of hyperspectral image processing. A group based low rank tensor model (GTLR) [34] was proposed for hyperspectral image dimensionality reduction. In GTLR, the non-local similarity and the low rankness are jointly considered to obtain the intrinsic structure information of hyperspectral images. Zhang et al. [31] proposed a patch tensor organization scheme and developed tensor discriminative locality alignment (TDLA) to remove redundant information for subsequent classification. Zhong et al. [35] proposed a tensor based spectral-spatial feature extraction method for hyperspectral images. The spectral and spatial features are extracted to generate second-order feature tensors, and then the local tensor discriminant analysis (LTDA) framework is employed to achieve dimensionality reduction for the hyperspectral dataset. By employing the graph bedding framework, a class-aware tensor neighborhood graph and patch alignment (CTNGPA) method was proposed for hyperspectral images dimensionality reduction [36]. In CTNGPA, a class-ware tensor neighborhood graph containing discriminative information is constructed using a tensor distance criterion. Then a patch alignment framework is employed to obtain the optimal projection matrix. By jointly considering the tensor characteristic and the tensor-based neighborhood information, CTNGPA can simultaneously explore both local spectral and spatial information of the hyperspectral data. Tensor locality preserving projection (TLPP) was proposed for hyperspectral images classification [37], which can effectively embed both spatial structures and spectral information into low-dimensional space simultaneously by a series of projection matrices trained for each mode of input samples. In [38], a low rank tensor recovery problem was formulated by using tensor singular value decomposition (tSVD), i.e.tensor tubal rank and tensor nuclear norm. Wei et al. [39] proposed a hierarchical feature learning method called Stacked Tensor Subspace Learning (STSL) for hyperspectral image classification. STSL can learn discriminative spectral-spatial features of the hyperspectral images at different scales.

In summary, exploiting intrinsic structure information and enhancing the discriminant ability are two important issues in tensor processing. But in the available tensor-based methods, these two issues are usually considered separately. In the proposed method, we consider the two issues in one framework, where the low rank constraint is utilized to exploit the global structure information whilst the multi-manifold is employed to enhance the discriminant ability of the processed hyperspectral dataset.

Jointly considering low rank representation, tensor analysis and multi-manifold information, a novel tensor-based low rank graph with multi-manifold regularization for dimensionality reduction of hyperspectral images is proposed in this paper. For simplicity, the proposed method is called T-LGMR. In T-LGMR, the low rank constraint is imposed to keep the global data structures whilst tensor analysis is employed to preserve the spatial neighborhood information. Multi-manifold is utilized to preserve the local geometrical property and

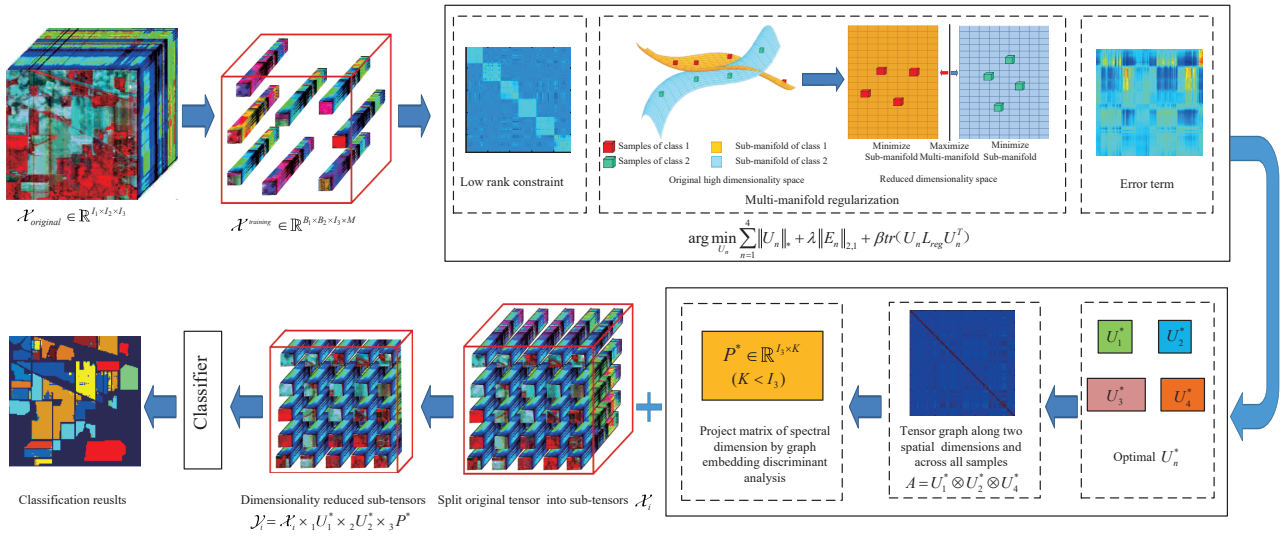


Fig. 2. Illustration of T-LGMR.

enhance the discriminability. In general, the proposed method can preserve the local and globe data structure simultaneously and enhance the discrimination. Compared with the established approaches, our proposed method has three new contributions: 1) Unlike traditional vector-based methods which treat each sample as an independent and identically distributed item, the samples in T-LGMR are represented in a tensor form which can preserve the original spatial neighborhood information. In addition, by adopting tensor training samples, only a small set of the labeled training samples are needed in T-LGMR. 2) With the assumption that the samples belonging to the same class lie on a unique sub-manifold, T-LGMR constructs tensor-based within-class and between-class graphs to characterize the within-class compactness and the between-class separability which make the resulting graphs more discriminative. 3) Different from the available vector-based graphs, the proposed tensor-based graph can exploit the geometric information of tensor samples along the spatial and spectral dimensions, which makes the resulting graph more informative. In summary, the learned tensor-based graph jointly utilizes spatial neighborhoods, discriminative and low rankness information which capture the local and global structures as well as the discriminative information simultaneously and make the resulting graph more robust and discriminative.

The remainder of this paper is organized as follows. In section II, we introduce the established work related to the proposed method. The proposed method is discussed in Section III. In Section IV, experiments are undertaken to evaluate the performance of the proposed method. The conclusion is finally given in Section V.

II. RELATED WORK

In this section, the related work is introduced to illustrate the theoretical origin and the difference between the proposed method and the available methods.

A. LPP and Multi-Manifold Discriminant Analysis

1) LPP: LPP aims at preserving the local geometry structure of the input data by building a graph incorporating neighborhood information. Let $X = [x_1, x_2, \cdots, x_m]$ be the input data, P is a projection matrix which maps the input data to a low dimensionality space and $Y = [y_1, y_2, \cdots, y_m]$ is the projected data. The objective function is described as follows:

$$\min \sum_{i,j} \|y_i - y_j\|^2 W_{ij} \tag{1}$$

where W_{ij} is the weight matrix which can be defined as:

$$W_{ij} = \begin{cases} \exp(-\|x_i - x_j\|^2/t) & \text{if } \mathrm{x_i} \text{ and } \mathrm{x_j} \text{ are connected} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where parameter $t > 0$. The objective function incurs a heavy penalty if inputs x_i and x_j are far apart. Therefore, minimizing Eq.(1) is an attempt to ensure that if x_i and x_j are close, then y_i and y_j are close as well. This means the local intrinsic geometrical structure can be preserved in the low dimensionality subspace. To seek the best solution, an extra constraint $P^T X D X^T P = I$ is imposed. Then, the minimization problem is given as

$$\begin{aligned} &\arg\min_P \ tr(P^T X L X^T P) \\ &s.t.\ P^T X D X^T P = I \end{aligned} \tag{3}$$

where $L = D - W$ is the Laplacian matrix, D is a degree matrix with $D_{ii} = \sum_j W_{ij}$. Then the projection matrix P is given by solving the following generalized eigenvalue problem:

$$X L X^T P = \lambda X D X^T P \tag{4}$$

2) MMDA: By using sub-manifold and multi-manifold information to enhance the discrimination, multi-manifold discriminate analysis (MMDA) was introduced in [40]. In MMDA, sub-manifold is represented by a within-class graph while multi-manifold is represented by a between-class graph.

For the within-class graph, the weight matrix is defined as:

$$W_{ij}^{w} = \begin{cases} \exp(-\|x_i - x_j\|^2/t) & \text{if labels } l_i = l_j \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

The within-class graph-preserving criterion is defined as

$$\arg\min \ P^T X L_w X^T P \quad (6)$$

where $L_w = D^w - W^w$ is a Laplacian matrix, D^w is a diagonal degree matrix with $D_{ii}^w = \sum_j W_{ij}^w$. For the between-class graph, after calculating the classes' centers $H = [\tilde{h}_1, \tilde{h}_2, \cdots, \tilde{h}_C]$, where $\tilde{h}_i$ is the mean value of the samples belonging to i-th class, C is the number of classes, and the weight matrix between the class centers is defined as:

$$W_{ij}^{b} = \exp(-\left\|\tilde{h}_i - \tilde{h}_j\right\|^2/t) \quad (7)$$

The between-class graph-penalizing criterion is defined as

$$\arg\max \ P^T H L_b H^T P \quad (8)$$

where $L_b = D^b - W^b$ is a Laplacian matrix, and D_b is a diagonal degree matrix with $D_{ii}^b = \sum_j W_{ij}^b$. The objective function of MMDA can be expressed as

$$P = \arg\max_P \frac{P^T H L_b H^T P}{P^T X L_w X^T P} \quad (9)$$

The projection matrix P can be obtained by solving the generalized eigenvalue problem.

Compared with LPP, MMDA can offer more discriminative information by explicitly considering the sub- and multi-manifold structures as well as the label information, which is important for the classification and recognition tasks. Furthermore, in order to further exploit the neighborhood information of the input data, the MMDA will be extended in a tensor form in the proposed method.

B. LRR and tensor LRR

1)LRR: In order to capture the global structure of the input data, Liu et al. [13] proposed the low rank representation and developed the LRR graph to render the global data structure by imposing a global low rank constraint.

Given a set of samples X, LRR aims at finding the lowest rank representation of X with a dedicated dictionary, and in real applications, X itself is usually used as the dictionary, so the basic objective function of LRR is as follows:

$$\begin{aligned} &\min_Z \ \ rank(Z) \\ &s.t. \ \ X = XZ \end{aligned} \quad (10)$$

where Z is the low rank coefficient matrix. Due to the discrete nature of the rank function, it is difficult to solve the optimization problem shown in Eq. (10). Zhang et al. [41] have proved that Eq. (10) can be relaxed to a convex optimization problem as follows:

$$\begin{aligned} &\min_Z \ \ \|Z\|_* \\ &s.t. \ \ X = XZ \end{aligned} \quad (11)$$

where $\|\cdot\|_*$ denotes the nuclear norm of a matrix. In a real application, the observations are often noisy or corrupted. By adding a noise term, Eq.(11) can be reformulated as

$$\begin{aligned} &\min_{Z,E} \ \ \|Z\|_* + \lambda\|E\|_{2,1} \\ &s.t. \ \ X = XZ + E \end{aligned} \quad (12)$$

where E is the noise in the observation and $\|E\|_{2,1} = \sum_{j=1}^{n}\sqrt{\sum_{i=1}^{m}([E]_{i,j})^2}$ is $\ell_{2,1}$-norm [42]. The parameter $\lambda > 0$ is used to compromise the outcome for better optimisation.

2) Tensor LRR: Recently, low rank representation algorithms have been extended to a tensor form and achieved promising performance in many applications [43]–[45]. Fu et al. [46] proposed tensor low rank representation for subspace clustering which aims at looking for a lowest rank representation over all the candidates while maintaining the inherent spatial structures between the samples. The affinity matrix used for spectral clustering is built from the combination of similarities along all the spatial directions. In [47], Fu et al. proposed a tensor based low rank representation and sparse coding-based (TLRRSC) subspace clustering method. In TLRRSC, tensor based low rank representation is used in order to obtain the lowest rank representation of the original dataset along all the spatial directions and sparse coding is used to learn a dictionary in the feature space thus the samples can be represented by a few atoms. TLRRSC can capture the global structures and the inherent feature information of the data and provide a robust subspace segmentation for the corrupted data. Zhang et al. [48] proposed a Low rank Tensor constrained Multi-view Subspace Clustering (LT-MSC) method. In LT-MSC, a low rank constraint is imposed on the tensor samples whilst considering the cross information between the different views in order to improve the clustering accuracy.

Vector- and tensor-based low rank representation methods with discriminant criteria have also been published in the research community. For example, Li et al. [12] proposed a sparse and low rank graph-based discriminant analysis (SLGDA) method. In SLGDA, an informative graph is constructed by combining both sparsity and low rankness to maintain global and local structures simultaneously and the discrimination is improved by introducing a Laplacian graph. Jia et al. [49] proposed a low rank tensor completion method for action classification and image recovery. This method integrates the global low rank and discriminative information by introducing inter-class and intra-class scatter matrices of tensor samples.

III. PROPOSED METHOD

A. Definitions and Notations

An N-dimensionality array can be represented as an N-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, where I_n $(1 \le n \le N)$ is the *n*-mode dimensionality. The entries of $\mathcal{X}$ are represented as $\mathcal{X}_{i_1 i_2 \cdots i_n \cdots i_N}$, where i_n $(1 \le i_n \le I_n)$ is the *n*-mode index. Before formulating the proposed method, we first introduce the main definitions and notations used in this paper.

Definition 1: (*n*-mode vector and *n*-mode flattening matrix): The *n*-mode vector of $\mathcal{X}$ is an *n*-dimensional vector by varying index i_n whilst keeping the other indices fixed. Taking all the

n-mode vectors as columns, the obtained matrix is a *n*-mode flattening matrix and denoted by

$$X_n \in \mathbb{R}^{I_n \times (I_1 \times I_2 \times \cdots I_{n-1} \times I_{n+1} \times \cdots \times I_N)} \tag{13}$$

Definition 2: (*n*-mode product): The *n*-mode product refers to a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots I_n \times \cdots \times I_N}$ times by a matrix $U \in \mathbb{R}^{J \times I_n}$, denoted by $\mathcal{Y} = \mathcal{X} \times_n U$, where $\mathcal{Y} \in \mathbb{R}^{I_1 \times \cdots \times J \times \cdots \times I_N}$ is a tensor with entries $\mathcal{Y}_{i_1 i_2 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n=1}^{I_n} x_{i_1 i_2 \cdots i_{n-1} i_n i_{n+1} \cdots i_N} u_{j i_n}$. The *n*-mode product is also denoted in terms of the tensor matrix

$$\mathcal{Y} = \mathcal{X} \times_n U \Leftrightarrow Y_n = U X_N \tag{14}$$

Definition 3: (Kronecker product): The Kronecker product of matrix $A \in \mathbb{R}^{I \times J}$ by matrix $B \in \mathbb{R}^{K \times L}$ is a matrix denoted by

$$Y = A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1J}B \\ a_{21}B & a_{22}B & \cdots & a_{2J}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}B & a_{I2}B & \cdots & a_{IJ}B \end{bmatrix} \tag{15}$$

where $Y \in \mathbb{R}^{(I \times K) \times (J \times L)}$.

Definition 4: (Tucker decomposition): Tucker decomposition is denoted by

$$\begin{aligned} \mathcal{Y} &= \mathcal{X} \times_1 U_1 \times_2 U_2 \times \cdots \times_N U_N \\ &= \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \cdots \sum_{r_N=1}^{R_N} x_{r_1 r_2 \cdots r_N} u_{1 r_1} \circ u_{2 r_2} \circ \cdots \circ u_{N r_N} \end{aligned} \tag{16}$$

where "o" is the outer product of the vector, $\mathcal{X} \in \mathbb{R}^{R_1 \times R_2 \times \cdots \times R_N}$ is called the core tensor, $U_n \in \mathbb{R}^{I_n \times R_n} (1 \leq n \leq N)$ are factor matrices along each mode and $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is an approximate tensor under a certain criterion.

Definition 5: (Tensor Frobenius norm): The Frobenius norm of tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is given by

$$\|\mathcal{X}\| = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} x_{i_1 i_2 \cdots i_N}^2} \tag{17}$$

B. Tensor Based Multi-manifold Discriminant Analysis

As discussed above, MMDA can exploit the sub-manifolds information and enhance the discriminant ability. In this section, we extend the MMDA to a tensor form (T-MMDA).

Manifold learning framework is under the basic assumption that the high-dimensionality data lies on a smooth low-dimensionality manifold. Furthermore, for a high-dimensionality dataset with different classes of objects, each class lies on a unique manifold which is called sub-manifold and all the sub-manifolds form a multi-manifold. Let $\{\mathcal{X}_i \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \ i = 1, \cdots, M\}$ be a set of tensor samples belonging to C classes, where M is the number of the training samples. With these tensor samples, we can construct a tensor-based within-class graph which characterizes the sub-manifold or the compactness within a class and a tensor-based between-class graph which characterizes the multi-manifold or the separation between different classes. The within-class graph is constructed as follows:

$$\arg\min_{U_n} \sum_{i=1}^{M} \sum_{j=1}^{M} \|\mathcal{Y}_i - \mathcal{Y}_j\|^2 W_{ij} \tag{18}$$

where $\mathcal{Y}_i = \mathcal{X}_i \times_1 U_1 \times \cdots \times_N U_N$ is the projected data of $\mathcal{X}_i$ in a low-dimensionality space. $W_{ij}$is the weight matrix which is defined as follows:

$$W_{ij} = \begin{cases} \exp(-\|\mathcal{X}_i - \mathcal{X}_j\|^2 / t) & \text{if class label } l_i = l_j \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

The between-class graph is defined as

$$\arg\max_{U_n} \sum_{i=1}^{C} \sum_{j=1}^{C} \left\| \widetilde{\mathcal{Y}}_i - \widetilde{\mathcal{Y}}_j \right\|^2 B_{ij} \tag{20}$$

where $\widetilde{\mathcal{Y}}_i = \widetilde{\mathcal{X}}_i \times_1 U_1 \times \cdots \times_N U_N$ is the projected tensor of $\widetilde{\mathcal{X}}_i$ in a low-dimensionality space and $\widetilde{\mathcal{X}}_i$ is the mean tensor of the tensor samples belonging to the i-th class. B_{ij} is the weight matrix of different classes, which is defined as follows:

$$B_{ij} = \exp(-\left\| \widetilde{\mathcal{X}}_i - \widetilde{\mathcal{X}}_j \right\|^2 / t) \tag{21}$$

The optimization problems shown in Eq. (18) and (20) are high-order nonlinear programming problems which cannot be solved by direct matrix transformation and general eigenvalue decomposition. As discussed in [50], Eq. (18) and (20) can be solved by applying an iteration scheme. Assuming that $U_1, \cdots, U_{n-1}, U_{n+1}, \cdots, U_N$ are known, we aim at finding a solution for U_n . For Eq. (18), we denote $\mathcal{X}_i \times_1 U_1 \times \cdots \times_{n-1} U_{n-1} \times_{n+1} U_{n+1} \times \cdots \times_N U_N$ by $\mathcal{Y}_i^n$ and Y_i^n is the flattening matrix of $\mathcal{Y}_i^n$. Based on the properties of the tensor analysis discussed above, Eq. (18) can be reformulated as follows:

$$\arg\min_{U_n} \ tr\{U_n L_w U_n^T\} \tag{22}$$

where $L_w = \sum_{i=1}^{M} \sum_{j=1}^{M} (Y_i^n - Y_j^n)(Y_i^n - Y_j^n)^T W(i,j)$. Similiarly, we denote $\widetilde{\mathcal{X}}_i \times_1 U_1 \times \cdots \times_{n-1} U_{n-1} \times_{n+1} U_{n+1} \times \cdots \times_N U_N$ by $\widetilde{\mathcal{Y}}_i^n$ and $\widetilde{Y}_i^n$ is the flattening matrix of $\widetilde{\mathcal{Y}}_i^n$. Eq. (20) can be reformulated as:

$$\arg\min_{U_n} \ tr\{U_n L_b U_n^T\} \tag{23}$$

where $L_b = \sum_{i=1}^{C} \sum_{j=1}^{C} (\widetilde{Y}_i^n - \widetilde{Y}_j^n)(\widetilde{Y}_i^n - \widetilde{Y}_j^n)^T B(i,j)$. In a graph embedding framework, the factor matrices U_n should satisfy the following two optimization criteria:

$$\begin{cases} \arg\min_{U_n} \ tr\{U_n L_w U_n^T\} \\ \arg\max_{U_n} \ tr\{U_n L_b U_n^T\} \end{cases} \tag{24}$$

By employing the difference scatter discriminant criterion [51] [52], the objective function of T-MMDA can be rewritten as

$$\arg\min_{U_n} \ tr\{U_n (L_w - \zeta L_b) U_n^T\} \tag{25}$$

where ζ is a tuning parameter to balance the effect of within-class compactness and between-class separation. Here, we set ζ as 1 following the way shown in [53] [54]. Thus, the optimal problem of Eq.(25) is reformulated as

$$\arg\min_{U_n} \ tr\{U_n L_{reg} U_n^T\} \tag{26}$$

where $L_{reg} = L_w - L_b$ is the regularization graph obtained by tensor based multi-manifold discriminant analysis.

C. Tensor LRR with Multi-manifold regularization

Let $\mathcal{X}$ be an *N*-order tensor, and the basic objective function of TLRR is described as follows:

$$\arg\min_{X_n} \ \sum_{n=1}^{N} \|X_n\|_* + \lambda \|E_n\|_{2,1} \tag{27}$$

where X_n and E_n are the n-mode flattening and error matrices respectively. $\ell_{2,1}$-norm encourages the columns of E_n to be zero, which means the corruptions are sample-specific, i.e., some data vectors are corrupted whilst the others are clean (this assumption is reasonable in the processing of hyperspectral images). With the property of tensor operations discussed above, the objective function Eq. (27) can be rewritten as

$$\begin{aligned} &\arg\min_{U_n} \ \sum_{n=1}^{N} \|U_n\|_* + \lambda \|E_n\|_{2,1} \\ &s.t. \ X_n = U_n G_n + E_n \end{aligned} \tag{28}$$

where G_n is the *n*-mode flattening matrix of tensor $(\mathcal{X} \times {}_1U_1 \times \cdots \times {}_{n-1}U_{n-1} \times {}_{n+1}U_{n+1} \times \cdots \times {}_N U_N)$. By solving the optimization problem in Eq. (28), the low rank property of $\mathcal{X}$ can be obtained from U_n. In addition, in order to utilize the sub-manifold and multi-manifold information discussed above, we incorporate the tensor multi-manifold constraint Eq. (26) into the TLRR model and the resulting objective function can be reformulated as follows:

$$\begin{aligned} &F(U_1, U_2, \cdots, U_N) \\ &= \arg\min_{U_n} \sum_{n=1}^{N} \|U_n\|_* + \lambda \|E_n\|_{2,1} + \beta tr\{U_n L_{reg} U_n^T\} \\ &s.t. \ X_n = U_n G_n + E_n \end{aligned} \tag{29}$$

where λ and β are used to balance the effects of the noise and the multi-manifold regularization terms. The minimization of the first term of the objective function is to learn a low rank factor matrix U_n and the low rank representation of tensor $\mathcal{X}$ can be obtained by $\mathcal{X} \times {}_1U_1 \times \cdots \times {}_N U_N$. The second term is the error of the low rank representation whilst the third term is the multi-manifold regularization which encourages the projected samples lying on the same sub-manifold to gather together and those in different sub-manifolds to be apart from each other.

D. Optimization

In this section, the Augmented Lagrange Multiplier (ALM) method [55] is employed to solve the constrained optimization problem Eq.(29). The objective function Eq.(29) can be rewritten as

$$\begin{aligned} &F(U_1, U_2, \cdots, U_N) \\ &= \arg\min_{U_n} \sum_{n=1}^{N} \|J_n\|_* + \lambda \|E_n\|_{2,1} + \beta tr\{Z_n L_{reg} Z_n^T\} \\ &s.t. \ U_n = J_n, \ U_n = Z_n, X_n = U_n G_n + E_n \end{aligned} \tag{30}$$

The Lagrange function Eq.(30) can be written as

$$\begin{aligned} L_n = &\arg\min \sum_{n=1}^{N} \|J_n\|_* + \lambda \|E\|_{2,1} + \beta tr\{Z_n L_{reg} Z_n^T\} \\ &+ tr[Y_1^T (X_n - U_n G_n - E_n)] \\ &+ tr[Y_2^T (U_n - J_n)] + tr[Y_3^T (U_n - Z_n)] \\ &+ \tfrac{\mu}{2} (\|X_n - U_n G_n - E_n\|_F^2 \\ &+ \|U_n - J_n\|_F^2 + \|U_n - Z_n\|_F^2) \end{aligned} \tag{31}$$

where Y_1, Y_2 and Y_3 are Lagrange multipliers, μ is the penalty operator and $tr(\cdot)$ is the trace of a matrix. The optimization problem Eq. (31) can be solved by updating one variable at a time with all the remaining variables fixed.

1) Fix all the remaining and update J_n:

$$J_n = \arg\min \|J_n\|_* + \frac{\mu}{2} \left\| J_n - (U_n + \frac{Y_2}{\mu}) \right\|_F^2 \tag{32}$$

2) Fix all the remaining and update U_n:

$$\begin{aligned} U_n = &(X_n G_n^T - E_n G_n^T + J_n + Z_n) \\ &+ \tfrac{1}{\mu} (Y_1 G_n^T - Y_2 - Y_3)(2I + G_n G_n^T)^{-1} \end{aligned} \tag{33}$$

3) Fix all the remaining and update Z_n:

$$Z_n = (Y_3 + \mu U_n)(\beta(L_{reg} + (L_{reg})^T) + \mu)^{-1} \tag{34}$$

4) Fix all the remaining and update E_n:

$$E_n = \lambda \|E\|_{2,1} + \frac{\mu}{2} (\left\| E_n - (X_n - U_n G_n + \frac{Y_1}{\mu}) \right\|_F^2 \tag{35}$$

5) Update multipliers Y_1, Y_2 and Y_3 :

$$\begin{cases} Y_1 = Y_1 + \mu(X_n - U_n G_n - E_n) \\ Y_2 = Y_2 + \mu(U_n - J_n) \\ Y_3 = Y_3 + \mu(U_n - Z_n) \end{cases} \tag{36}$$

6) Update μ:

$$\mu = min(\rho \mu_0, \mu_{max}) \tag{37}$$

It is noted that, for the application of hyperspeatral image dimensionality reduction, the tensor training samples are 3-order tensors, i.e. $\mathcal{X}_i^{training} \in \mathbb{R}^{B_1 \times B_2 \times I_3}, 1 \leq i \leq M$, where B_1 and B_2 are two spatial sizes of tensor samples, I_3 is the number of the original spectral bands and M is the number of the training samples. By stacking all the tensor samples together, we obtain a 4-order tensor training dataset $\mathcal{X}^{training} \in \mathbb{R}^{B_1 \times B_2 \times I_3 \times M}$. Using this dataset, the tensor low rank representation with the multi-manifold regularization algorithm is outlined in Algorithm 1.

Algorithm 1: Tensor low rank decomposition with multi-manifold regularization for hyperspectral dataset

INPUT: M labeled 3-order tensor training samples $\{\mathcal{X}_i^{training} \in \mathbb{R}^{B_1\times B_2\times I_3}, 1\le i\le M\}$, the parameters λ, β and the maximum number of the training iterations $T_{\max}$.
Initialize U_n as identity matrix. $J_n=0$, $Z_n=0$, $Y_1=Y_2=Y_3=0$, $\mu_0=10^{-6}$, $\mu_{\max}=10^6$, $\rho=1.1$, $\varepsilon=10^{-6}$.
Stack all M tensor samples to obtain 4-order tensor training dataset $\mathcal{X}^{training}\in\mathbb{R}^{B_1\times B_2\times B_3\times M}$.
for $t=1$ to $T_{\max}$ do
  for $n=1$ to 4 do
    calculate L_{reg} by Eqs. (22), (23) and (26).
    update J_n, U_n, Z_n, E_n by Eqs. (32)-(35).
    update Y_1, Y_2 and Y_3 by Eq. (36).
    update μ by Eq. (37).
  end for
    check convergence: if $t>0$ and $\left\|U_n^t-U_n^{t-1}\right\|_F^2<\varepsilon$ for each n, break
end for
OUTPUT: optimal factor matrices U_n^* $(1\le n\le 4)$.

Algorithm 2: Proposed T-LGMR algorithm

INPUT: Original hyperpectral image, tensor training samples spatial size B_1 and B_2, M labeled pixel training samples, the parameters λ, β and the maximum number of the iterations $T_{\max}$.
Construct M labeled tensor training samples using fixed window criterion.
Calculate optimal factor matrices U_n^* by algorithm 1.
Calculate affinity matrix by Eqs. (38) and (39).
Compute the eigenvalue decomposition problem in Eq. (41).
Split original tensor data into sub-tensors using fixed window with size of $B_1\times B_2$.
Calculate dimensionality reduced sub-tensors by Eq. (42).
Rearrange the dimensionality reduced sub-tensors.
OUTPUT: Dimensionality reduced dataset.

E. The complete graph based dimensionality reduction algorithm

In this section, we introduce the proposed graph based dimensionality reduction algorithm for hyperspectral images. Using Algorithm 1, the obtained factor matrices $U_1^*\in\mathbb{R}^{B_1\times B_1}$ and $U_2^*\in\mathbb{R}^{B_2\times B_2}$ reveal the property in the row and column spaces, $U_3^*\in\mathbb{R}^{I_3\times I_3}$ preserves the spectral property and $U_4^*\in\mathbb{R}^{M\times M}$ reflects the joint information of all the training samples. After having obtained the optimal factor matrices $U_n^*(n=1,2,3,4)$, we construct a similarity matrix by

$$A=U_1^*\otimes U_2^*\otimes U_4^* \tag{38}$$

where $A\in\mathbb{R}^{(B_1\times B_2\times M)\times(B_1\times B_2\times M)}$.

Then the affinity matrix can be computed by

$$W=|A|+\left|A^T\right| \tag{39}$$

It should be noted that, different from other image-based applications, such as face recognition, hyperspectral image processing is a pixel-based task. Here we flatten the tensor data $\mathcal{X}\in\mathbb{R}^{I_1\times I_2\times I_3\times M}$ into matrix $X\in\mathbb{R}^{I_3\times(I_1\times I_2\times M)}$ so as to take into account the contribution of each pixel within the tensor training data. Finally, using the framework of graph-embedding for dimensionality reduction [26], we have the Laplacian matrix by $L=D-W$, where $D_{ii}=\sum_j W_{ij}$ and the resulting projection matrix P^* can be obtained by solving the following optimization problem

$$P^*=\arg\min_{P^TP=I}\left|P^TXLX^TP\right| \tag{40}$$

which can be reformulated as an eigenvalue decomposition problem

$$XLX^TP=\Lambda P \tag{41}$$

where Λ is a diagonal eigenvalue matrix. The resulting projection matrix $P^*\in\mathbb{R}^{I_3\times K}$ $(K<I_3)$ is constructed by the K eigenvectors corresponding to the K smallest eigenvalues.

To apply the T-LGMR algorithm to the original tensor hyperspectral dataset, we need to split the original data into sub-tensors in spatial dimension by fixed window with the same size as that of the tensor training samples. The sub-tensors are represented as $\mathcal{X}_i$, $i=1,2,\cdots Q$, where Q is the number of the sub-tensors. Then the corresponding dimensionality reduced data of $\mathcal{X}_i$ can be calculated by

$$\mathcal{Y}_i=\mathcal{X}_i\times_1 U_1^*\times_2 U_2^*\times_3 P^* \tag{42}$$

where $\mathcal{Y}_i$ is the dimensionality reduced dataset of $\mathcal{X}_i$, after the sub-tensors have been rearranged, we obtain the dimensionality reduced dataset.

The proposed algorithm is shown in Algorithm 2.

F. Computational complexity analysis

In this section, we analyze the computation complexity of T-LGMR. After stacking the N-order tensor training samples, we obtain a $(N+1)$-order tensor dataset $\mathcal{X}^{training}\in\mathbb{R}^{I_1\times I_2\times\cdots\times I_N\times I_{N+1}}$. For simplicity, we assume the training dataset is of a uniform size in every dimension, i.e., $I_1=I_2=\cdots=I_N=I_{N+1}=I$, the iteration times is T and the number of classes is C. The proposed method is composed of an iterative updating step and a graph based dimensionality reduction step. In the iterative updating step, for each iteration, the main computational costs regarding computing and updating J_n, U_n, Z_n, E_n are $O((N+1)I^3)$, $O((N+1)I^N)$, $O((N+1)I^{N+2})$, $O((N+1)I^N)$ and $O((N+1)I^{N+2})$ respectively. As $N\ge 3$, the computational complexity of the iterative updating step for calculating U_n $(1\le n\le N+1)$ is approximately $O(T(N+1)I^{N+2})$.

In the graph based dimensionality reduction step, after obtaining the optimal factor matrices $U_n(1\le n\le N+1)$, the time complexity of creating the affinity matrix is $O(I^3)$. With the affinity matrix, the projection matrix can be obtained by solving an eigenvalue composition problem with the complexity of $O(I^3)$. Finally, the reduced dimensionality of the hyperspectral dataset can be calculated with a complexity of $O(Q(I^4+KI^3))$. As $K\ll I$, the computational complexity of this step is approximately $O(QI^4)$.

With the above analysis, the total computational complexity of T-LGMR is $O(T(N+1)I^{N+2}+QI^4)$.

IV. EXPERIMENTAL RESULTS

In this section, we present a number of experiments on three real hyperspectral image datasets to validate the proposed

method. In the following experiments, nearest neighborhood (1NN) and support vector machine (SVM) are employed to classify the hyperspectral datasets with dimensionality reduction in the experiments. The LIBSVM [36] with radial basis function (RBF) kernels is applied in the experiments and the parameters are obtained by cross-validation.

A. Hyperspectral datasets

The proposed method is evaluated on three real hyperspectral datasets, the first dataset was collected by the National Aeronautics and Space Administrations Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor over the northwest Indians Indian Pines in June 1992. The image contains 145×145 pixels and 220 spectral bands in the wave length range of 0.4-2.5m with a spatial resolution of 20m. There are 16 classes land covers in total, after we have removed 20 spectral bands due to noise and water absorption, 200 bands are used in the experiments. The synthetic color and the corresponding ground truth maps are shown in Fig.5 (a) and (b).

The second dataset was collected by the Reflective Optics System Imaging Spectrometer (ROSIS) optical sensor over the urban area of Pavia University, Italy. The spectral range is from 430 to 860 nm with a spatial resolution of 1.3 m. It contains 610×340 pixels and 115 spectral bands. After removing some noisy and water absorption bands, we use 103 spectral bands in the experiments. There are 9 classes land covers in total and its synthetic color and the corresponding ground truth maps are shown in Fig.6 (a) and (b).

The third dataset was collected by the AVIRIS sensor over Salinas Valley, California, which comprises 512×217 pixels and 204 spectral bands after removing 20 noise and water absorption bands. There are 16 classes of land cover in total and its synthetic color and the corresponding ground truth maps are shown in Fig.7 (a) and (b).

B. Comparison algorithms

To evaluate the performance of T-LGMR, several state-of-the-art methods are chosen as the comparison methods.

1) Tensor based dimensionality reduction methods: These include tensor discriminative locality alignment (TDLA) [31], tensor locality preserving projection (TLPP) [37] and tensor low rank representation without any regularization (TLRR). For the TDLA, the spatial size of the tensor samples is set as 9×9, the number of the neighbors samples is set as 4. For the TLPP, the spatial size of the tensor sample is set as 9×9. TLRR is presented to test the effect of the multi-manifold constraint term and the parameters of TLRR are the same as those of T-LGMR. For these tensor-based methods, 5 tensor samples are selected randomly for each class to form the training samples.

2) Vector based dimensionality reduction methods: These include multi-manifold discriminant analysis (MMDA) [40], sparse and low rank graph for discriminant analysis (SLGDA) [12] and Laplacian regularized low rank representation (LAPLRR) [5]. In these three vector-based methods, 10 samples for each class are randomly chosen to form the training samples. In SLGDA, the balance parameters of the sparsity and error constraint terms are set as 0.1 and 0.001 respectively. In LAPLRR, the balance parameters of the Laplacian graph regularization and error constraint terms are set as 0.2 and 0.1 respectively. In addition, the original spectral bands without any processing and the classical LDA with 20% random samples from each class are also chosen as the comparison methods.

C. Preparation of tensor training samples

In tensor-based methods for hyperspectral image processing, how to construct tensor training samples is an important issue. Here, we elaborate the criterion of how to construct training samples in T-LGMR. After we have selected pixel samples randomly, tensor training samples are obtained by using fixed spatial windows with the pixel samples located at the center of the windows. As shown in Fig.5(c), Fig.6(c) and Fig.7(c), the black spots at the center of the white windows are pixel samples and the white windows are tensor samples. In addition, due to the region uniformity of the land covers, the labels of the tensor samples are regarded as the same as the labels of the corresponding center pixel samples. Compared with the pixel training samples, the tensor training samples can provide more geometric structural information and achieve better performance especially when the labeled data is insufficient. Another question about the construction of the tensor samples is the spatial size of the tensor samples. With an increasing spatial size, more pixels will be included within the tensor samples which may offer more spatial and spectral information, but if the spatial size is too large, the pixels within a tensor sample may belong to different classes which may destroy the consistency of spatial and spectral information. There is no fixed rule that can be used to define the spatial size of the tensor samples. In fact, the spatial size depends on the size of the hyperspectral image, the class number of different land covers and the region consistence of different land covers. In practice, this size is usually determined empirically and experimentally.

Here, experiments are undertaken to investigate the effects of the number of the training samples and the spatial size on the performance of T-LGMR. The number of the labeled training samples varies from 1 to 10 for each class for all the three experimental datasets, and the window size is set as $\{3 \times 3,\ 5 \times 5,\ 7 \times 7,\ 11 \times 11\ 15 \times 15,\ 19 \times 19,\ 25 \times 25,\ 31 \times 31\}$ for Indian Pines and Salinas datasets and $\{3 \times 3,\ 5 \times 5,\ 7 \times 7,\ 9 \times 9, 11 \times 11\ \ 15 \times 15,\ \ 15 \times 15,\ \ 17 \times 17\}$ for Pavia University dataset. Fig.3 shows the overall accuracy with the variations of the number of the training samples and the spatial window size. From Fig.3 it can be observed that, the proposed method achieves satisfactory and stable classification accuracy even though there are only a few tensor training samples. Jointly considering the computational complex and classification performance, we set the number of the training samples to 5 for each class of all the experimental datasets.

Meanwhile, with the increase of the window size, the overall accuracy (OA) increases first and then gradually decreases. This is because when the window size is not large, more pixels belonging to the same class will be included in a tensor sample

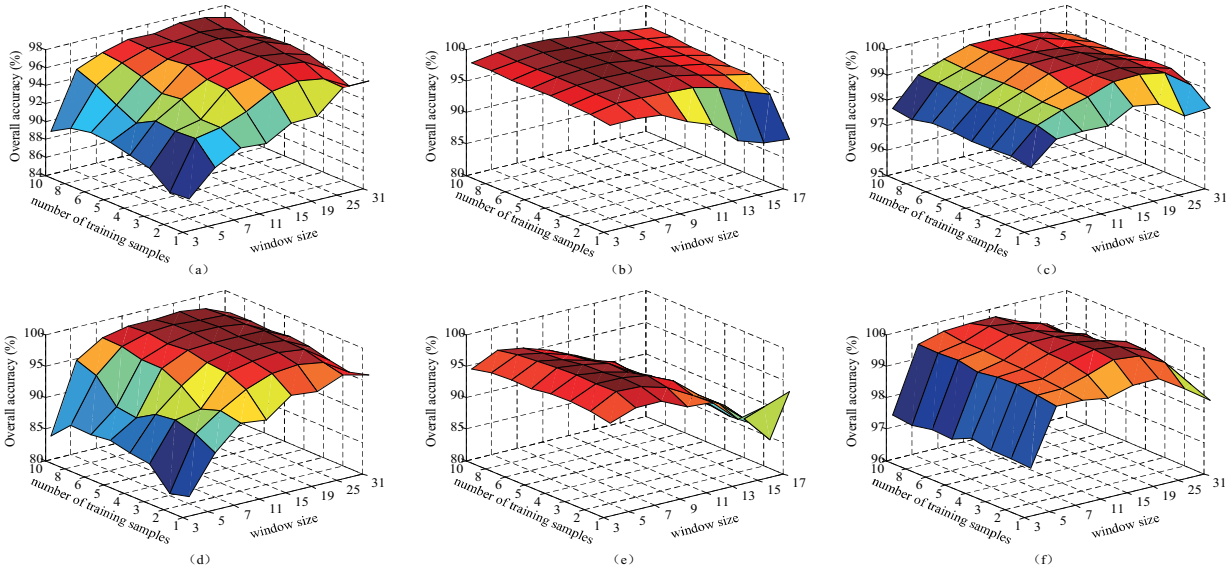


Fig. 3. Overall Accuracy with the variation of the window size and the number of training samples for each class. (a) and (d) are on the Indian Pines dataset with the SVM and 1NN classifiers respectively, (b) and (e) are on the Pavia University dataset with the SVM and 1NN classifiers respectively, (c) and (f) are on the Salinas dataset with the SVM and 1NN classifiers respectively.

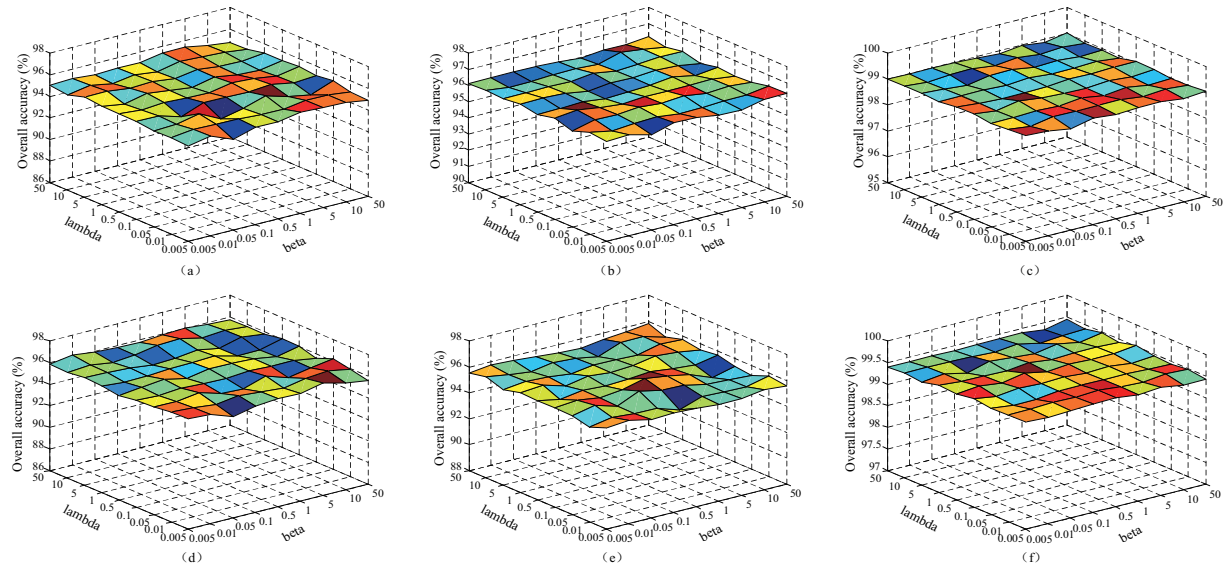


Fig. 4. Overall Accuracy with the parameter tuning of λ and β. (a) and (d) are on the Indian Pines dataset with the SVM and 1NN classifiers respectively, (b) and (e) are on the Pavia University dataset with the SVM and 1NN classifiers respectively, (c) and (f) are on the Salinas dataset with the SVM and 1NN classifiers respectively.

with the increasing of the window size which may provide more useful information. But when the window size is too large, pixels belonging to different classes may be included in the same tensor sample, the class and structural information may be destroyed and the OA decreases. From a general viewpoint, we set the window size as 11×11 for Indian Pines and Salinas datasets and 5×5 for Pavia University dataset.

D. Parameter tuning

In the proposed method, there are two regularization parameters (i.e. λ and β) in the objective function. These two parameters are used to balance the effect of the error and the regularization terms. In general, the parameter tuning may affect the resulting classification performance to some extent. In this section, we evaluate the system on the three experimental datasets to demonstrate the sensitivity of the proposed method over a wide range of parameter tuning. The parameter space is set as $\{0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50\}$. In addition, the number of the training samples for each class and the spatial size of the tensor samples are set as the optimal values as discussed above. Fig.4 shows the overall accuracy with the parameter tuning of λ and β on the experimental datasets.

From Fig.4, it can be observed that OA varies slightly with

TABLE I
OA, AA, CLASS-SPECIFIC ACCURACY (IN PERCENT), KAPPA COEFFICIENT AND THE CORRESPONDING VARIANCES OF DIFFERENT METHODS ON INDIAN PINES DATASET (REDUCED DIMENSIONALITY =30)

Class	Original		LDA		MMDA		LAPLRR		SLGDA		TDLA		TLPP		TLRR		T-LGMR	
	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN
1	73.75 ±10.68	67.92 ±12.89	68.54 ±10.18	63.75 ±10.40	81.67 ±5.97	68.75 ±10.10	67.92 ±10.27	64.17 ±6.81	75.42 ±14.76	67.92 ±7.31	89.27 ±6.40	87.60 ±10.45	92.71 ±3.26	86.88 ±10.04	79.17 ±8.07	77.50 ±5.78	**93.33 ±4.52**	**95.42 ±4.97**
2	83.77 ±1.17	63.33 ±1.42	80.59 ±2.20	56.02 ±7.21	81.13 ±0.96	66.43 ±3.11	81.52 ±2.27	69.24 ±2.11	83.15 ±1.32	69.78 ±1.28	90.71 ±2.44	92.29 ±3.50	91.45 ±2.80	89.21 ±4.01	90.47 ±1.40	86.57 ±1.49	**94.57 ±1.27**	**94.09 ±1.24**
3	78.93 ±2.39	62.37 ±1.72	66.13 ±1.70	49.95 ±5.07	74.91 ±2.04	63.63 ±1.18	70.40 ±3.76	66.00 ±2.43	69.73 ±1.95	63.84 ±2.44	86.33 ±3.66	90.42 ±4.19	86.27 ±4.20	82.31 ±8.60	83.28 ±1.51	74.77 ±2.29	**91.57 ±3.06**	**92.59 ±3.10**
4	70.10 ±7.43	46.10 ±7.81	62.57 ±7.62	40.14 ±8.27	62.86 ±9.09	45.52 ±6.75	82.86 ±4.68	44.76 ±4.03	75.52 ±8.22	46.38 ±7.37	86.07 ±6.83	**92.40 ±6.44**	87.29 ±6.22	86.67 ±5.53	77.33 ±5.44	76.95 ±4.28	**90.95 ±2.63**	87.62 ±5.76
5	94.05 ±1.63	91.59 ±2.01	91.25 ±2.24	83.98 ±6.87	93.29 ±2.01	90.20 ±3.24	94.00 ±1.27	90.20 ±3.26	92.17 ±3.39	87.96 ±1.13	93.80 ±1.96	94.00 ±3.77	**95.12 ±2.29**	93.94 ±3.70	93.69 ±2.28	93.87 ±3.07	94.59 ±3.35	**95.48 ±3.64**
6	96.67 ±1.11	94.67 ±1.29	95.33 ±1.62	90.10 ±5.86	95.92 ±2.11	95.30 ±1.95	95.92 ±1.18	95.15 ±1.23	96.64 ±1.52	95.48 ±1.89	97.48 ±1.30	96.83 ±3.88	97.53 ±1.13	96.01 ±1.30	96.88 ±2.37	93.63 ±2.47	**98.07 ±1.27**	**97.32 ±1.27**
7	78.26 ±10.2	83.48 ±7.78	68.26 ±20.91	60.00 ±18.08	77.39 ±19.3	77.39 ±13.54	69.57 ±13.40	73.91 ±14.42	75.65 ±11.75	81.74 ±6.45	73.26 ±33.85	**89.35 ±16.00**	80.43 ±16.53	86.96 ±4.76	83.48 ±3.64	76.52 ±10.01	**88.70 ±16.15**	77.39 ±16.67
8	98.18 ±0.89	98.05 ±0.83	97.20 ±1.73	96.82 ±1.72	97.36 ±1.14	97.82 ±0.61	98.45 ±1.05	98.09 ±1.29	99.05 ±0.44	97.59 ±1.17	98.22 ±1.13	98.37 ±1.77	**99.93 ±0.20**	99.57 ±0.44	99.50 ±0.10	**99.68 ±0.20**	99.64 ±0.20	99.64 ±0.20
9	72.22 ±17.10	57.78 ±9.30	70.56 ±20.96	56.67 ±27.57	74.44 ±17.83	58.89 ±11.52	63.33 ±29.03	42.22 ±14.49	65.56 ±15.91	54.44 ±19.4	48.61 ±32.53	**83.06 ±15.66**	70.56 ±19.88	78.33 ±14.37	61.11 ±25.76	66.67 ±23.9	**78.89 ±14.38**	82.22 ±17.3
10	78.99 ±2.35	74.28 ±2.39	69.43 ±2.82	54.59 ±6.06	73.27 ±5.01	74.79 ±2.05	71.55 ±4.25	78.87 ±2.63	71.96 ±2.43	77.47 ±0.50	81.46 ±4.75	90.22 ±4.30	84.10 ±4.05	84.80 ±6.61	82.46 ±3.27	73.87 ±3.08	**92.74 ±1.45**	**94.76 ±0.99**
11	84.07 ±1.43	77.03 ±2.13	81.31 ±1.54	65.38 ±2.59	81.75 ±0.81	76.79 ±1.69	86.66 ±1.04	79.69 ±1.46	87.32 ±0.86	78.51 ±2.28	92.12 ±1.82	93.22 ±3.24	90.72 ±2.64	92.34 ±2.72	88.54 ±1.70	88.71 ±0.73	**95.61 ±1.27**	**96.88 ±1.39**
12	84.42 ±2.62	58.80 ±4.19	74.51 ±3.47	61.05 ±8.80	81.01 ±3.12	60.36 ±2.03	87.83 ±4.33	58.26 ±4.34	86.70 ±6.43	59.24 ±1.93	86.08 ±5.24	87.98 ±5.81	90.94 ±3.49	87.84 ±5.74	88.01 ±4.10	76.09 ±3.27	**94.64 ±2.15**	**94.24 ±1.83**
13	99.05 ±0.69	98.00 ±1.36	98.74 ±0.93	94.63 ±7.67	98.74 ±1.52	98.11 ±1.15	97.16 ±2.62	98.11 ±0.71	99.26 ±0.47	98.63 ±0.47	94.24 ±4.01	94.08 ±6.66	97.11 ±2.79	95.63 ±2.35	99.26 ±0.29	**98.42 ±0.64**	**97.37 ±2.93**	96.00 ±4.20
14	94.64 ±1.43	93.06 ±1.39	95.64 ±2.13	93.33 ±2.36	92.78 ±1.71	93.14 ±1.55	96.82 ±0.70	92.63 ±1.33	96.49 ±0.90	92.90 ±1.36	97.46 ±1.47	97.53 ±1.33	98.23 ±0.72	97.92 ±1.15	95.45 ±1.72	97.25 ±0.47	**98.30 ±1.21**	**99.02 ±0.51**
15	61.05 ±6.07	42.40 ±3.74	67.98 ±6.33	50.88 ±8.84	60.94 ±4.57	47.89 ±3.68	57.72 ±4.43	42.75 ±2.07	59.30 ±4.74	40.12 ±3.39	88.86 ±5.89	92.46 ±5.38	89.74 ±5.39	87.95 ±4.17	79.53 ±3.27	79.47 ±5.33	**96.14 ±1.67**	**95.85 ±1.84**
16	90.82 ±3.37	93.18 ±3.16	82.00 ±6.13	69.41 ±15.37	89.88 ±2.44	86.59 ±4.75	88.24 ±7.98	86.35 ±5.74	89.65 ±4.43	85.41 ±3.39	**91.29 ±6.07**	90.94 ±6.24	89.53 ±8.16	**92.24 ±2.89**	88.47 ±4.19	85.65 ±1.93	90.35 ±4.19	91.76 ±2.20
OA	85.67 ±0.37	76.12 ±0.69	81.83 ±0.46	68.72 ±4.00	83.17 ±0.58	76.84 ±0.26	85.04 ±0.21	78.03 ±0.53	85.30 ±0.53	77.51 ±0.97	92.16 ±1.21	93.25 ±3.21	91.88 ±2.21	91.22 ±3.25	89.53 ±0.39	86.78 ±0.59	**95.31 ±0.65**	**95.76 ±0.54**
AA	83.71 ±0.49	75.13 ±1.14	79.38 ±2.08	67.92 ±5.40	82.33 ±1.62	75.10 ±2.29	81.87 ±2.17	73.78 ±1.25	82.72 ±1.82	74.84 ±1.50	87.20 ±3.58	91.92 ±4.58	90.10 ±3.61	89.91 ±3.27	86.66 ±1.87	84.10 ±1.67	**93.47 ±2.40**	**93.14 ±2.44**
Kappa	0.85 ±0	0.75 ±0.01	0.81 ±0	0.68 ±0.04	0.83 ±0.01	0.76 ±0	0.85 ±0	0.77 ±0.01	0.85 ±0.01	0.77 ±0.01	0.91 ±0.01	0.93 ±0.03	0.92 ±0.02	0.91 ±0.03	0.89 ±0	0.86 ±0.01	**0.95 ±0.01**	**0.96 ±0.01**

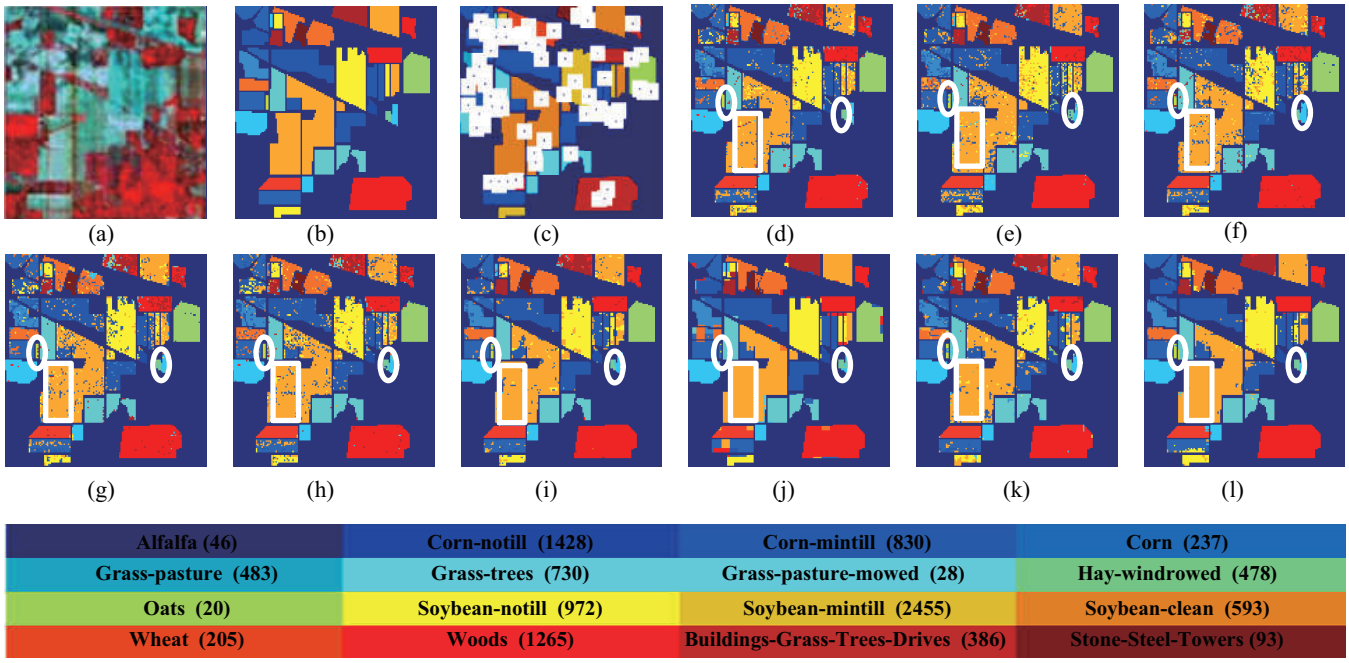


Fig. 5. Classification maps with SVM classifier of different methods on Indian Pines dataset. (a) Three-band color composite image. (b) Ground-truth. (c) Pixel and tensor training samples. (d) Original. (e) LDA. (f) MMDA. (g) LAPLRR. (h) SLGDA. (i) TDLA. (j) TLPP. (k)TLRR. (l) T-LGMR. Note that the number in the bracket on the bottom of the figures refers to the sample number of the corresponding class.

the tuning of λ and β. For example, in the entire parameter space, the largest difference between the highest and the lowest OA is only 1.49% which is obtained on the Pavia University dataset with the 1NN classifier (Fig.4 (d)) and the smallest one is only 0.23% which is obtained on the Salinas data set with the 1NN classifier (Fig.4 (f)). So we believe that the proposed method is robust against the change of the regularization parameters. Experimentally, we set $\lambda = 0.5$ and $\beta = 0.1$ for all the experiments.

TABLE II
OA, AA, CLASS-SPECIFIC ACCURACY (IN PERCENT), KAPPA COEFFICIENT AND THE CORRESPONDING VARIANCES OF DIFFERENT METHODS ON PAVIA UNIVERSITY DATASET (REDUCED DIMENSIONALITY =30)

Class	Original		LDA		MMDA		LAPLRR		SLGDA		TDLA		TLPP		TLRR		T-LGMR	
	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN
1	89.65 ±0.69	75.92 ±0.99	91.75 ±0.57	83.80 ±1.02	88.44 ±0.17	78.59 ±0.74	90.33 ±1.31	76.94 ±1.04	89.88 ±1.05	78.09 ±0.73	94.83 ±1.31	90.20 ±4.62	**96.73 ±0.70**	86.16 ±3.96	94.08 ±0.83	70.67 ±2.85	96.41 ±0.83	**92.38 ±1.23**
2	95.54 ±0.63	95.27 ±0.15	94.90 ±0.35	93.20 ±0.42	93.63 ±0.74	93.61 ±0.73	94.39 ±0.25	95.30 ±0.13	94.02 ±0.23	94.19 ±0.48	97.94 ±0.82	99.09 ±0.46	98.71 ±0.31	99.17 ±0.37	97.75 ±0.53	97.97 ±0.40	**99.18 ±0.16**	**99.73 ±0.09**
3	70.92 ±2.02	60.30 ±1.60	65.24 ±1.69	65.33 ±1.34	69.25 ±2.26	60.48 ±2.65	70.50 ±2.48	61.82 ±1.14	70.20 ±1.50	60.79 ±1.40	80.06 ±1.34	85.76 ±2.10	83.59 ±2.87	83.69 ±2.24	76.82 ±2.95	68.93 ±1.93	**85.08 ±1.52**	**89.09 ±1.32**
4	92.74 ±1.17	83.29 ±1.40	87.39 ±1.39	84.19 ±0.77	90.56 ±0.84	83.68 ±1.01	92.31 ±1.18	84.45 ±1.12	92.16 ±0.73	84.13 ±0.72	94.83 ±2.06	90.34 ±2.72	96.21 ±1.06	92.08 ±2.67	94.76 ±0.85	91.57 ±1.13	**96.93 ±0.98**	**93.02 ±0.91**
5	99.31 ±0.37	99.27 ±0.31	99.81 ±0.13	99.73 ±0.11	99.52 ±0.17	99.31 ±0.28	99.50 ±0.14	99.61 ±0.12	99.53 ±0.07	99.58 ±0.09	99.89 ±0.13	99.73 ±0.35	99.96 ±0.12	99.77 ±0.46	99.92 ±0.06	**99.79 ±0.04**	**99.97 ±0.07**	99.57 ±0.4
6	78.77 ±2.08	56.48 ±1.03	76.81 ±1.00	67.63 ±0.74	75.36 ±1.81	55.88 ±1.80	77.00 ±1.93	57.80 ±1.00	76.51 ±0.93	57.66 ±0.63	81.03 ±2.52	93.92 ±1.53	91.71 ±1.78	90.30 ±1.99	82.03 ±1.35	71.79 ±1.13	**93.49 ±0.69**	**97.69 ±0.85**
7	83.31 ±2.09	77.77 ±1.34	61.10 ±2.69	73.73 ±2.22	80.18 ±2.72	76.75 ±1.92	81.64 ±1.41	78.03 ±0.47	81.39 ±1.17	77.89 ±1.42	94.72 ±2.52	92.16 ±2.33	91.92 ±2.75	92.30 ±1.63	89.36 ±1.50	85.16 ±1.90	**94.82 ±0.48**	**96.25 ±1.07**
8	78.64 ±0.73	76.82 ±1.66	84.34 ±0.80	72.22 ±1.88	73.99 ±1.59	74.98 ±1.79	76.56 ±0.88	76.53 ±0.92	75.66 ±2.03	75.74 ±0.56	87.37 ±1.09	86.87 ±2.75	**90.35 ±1.32**	88.96 ±2.10	87.14 ±1.83	81.01 ±2.39	89.42 ±2.29	**89.7± 1.35**
9	**98.94 ±0.35**	92.03 ±1.14	98.94 ±0.25	96.29 ±0.91	95.62 ±0.70	92.22 ±1.69	98.66 ±0.42	91.48 ±2.70	98.70 ±0.17	**93.20 ±2.39**	97.22 ±1.39	84.59 ±5.57	97.26 ±1.26	69.65 ±3.62	91.09 ±2.24	61.86 ±4.89	96.45 ±1.85	76.22 ±4.09
OA	89.54 ±0.25	82.93 ±0.37	88.50 ±0.14	84.48 ±0.34	87.31 ±0.59	82.42 ±0.49	88.66 ±0.17	83.41 ±0.24	88.27 ±0.42	82.98 ±0.32	93.36 ±0.89	94.14 ±1.02	94.19 ±0.74	92.99 ±0.85	92.78 ±0.95	86.03 ±1.21	**96.17 ±0.22**	**95.74 ±0.43**
AA	87.53 ±0.57	79.68 ±0.49	84.48 ±0.24	81.79 ±0.40	85.17 ±0.51	79.5 ±0.39	86.77 ±0.31	80.22 ±0.28	86.45 ±0.42	80.14 ±0.54	91.99 ±1.21	91.41 ±1.92	94.05 ±1.00	89.12 ±1.56	90.33 ±1.14	80.97 ±1.30	**94.64 ±0.24**	**92.63 ±0.73**
Kappa	0.88 ±0	0.80 ±0	0.86 ±0	0.82 ±0	0.85 ±0.01	0.79 ±0.01	0.87 ±0	0.80 ±0	0.86 ±0	0.80 ±0	0.92 ±0.01	0.93 ±0.03	0.95 ±0.01	0.92 ±0.02	0.91 ±0.01	0.83 ±0.03	**0.95 ±0**	**0.95 ±0.01**

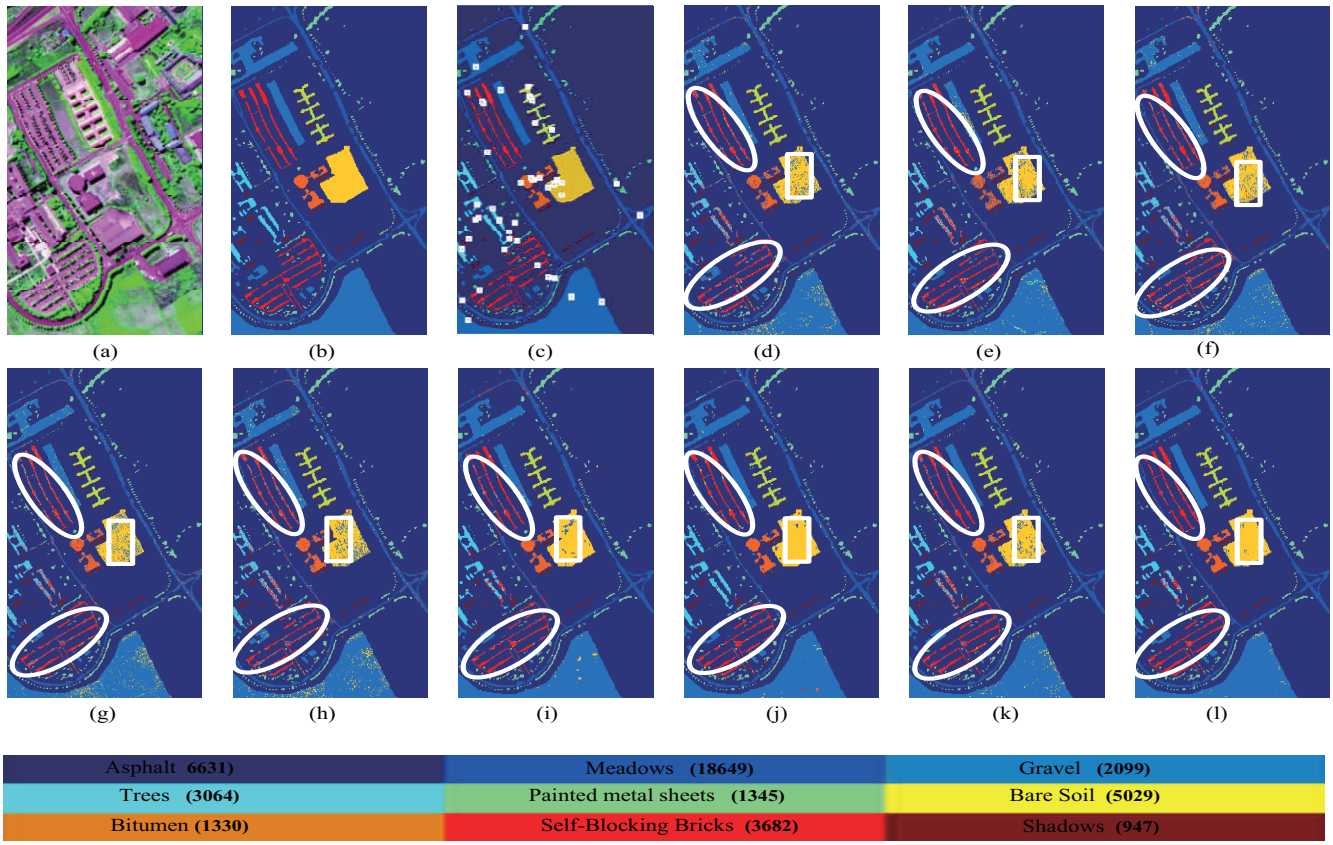


Fig. 6. Classification maps with SVM classifier of different methods on the Pavia University dataset. (a) Three-band color composite image. (b) Ground-truth. (c) Pixel and tensor training samples. (d) Original. (e) LDA. (f) MMDA. (g) LAPLRR. (h) SLGDA. (i) TDLA. (j) TLPP. (k)TLRR. (l) T-LGMR.

E. Classification results

In this section, experiments are conducted on the three real hyperspectral datasets to demonstrate the performance of the proposed dimensionality reduction method for hyperspectral classification. As discussed above, seven dimensionality reduction methods are used for comparison and the original spectral bands are used as the baseline. To evaluate the classification performance, the training samples are selected randomly in all the comparison methods and all the experiments are repeated 20 times. We report the means and the variances of the classification results on the three real hyperspectral datasets. The classification results (including OA, AA, Class-specific Accuracy and Kappa coefficient) and classification maps are shown in Tables I, II, and III and Figs.5, 6, and 7.

It can be seen that the proposed method achieves much better classification performance in terms of OA, AA and

TABLE III
OA, AA, CLASS-SPECIFIC ACCURACY (IN PERCENT), KAPPA COEFFICIENT AND THE CORRESPONDING VARIANCES OF DIFFERENT METHODS ON SALINAS DATASET (REDUCED DIMENSIONALITY =30)

Class	Original		LDA		MMDA		LAPLRR		SLGDA		TDLA		TLPP		TLRR		T-LGMR	
	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN	SVM	1NN
1	99.57 ±0.30	99.36 ±0.38	99.81 ±0.15	99.78 ±0.12	99.37 ±0.27	99.14 ±0.08	99.68 ±0.28	99.38 ±0.32	99.40 ±0.25	99.25 ±0.29	99.57 ±0.39	99.74 ±0.22	99.84 ±0.30	99.72 ±0.38	99.93 ±0.15	99.88 ±0.21	**99.98 ±0.05**	**99.93 ±0.09**
2	99.67 ±0.26	99.18 ±0.15	99.96 ±0.04	99.94 ±0.05	99.74 ±0.16	99.34 ±0.23	99.61 ±0.20	99.27 ±0.20	99.79 ±0.20	99.39 ±0.23	99.89 ±0.15	99.97 ±0.04	99.99 ±0.04	99.98 ±0.04	100 ±0	99.99 ±0.02	**100 ±0**	**100 ±0**
3	99.71 ±0.26	98.14 ±0.41	99.76 ±0.12	99.66 ±0.13	99.46 ±0.09	97.96 ±0.55	99.26 ±0.32	97.72 ±0.70	99.36 ±0.59	97.71 ±0.51	99.76 ±0.33	99.81 ±0.31	99.87 ±0.15	99.22 ±0.89	99.99 ±0.03	99.89 ±0.11	**99.99 ±0.03**	**100 ±0**
4	99.25 ±0.60	99.49 ±0.27	99.06 ±0.38	99.30 ±0.22	98.88 ±1.41	99.43 ±0.25	**99.51 ±0.17**	**99.57 ±0.17**	99.36 ±0.11	99.51 ±0.13	98.50 ±0.63	98.50 ±0.76	94.82 ±3.80	90.16 ±9.85	98.87 ±0.37	98.68 ±0.43	98.28 ±1.19	98.92 ±0.42
5	99.18 ±0.45	97.16 ±0.75	99.00 ±0.28	98.87 ±0.20	98.35 ±0.40	97.10 ±0.46	98.73 ±0.38	96.69 ±0.35	98.79 ±0.42	96.90 ±0.38	**99.02 ±0.43**	99.23 ±0.60	96.87 ±2.09	97.02 ±2.10	99.37 ±0.42	**99.25 ±0.18**	98.76 ±0.64	98.84 ±0.44
6	99.87 ±0.16	99.84 ±0.19	99.96 ±0.02	99.93 ±0.04	99.82 ±0.08	99.67 ±0.16	99.81 ±0.10	99.78 ±0.08	99.78 ±0.16	99.76 ±0.16	**99.99 ±0.01**	**100 ±0.01**	99.97 ±0.06	99.98 ±0.04	99.99 ±0.01	99.99 ±0.02	99.65 ±0.12	99.58 ±0.19
7	99.80 ±0.13	99.59 ±0.06	99.91 ±0.06	99.88 ±0.06	99.65 ±0.13	99.50 ±0.18	99.58 ±0.06	99.45 ±0.05	99.63 ±0.24	99.48 ±0.19	99.71 ±0.21	99.84 ±0.14	**99.99 ±0.02**	99.95 ±0.05	99.94 ±0.09	99.95 ±0.06	99.94 ±0.06	**99.98 ±0.03**
8	84.26 ±0.90	78.34 ±0.95	88.80 ±0.71	76.99 ±0.65	88.70 ±0.92	77.19 ±1.39	88.08 ±0.63	77.37 ±0.66	88.20 ±0.32	77.16 ±0.55	91.38 ±1.60	98.39 ±0.78	92.79 ±3.29	87.51 ±9.12	95.27 ±0.76	94.58 ±1.39	**98.39 ±0.66**	**99.14 ±0.30**
9	99.84 ±0.22	99.24 ±0.27	99.94 ±0.05	99.73 ±0.25	99.07 ±0.06	98.70 ±0.42	99.58 ±0.42	99.18 ±0.24	99.62 ±0.26	99.00 ±0.19	99.99 ±0.02	99.97 ±0.05	99.84 ±0.21	99.93 ±0.06	99.92 ±0.09	99.98 ±0.02	**99.99 ±0.02**	**99.99 ±0.02**
10	96.89 ±0.77	94.43 ±0.69	97.60 ±0.39	97.41 ±0.59	96.70 ±0.55	94.31 ±0.85	96.81 ±0.52	94.77 ±0.29	97.23 ±0.27	95.13 ±0.54	99.46 ±0.32	98.69 ±0.20	99.16 ±0.47	98.38 ±1.12	98.49 ±0.47	98.70 ±0.22	**99.44 ±0.22**	**99.52 ±0.13**
11	99.23 ±0.26	98.88 ±0.40	98.82 ±0.61	97.60 ±0.53	98.63 ±0.76	98.21 ±1.28	99.23 ±0.43	99.25 ±0.20	98.69 ±1.07	98.63 ±0.93	99.73 ±0.48	98.66 ±0.32	99.50 ±0.63	98.73 ±1.94	**99.83 ±0.17**	**99.83 ±0.12**	99.19 ±0.55	99.38 ±0.88
12	99.87 ±0.22	99.75 ±0.21	99.57 ±0.22	99.56 ±0.23	99.91 ±0.10	99.86 ±0.07	99.88 ±0.18	99.85 ±0.18	99.77 ±0.17	99.82 ±0.13	99.93 ±0.09	99.89 ±0.24	99.64 ±0.55	98.25 ±2.93	**99.94 ±0.06**	**99.82 ±0.13**	99.61 ±0.16	99.54 ±0.14
13	99.22 ±0.82	98.03 ±0.50	99.02 ±0.45	98.62 ±0.52	99.00 ±0.52	97.86 ±0.50	99.39 ±0.51	98.03 ±0.68	99.30 ±0.38	98.01 ±0.59	99.84 ±0.27	**99.90 ±0.19**	99.39 ±0.71	97.28 ±2.87	99.83 ±0.25	**99.68 ±0.32**	97.67 ±1.79	97.48 ±1.65
14	98.07 ±0.66	94.77 ±0.68	97.62 ±0.52	96.57 ±0.62	98.80 ±0.37	96.12 ±0.64	98.38 ±0.44	95.60 ±1.28	98.15 ±1.13	94.54 ±1.31	**99.44 ±0.80**	**98.60 ±0.65**	98.97 ±0.73	98.55 ±1.18	98.40 ±0.55	98.57 ±0.57	97.78 ±0.63	98.15 ±1.15
15	75.59 ±1.19	68.38 ±0.78	65.95 ±1.41	68.53 ±0.99	76.88 ±1.72	67.77 ±0.90	78.35 ±0.77	67.56 ±1.34	77.68 ±1.03	67.28 ±1.40	85.11 ±2.34	97.84 ±1.15	85.90 ±6.48	84.89 ±8.86	88.64 ±0.26	91.31 ±1.90	**96.26 ±0.71**	**98.92 ±0.41**
16	98.51 ±0.28	98.27 ±0.35	99.27 ±0.39	99.30 ±0.39	98.86 ±0.21	98.51 ±0.22	98.71 ±0.68	98.34 ±0.43	98.75 ±0.36	98.30 ±0.62	99.68 ±0.41	99.70 ±0.46	99.90 ±0.16	99.72 ±0.34	99.47 ±0.49	99.45 ±0.47	**100 ±0**	**99.94 ±0.06**
OA	92.98 ±0.23	90.25 ±0.24	92.72 ±0.08	90.51 ±0.22	93.99 ±0.14	89.86 ±0.30	94.08 ±0.15	89.94 ±0.29	94.03 ±0.13	89.84 ±0.19	96.00 ±0.41	98.02 ±0.37	96.18 ±1.79	94.65 ±3.69	97.27 ±0.20	97.48 ±0.54	**98.88 ±0.14**	**99.42 ±0.08**
AA	96.78 ±0.11	95.18 ±0.14	96.5 ±0.08	95.73 ±0.11	97.03 ±0.19	95.04 ±0.17	97.16 ±0.12	95.11 ±0.20	97.09 ±0.14	94.99 ±0.07	98.19 ±0.18	99.48 ±0.23	97.90 ±1.08	96.83 ±2.47	98.62 ±0.09	98.72 ±0.21	**99.06 ±0.11**	**99.33 ±0.21**
Kappa	0.93 ±0	0.90 ±0	0.93 ±0	0.90 ±0	0.94 ±0	0.90 ±0	0.94 ±0	0.9 ±0	0.94 ±0	0.90 ±0	0.96 ±0	0.99 ±0	0.96 ±0.02	0.95 ±0.04	0.97 ±0	0.97 ±0.01	**0.99 ±0**	**0.99 ±0**

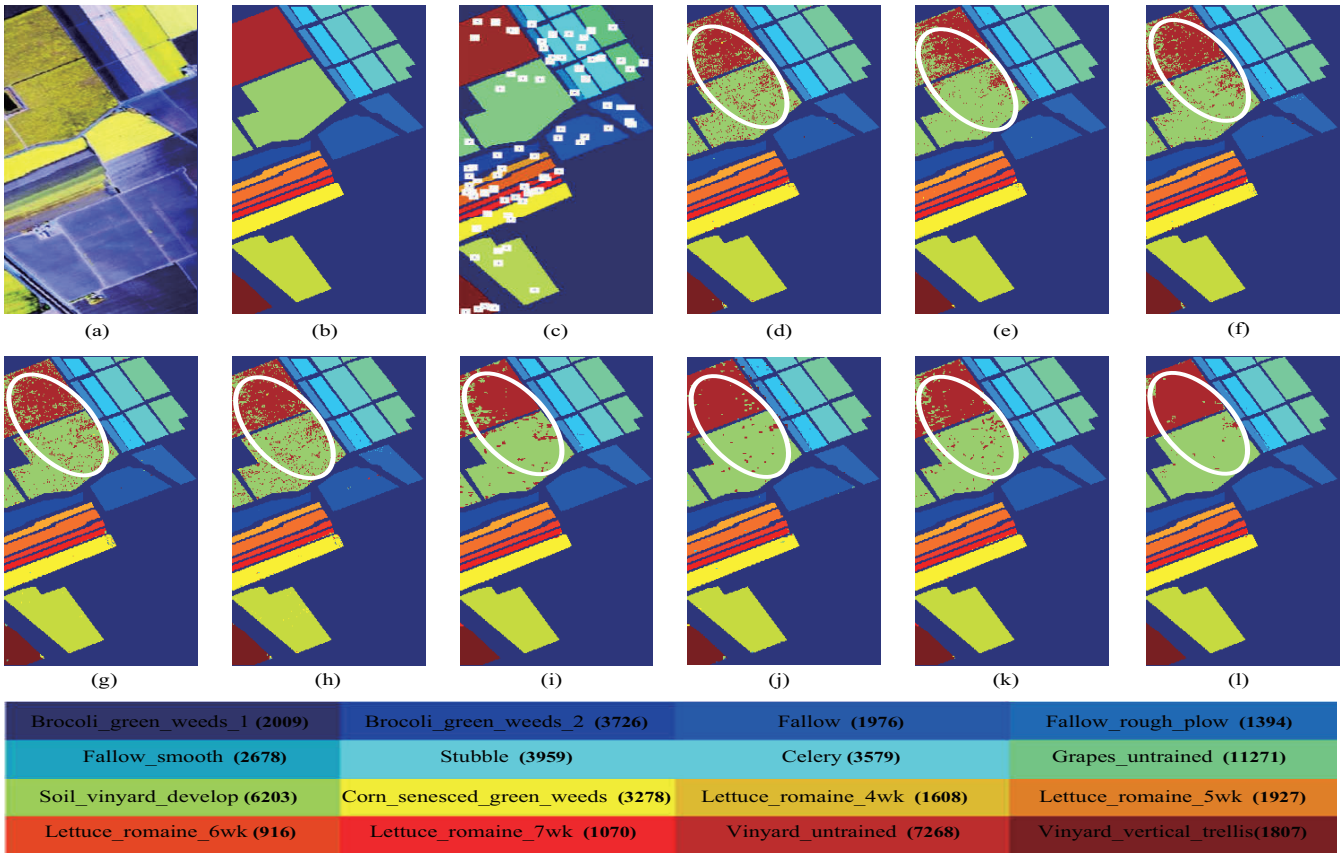


Fig. 7. Classification maps with SVM classifier of different methods on Salinas dataset. (a) Three-band color composite image. (b) Ground-truth. (c) Pixel and tensor training samples. (d) Original. (e) LDA. (f) MMDA. (g) LAPLRR. (h) SLGDA. (i) TDLA. (j) TLPP. (k)TLRR. (l) T-LGMR.

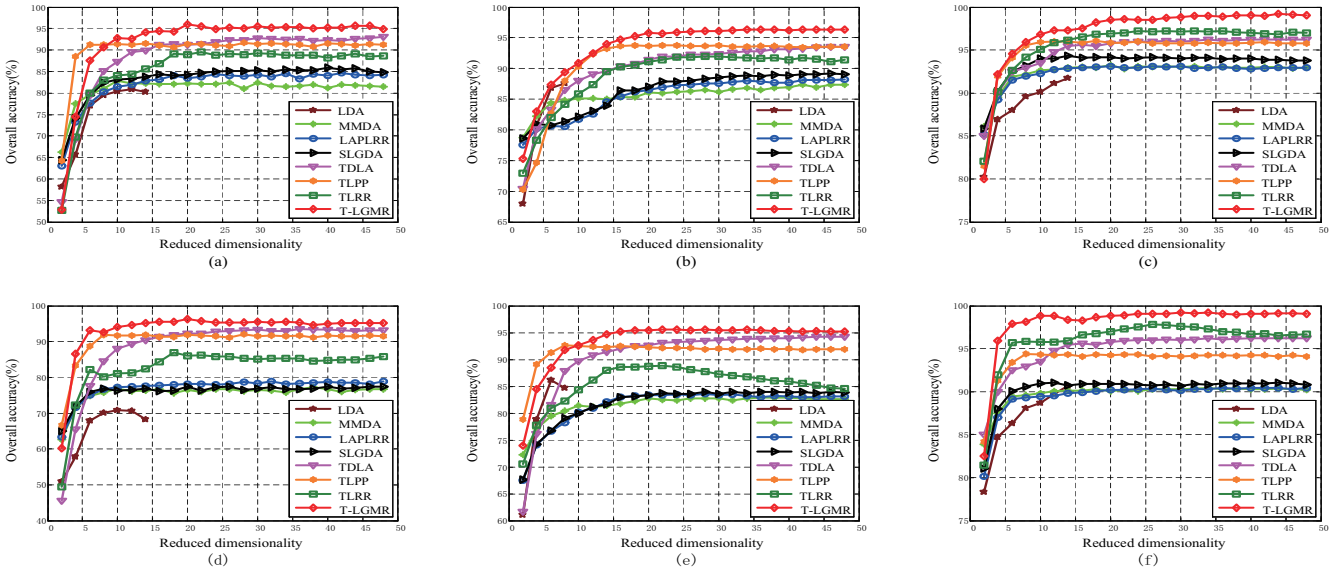


Fig. 8. Overall accuracy versus reduced dimensionality. (a) and (d) are on the Indian Pines dataset with the SVM and 1NN classifiers respectively, (b) and (e) are on the Pavia University dataset with SVM and 1NN classifier respectively, (c) and (f) are on the Salinas dataset with the SVM and 1NN classifiers respectively.

TABLE IV
COMPUTATIONAL COMPLEXITY IN TERMS OF RUNNING TIME (IN SECONDS) ON THREE EXPERIMENTAL DATASETS

	Original	LDA	MMDA	LAPLRR	SLGDA	TDLA	TLPP	TLRR	T-LGMR
Indian Pines	1.90	1.02	0.72	2.72	1.78	45.57	44.26	36.84	41.41
Pavia University	10.26	7.52	5.53	5.69	5.42	64.69	51.23	12.69	17.17
Salinas	18.97	13.46	6.61	9.14	7.65	105.32	85.32	32.74	39.24

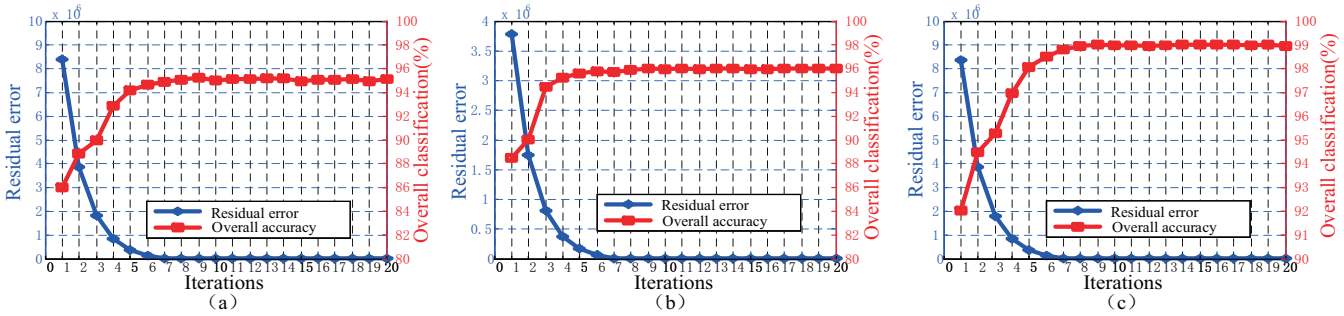


Fig. 9. The variation of overall accuracy versus the residual error between two iterations. (a) Indian Pines. (b) Pavia University. (c) Salinas.

Kappa than all the other state-of-the-art methods. For example, in terms of OA, the proposed method is approximately 2%, 2% and 1% better than that of the second best method. This demonstrates that the proposed method is an effective discriminative dimensionality reduction method. In addition, the results show that the tensor-based methods achieve better performance than the pixel-based methods.

For the Indian pines image, it can be observed from Fig.5 that, the proposed method has significantly region uniformity (marked by the white rectangle) compared to the other methods (see Fig.5 (d)-(j)). In addition, for small sample classes, such as classes 9 and 7, there are only 20 and 26 samples, respectively. Such small samples pose a challenge to the classification task. By fully exploiting both spatial-spectral and discriminative information, the proposed method yields promising results(see Table I), which is in accordance with the results shown in Fig. 5 (d)-(j) (marked by the white ellipse).

For the Pavia University image, good region uniformity is also obtained by T-LGMR (marked by the white rectangle in Fig.6 (d)-(j)). Moreover, for a ribbon distribution, such as class 8 (marked by the white ellipse in Fig.6 (d)-(j)), the proposed method can also achieve satisfactory performance, which is in accordance with the results shown in Table II.

For the Salinas image, land cover in this scene has good

region homogeneity (see Fig.7(b)) and all the methods can achieve promising results. It should be noted that, for classes 2 and 16 (marked by the white rectangle), the proposed method delivers 100% classification accuracy with the SVM classifier. Classes 8 and 15 (marked by the white ellipse) are difficult to be distinguished in this scene. For these two classes, the proposed method performs much better than the other methods and the classification accuracy is significantly improved (see Fig.7 (d)-(j)).

F. Sensibility analysis of reduced dimensionality

We also investigate the effect of reduced dimensionality on the performance of our method. Fig.8 shows the Overall Accuracy results with the SVM and 1NN classifiers on all the experimental datasets when the dimensionality ranges from 3 to 49 with the step length of 2. From Fig.8, we can see that the Overall Accuracy is improved with the increasing of the dimensionality and tend to be stable after the dimensionality is larger than 30. As a result, we set the dimensionality to 30. Fig.8 also illustrates the advantage of the proposed method over the other comparison methods when the dimensionality is low, which further demonstrates the ability of the proposed method in dimensionality reduction.

G. Algorithm analysis

1) Computational complexity analysis: In this section, we analyze the computational complexity (in terms of running time) of T-LGMR. All the experiments are carried out using Matlab R2014a on a PC with Intel Core i5-5490 CPU and 8 GB RAM. The procedures of all the comparison methods involved in this paper contain two main processing steps: dimensionality reduction and classification. For all the comparison methods, the dimensionality reduction techniques are unique while the classification methods are the same. In order to evaluate the computational efficiency, we record the running time consumed by each comparison method. Table IV shows the running time of each method with the SVM classifier on the three experimental datasets. It is shown that the running time of LDA and MMDA whose solutions do not need the iterative strategy are shorter than that of the others. The running time of LAPLRR and SLGDA are slightly longer than that of LDA and MMDA due to the iterative strategy in the matrix form which is employed in these two methods which costs more time to converge to the optimal solution. Tensor-based methods, i.e., TDLA, TLPP, TLRR and T-LGMR need longer running time as the tensor-based methods need more time to calculate the solution for each mode. Furthermore, T-LGMR consumes slightly longer time than TLRR and much shorter time than TDLA and TLPP. Compared with the excellent classification performance, the running time of T-LGMR is acceptable.

2)Convergence analysis: To illustrate the convergence of the proposed method, the residual errors between two iterations and the corresponding overall classification accuracy on the three hyperspectral datasets are presented. It can be seen from Fig.9 that the residual errors between the two iterations can quickly converge to zero after 10 iterations. In addition, by setting the iteration times as 20 for all the three datasets which can guarantee the proposed method to converge to an optimal solution, we record the corresponding overall classification accuracy with the SVM classifier in each iteration. It can be seen from Fig.9 that the overall accuracy increases and then reaches the optimal result with the decrease of the residual error. All these experimental results suggest the proposed method can converge in a few iterations.

V. CONCLUSION

In this paper, we proposed a tensor-based low rank graph with multi-manifold regularization for dimensionality reduction of hyperspectral images. By jointly utilizing the low rank constraint and multi-manifold information, an informative and discriminative graph is constructed for dimensionality reduction of hyperspectral images. In addition, different from the available vector-based graphs, the proposed tensor-based graph can fully exploit the information of all the training samples along two spatial dimensions. Experimental results on several real hyperspectral datasets prove its efficiency and superiority to several state-of-the-art techniques. In our future work, more intrinsic information within the tensor data, e.g., the sparsity of hyperspectral image will be jointly considered to enhance the classification performance.

ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China (nos. 61772400, 61501353, 61772399, 91438201, 61573267). Dr. H. Zhou is supported by UK EPSRC under Grants EP/N508664/1, EP/R007187/1 and EP/N011074/1, and Royal Society-Newton Advanced Fellowship under Grant NA160342.

REFERENCES

[1] X. Zhang, Z. Gao, L. Jiao, and H. Zhou, "Multifeature hyperspectral image classification with local and nonlocal spatial information via markov random field in semantic space," *IEEE Transactions on Geoscience & Remote Sensing*, vol. PP, no. 99, pp. 1–16, 2017.

[2] X. Zhang, Y. Liang, C. Li, H. Ning, L. Jiao, and H. Zhou, "Recursive autoencoders-based unsupervised feature learning for hyperspectral image classification," *IEEE Geoscience & Remote Sensing Letters*, vol. 14, no. 11, pp. 1928–1932, 2017.

[3] H. Kwon and N. M. Nasrabadi, "Kernel matched subspace detectors for hyperspectral target detection," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 28, no. 2, pp. 178–194, 2005.

[4] D. Manolakis, D. Marden, and G. A. Shaw, "Hyperspectral image processing for automatic target detection applications," 2003.

[5] A. Banerjee, P. Burlina, and C. Diehl, "A support vector method for anomaly detection in hyperspectral imagery," *IEEE Transactions on Geoscience Remote Sensing*, vol. 44, no. 8, pp. 2282–2291, 2006.

[6] O. Duran and M. Petrou, "A time-efficient method for anomaly detection in hyperspectral images," *IEEE Transactions on Geoscience Remote Sensing*, vol. 45, no. 12, pp. 3894–3904, 2007.

[7] X. Zhang, C. Li, J. Zhang, Q. Chen, J. Feng, L. Jiao, and H. Zhou, "Hyperspectral unmixing via low-rank representation with space consistency constraint and spectral library pruning," *Remote Sensing*, vol. 10, no. 2, p. 339, 2018.

[8] Q. Du, E. Michaelsen, P. Du, and L. Bruzzone, "Foreword to the special issue on pattern recognition in remote sensing," *Selected Topics in Applied Earth Observations Remote Sensing IEEE Journal of*, vol. 7, no. 12, pp. 4615–4619, 2014.

[9] H. Huang, J. Li, and J. Liu, "Enhanced semi-supervised local fisher discriminant analysis for face recognition," *Future Generation Computer Systems*, vol. 28, no. 1, pp. 244–253, 2012.

[10] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics Intelligent Laboratory Systems*, vol. 2, no. 1, pp. 37–52, 1987.

[11] A. M. Martinez and A. C. Kak, "Pca versus lda," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 23, no. 2, pp. 228–233, 2002.

[12] W. Li, J. Liu, and Q. Du, "Sparse and low-rank graph for discriminant analysis of hyperspectral imagery," *IEEE Transactions on Geoscience Remote Sensing*, vol. 54, no. 7, pp. 4094–4105, 2016.

[13] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation." *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2012.

[14] H. Zhang, W. He, L. Zhang, H. Shen, and Q. Yuan, "Hyperspectral image restoration using low-rank matrix recovery," *IEEE Transactions on Geoscience Remote Sensing*, vol. 52, no. 8, pp. 4729–4743, 2014.

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

[16] Y. Q. Zhao and J. Yang, "Hyperspectral image denoising via sparse representation and low-rank constraint," *IEEE Transactions on Geoscience Remote Sensing*, vol. 53, no. 1, pp. 296–308, 2014.

[17] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 2790–2797.

[18] H. S. Seung and D. D. Lee, "The manifold ways of perception," *Science*, vol. 290, no. 5500, pp. 2268–9, 2000.

[19] B. Du, L. Zhang, T. Chen, and K. Wu, "A discriminative manifold learning based dimension reduction method for hyperspectral classification," *International Journal of Fuzzy Systems*, vol. 14, no. 2, pp. 272–277, 2012.

[20] L. Ma, M. M. Crawford, and J. Tian, "Local manifold learning-based k -nearest-neighbor for hyperspectral image classification," *IEEE Transactions on Geoscience Remote Sensing*, vol. 48, no. 11, pp. 4099–4109, 2010.

[21] C. M. Bachmann, T. L. Ainsworth, and R. A. Fusina, "Exploiting manifold geometry in hyperspectral imagery," *IEEE Transactions on Geoscience Remote Sensing*, vol. 43, no. 3, pp. 441–454, 2005.

[22] J. J. Egozcue, V. Pawlowsky-Glahn, G. Mateu-Figueras, and C. Barcel-Vidal, "Isometric logratio transformations for compositional data analysis," *Mathematical Geology*, vol. 35, no. 3, pp. 279–300, 2003.

[23] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–6, 2000.

[24] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," vol. 15, no. 6, pp. 1373—1396, 2003.

[25] X. He and P. Niyogi, "Locality preserving projections," *Advances in Neural Information Processing Systems*, vol. 16, no. 1, pp. 186–197, 2003.

[26] S. Yan, D. Xu, B. Zhang, H. J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extension: A general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2006.

[27] W. Li and Q. Du, "Laplacian regularized collaborative graph for discriminant analysis of hyperspectral imagery," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 54, no. 12, pp. 7066–7076, 2016.

[28] F. Feng, W. Li, Q. Du, and B. Zhang, "Dimensionality reduction of hyperspectral image with graph-based discriminant analysis considering spectral similarity," *Remote Sensing*, vol. 9, no. 4, p. 323, 2017.

[29] W. Li, J. Liu, and Q. Du, "Sparse and low-rank graph for discriminant analysis of hyperspectral imagery," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 54, no. 7, pp. 4094–4105, 2016.

[30] L. Pan, H. C. Li, W. Li, X. D. Chen, G. N. Wu, and Q. Du, "Discriminant analysis of hyperspectral imagery using fast kernel sparse and low-rank graph," *IEEE Transactions on Geoscience & Remote Sensing*, vol. PP, no. 99, pp. 1–14, 2017.

[31] L. Zhang, L. Zhang, D. Tao, and X. Huang, "Tensor discriminative locality alignment for hyperspectral image spectralcspatial feature extraction," *IEEE Transactions on Geoscience Remote Sensing*, vol. 51, no. 1, pp. 242–256, 2012.

[32] Q. Zhang, H. Wang, R. J. Plemmons, and V. P. Pauca, "Tensor methods for hyperspectral data analysis: a space object material identification study." *Journal of the Optical Society of America A Optics Image Science Vision*, vol. 25, no. 12, pp. 3001–12, 2008.

[33] S. Bourennane, C. Fossati, and A. Cailly, "Improvement of classification for hyperspectral images based on tensor modeling," *IEEE Geoscience Remote Sensing Letters*, vol. 7, no. 4, pp. 801–805, 2010.

[34] J. An, X. Zhang, and L. C. Jiao, "Dimensionality reduction based on group-based tensor model for hyperspectral image classification," *IEEE Geoscience & Remote Sensing Letters*, vol. 13, no. 10, pp. 1497–1501, 2016.

[35] Z. Zhong, B. Fan, J. Duan, L. Wang, K. Ding, S. Xiang, and C. Pan, "Discriminant tensor spectralcspatial feature extraction for hyperspectral image classification," *IEEE Geoscience & Remote Sensing Letters*, vol. 12, no. 5, pp. 1028–1032, 2017.

[36] Y. Gao, X. Wang, Y. Cheng, and Z. J. Wang, "Dimensionality reduction for hyperspectral data based on class-aware tensor neighborhood graph and patch alignment." *IEEE Transactions on Neural Networks & Learning Systems*, vol. 26, no. 8, pp. 1582–1593, 2015.

[37] Y. J. Deng, H. C. Li, L. Pan, and W. J. Emery, "Tensor locality preserving projection for hyperspectral image classification," in *IGARSS 2017 - 2017 IEEE International Geoscience and Remote Sensing Symposium*, 2017, pp. 771–774.

[38] H. Fan, Y. Chen, Y. Guo, H. Zhang, and G. Kuang, "Hyperspectral image restoration using low-rank tensor recovery," *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing*, vol. PP, no. 99, pp. 1–16, 2017.

[39] Y. Wei and Y. Zhou, "Stacked tensor subspace learning for hyperspectral image classification," in *International Joint Conference on Neural Networks*, 2016, pp. 1985–1992.

[40] W. Yang, C. Sun, and L. Zhang, *A multi-manifold discriminant analysis method for image feature extraction*. Elsevier Science Inc., 2011.

[41] H. Zhang, Z. Lin, and C. Zhang, "A counterexample for the validity of using nuclear norm as a convex surrogate of rank," *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, vol. 8189, pp. 226–241, 2013.

[42] J. Liu, S. Ji, and J. Ye, "Multi-task feature learning via efficient l2,1-norm minimization," *Computer Science*, pp. 339–348, 2012.

[43] C. Li, Y. Ma, J. Huang, X. Mei, and J. Ma, "Hyperspectral image denoising using the robust low-rank tensor recovery." *Journal of the Optical Society of America A Optics Image Science Vision*, vol. 32, no. 9, pp. 1604–12, 2015.

[44] M. Zhang, B. Du, L. Zhang, and X. Li, "A low-rank tensor decomposition based hyperspectral image compression algorithm," 2016.

[45] W. Dong, G. Li, G. Shi, X. Li, and Y. Ma, "Low-rank tensor approximation with laplacian scale mixture modeling for multiframe image denoising," in *IEEE International Conference on Computer Vision*, 2015, pp. 442–449.

[46] Y. Fu, J. Gao, D. Tien, and Z. Lin, "Tensor lrr based subspace clustering," in *International Joint Conference on Neural Networks*, 2014, pp. 1877–1884.

[47] Y. Fu, J. Gao, D. Tien, Z. Lin, and X. Hong, "Tensor lrr and sparse coding-based subspace clustering," *IEEE Transactions on Neural Networks Learning Systems*, vol. 27, no. 10, pp. 2120–2133, 2016.

[48] C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao, "Low-rank tensor constrained multiview subspace clustering," in *IEEE International Conference on Computer Vision*, 2016, pp. 1582–1590.

[49] C. Jia, G. Zhong, and Y. Fu, "Low-rank tensor learning with discriminant analysis for action classification and image recovery," in *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[50] G. Dai and D. Y. Yeung, "Tensor embedding methods." in *National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, Usa*, 2006.

[51] D. Tao, X. Li, X. Wu, and S. J. Maybank, "General tensor discriminant analysis and gabor featuresforgaitrecognition,ieee trans," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 29, no. 10, pp. 1700–1715, 2007.

[52] Q. Li and D. Schonfeld, "Multilinear discriminant analysis for higher-order tensor data classification." *Pattern Analysis & Machine Intelligence IEEE Transactions on*, vol. 36, no. 12, pp. 2524–2537, 2014.

[53] H. Li, T. Jiang, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, p. 157, 2006.

[54] Z. Feng, S. Yang, S. Wang, and L. Jiao, "Discriminative spectralcspatial margin-based semisupervised dimensionality reduction of hyperspectral data," *IEEE Geoscience & Remote Sensing Letters*, vol. 12, no. 2, pp. 224–228, 2015.

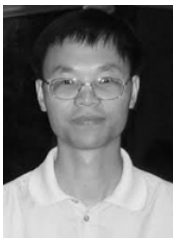
[55] Y. Shen, Z. Wen, and Y. Zhang, "Augmented lagrangian alternating direction method for matrix separation based on low-rank factorization," *Optimization Methods Software*, vol. 29, no. 2, pp. 239–263, 2014.

Jinliang An received the B.S. degree from the School of Information Engineering, Zhengzhou University, and the M.S. degrees from the School of Electronic Engineering, Xidian University, respectively. He is now a Ph.D. student in the school of Artificial Intelligence, Xidian University, Xian, China.

He is also a lecture with the School of Information Engineering, Henan Institute of Science and Technology, Xinxiang 453003, China. His current research interests include machine learning and hyperspectral image processing.

Xiangrong Zhang received the B.S. and M.S. degrees from the School of Computer Science, Xidian University, Xi'an, China, in 1999 and 2003, respectively, and the Ph.D. degree from the School of Electronic Engineering, Xidian University, in 2006.

Currently, she is a professor in the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, Xidian University, China. She has been a visiting scientist in Computer Science and Artificial Intelligence Laboratory, MIT between Jan. 2015 and March 2016. Her research interests include pattern recognition, machine learning, and remote sensing image analysis and understanding.



Huiyu Zhou received the B. Eng in radio technology from the Huazhong University of Science and Technology of China, Wuhan, China, and the M.S. degree in biomedical engineering from the University of Dundee, Dundee, U.K., and the Ph.D degree in computer vision from Heriot-Watt University, Edinburgh, U.K.

He is currently a Reader with the Department of Informatics, University of Leicester, Leicester, U.K. His research interests include computer vision, intelligent systems, and data mining.

Licheng Jiao received the Ph.D. degree from Xian Jiaotong University, Xian, China, in 1990. From 1990 to 1991, he was a Postdoctoral Fellow in the National Key Laboratory for Radar Signal Processing, Xidian University, Xian, China.

Currently, he is the Dean of the Electronic Engineering School and the Director of the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University. He is a Fellow of IEEE and IET. His current research interests include signal and image processing, machine learning, natural computation, and intelligent information processing.