Learning a Robust Local Manifold Representation for Hyperspectral Dimensionality Reduction

Danfeng Hong, *Student Member, IEEE,* Naoto Yokoya, *Member, IEEE,* Xiao Xiang Zhu, *Senior Member, IEEE*

Abstract*—Local manifold learning has been successfully applied to hyperspectral dimensionality reduction in order to embed nonlinear and non-convex manifolds in the data. Local manifold learning is mainly characterized by affinity matrix construction, which is composed of two steps: neighbor selection and computation of affinity weights. There is a challenge in each step: (1) neighbor selection is sensitive to complex spectral variability due to non-uniform data distribution, illumination variations, and sensor noise; (2) the computation of affinity weights is challenging due to highly correlated spectral signatures in the neighborhood. To address the two issues, in this work a novel manifold learning methodology based on locally linear embedding (LLE) is proposed through learning a robust local manifold representation (RLMR). More specifically, a hierarchical neighbor selection (HNS) is designed to progressively eliminate the effects of complex spectral variability using joint normalization (JN) and to robustly compute affinity (or reconstruction) weights reducing collinearity via refined neighbor selection (RNS). Additionally, an idea that combines spatial-spectral information is introduced into the proposed manifold learning methodology to further improve the robustness of affinity calculations. Classification is explored as a potential application for validating the proposed algorithm. Classification accuracy in the use of different dimensionality reduction methods is evaluated and compared, while two kinds of strategies are applied in selecting the training and test samples: random sampling and region-based sampling. Experimental results show the classification accuracy obtained by the proposed method is superior to those state-of-the-art dimensionality reduction methods.

***Index Terms*—Hyperspectral image, dimensionality reduction, local manifold learning, non-uniform data distribution, collinearity**

I. Introduction

HYPERSPECTRAL data is characterized by very rich spectral information, which enables us to detect targets of interest and analyze data attributes more easily, but also introduces drawbacks caused by its high dimensionality. As a result, dimensionality reduction (DR) is a necessary and essential ingredient to address the aforementioned issue. A large number of dimensionality reduction techniques have been developed for a wide range of applications, including image segmentation [1] biometric [2], large-scale data classification [3], image/video analysis [4], visualization [5]. Generally, these dimensionality reduction approaches can be categorized into linear and nonlinear methods.

Classical linear methods, such as principal component analysis (PCA) [6], easily fail to excavate the underlying data structure that lies in the complex real world. Comparatively, many nonlinear techniques, such as Isomap [7], locally linear embedding (LLE) [8], Laplacian eigenmaps (LE) [9], and local tangent space alignment (LTSA) [10], exhibit unique advantages in dimensionality reduction and obtain state-of-the-art results in many fields. These examples of successful use of manifold learning mentioned above have widely attracted the attention of researchers working in the field of hyperspectral data analysis. Owing to merits of manifold learning, which can effectively map nonlinear and non-convex manifolds in low-dimensional space, massive related approaches are introduced into hyperspectral image processing and successfully applied to various tasks, e.g. feature extraction [11][12], classification [13][14][15][16], detection [17][18], and multi-temporal analysis[19]. In addition, it has been proven in [3] that the algorithm performance with global manifold methods is inferior to that with local manifold methods. As a typical and benchmark local manifold learning (LML) method, LLE explores locally linear and globally nonlinear assumptions to effectively capture the underlying intrinsic structure of data. LLE has been successfully applied to hyperspectral classification. Ma et al. [13] integrated LML with improved k-nearest neighbor for hyperspectral classification tasks. In [14], Ma et al. extended their work and proposed a kind of semi-supervised hyperspectral image classification method based on LML. Tang et al. [16]

This work is supported by the Helmholtz Association under the framework of the Young Investigators Group "SiPEO" (VH-NG-1018).

Danfeng Hong, Naoto Yokoya and Xiao Xiang Zhu are with the Remote Sensing Technology Institute (IMF), German Aerospace Center, 82234 Weßling, Germany, and Signal Processing in Earth Observation (SiPEO), Technical University of Munich, 80333 Munich, Germany. (e-mail: danfeng.hong@dlr.de; naoto.yokoya@dlr.de; xiao.zhu@dlr.de)

Naoto Yokoya is also with the Research Center for Advanced Science and Technology, The University of Tokyo, 153-8904 Tokyo, Japan

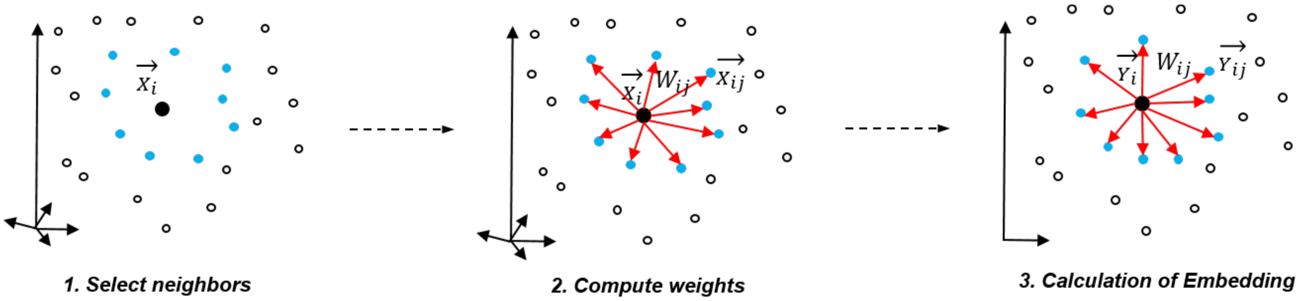


Figure 1. A unified framework of the LML algorithm

proposed manifold-based on sparse representation for hyperspectral classification, and they embedded the local geometric property using local manifold representation into classification framework based on sparse representation in order to enforcedly keep consistent from sparse code to local manifold representation.

Current research on manifold learning methods in hyperspectral data processing mostly focuses on their potential for improving classification and frequently neglects the shortcomings of manifold learning itself. In other words, considerable attention has been paid to feature fusion and classifier design; however, studies on feature representation (manifold learning) are still lacking in the context of dimensionality reduction of hyperspectral data. Consequently, the classification accuracy can be limited by bottlenecks in manifold learning, where a breakthrough in the level of the classifier is hardly made. To this end, a good feature representation can break the stalemate.

In general, LML can be regarded as local graph embedding [20], while the most important part of the graph-embedding framework is the calculation of affinities (or similarities) of vertex pairs in a graph, i.e., the affinity matrix. The construction of the affinity matrix comprises two steps: neighbor selection and computation of affinity weights. There is a challenge in each step: (1) neighbor selection is sensitive to complex spectral variability due to environmental conditions (e.g., illumination and atmospheric conditions) and instrumental configurations (e.g., sensor noise) as well as data inherent structure (e.g., data distribution); (2) the computation of affinity weights is challenging due to highly correlated spectral signatures in the neighborhood. The latter issue is called multi-collinearity when multiple regression analysis is used to obtain affinity weights. More specifically, multi-collinearity refers to the singularity due to highly correlated spectral signatures in the neighborhood, easily resulting in inaccurate estimation of affinity matrix.

To tackle these challenges, it is important to develop a robust and effective local manifold representation approach. In this paper, we mainly focus on improving LLE, which is one of the benchmark LML methods in many fields. A novel LML methodology on the basis of LLE is proposed, which aims at learning a robust local manifold representation (RLMR). Two main contributions of this paper are as follows: Firstly, hierarchical neighbor selection (HNS), which comprises joint normalization (JN) and refined neighbor selection (RNS), has been embedded into the original LLE framework to robustly select neighbors and mitigate multi-collinearity in calculating affinity weights at the same time; Secondly, inspired by successful applications of spatial information in hyperspectral classification, we model the spatial information into the proposed dimensionality reduction methodology in order to further improve the robustness of affinity calculations.

The remainder of this paper is described as follows: in Section 2, we begin with a brief review of LML with three representative LML methods and provide comparative analysis. Section 3 introduces our methodology. Experimental results on classification are presented in Section 4. Finally, we provide conclusions and future outlook in Section 5.

II. LOCAL MANIFOLD LEARNING

In this section, three representative LML methods, i.e., LE, LLE, and LTSA, are introduced in the graph-embedding framework, focusing on their advantages and disadvantages.

Generally, LML methods attempt to capture the underlying local manifold structure of the original data and preserve it in a low-dimensional space, which enables nonlinear dimensionality reduction. Let $\mathrm{X} = [\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_\mathrm{N}] \in \mathbb{R}^{\mathrm{D}\times\mathrm{N}}$ denotes N data samples that have D-dimensional features and $\mathrm{Y} = [\mathrm{y}_1, \mathrm{y}_2, \dots, \mathrm{y}_\mathrm{N}] \in \mathbb{R}^{\mathrm{d}\times\mathrm{N}}$ denotes their low-dimensional representations, where $\mathrm{d} \ll \mathrm{D}$. LML comprised mainly three steps: 1) neighbor selection; 2) computation of affinity weights; 3) calculation of embedding, as illustrated in Figure 1. Pairwise similarity measurements are performed to selected k neighbors for each data sample. Euclidean distance is commonly used for similarity measurement. Let $\mathrm{W} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ be a sparse affinity matrix with the (i,j)-th entry of the matrix representing the affinity weight from the i-th sample and j-th sample, where $\mathrm{j} \in \varphi_\mathrm{i}$ and φ_i is a set of neighbors of the i-th sample. The calculation embedding coordinates is generally formulated as [20]

$$\begin{aligned}\hat{\mathbf{Y}} &= \arg\min_{\mathrm{Y}} \left\{ \sum_{i=1}^{N} \sum_{j\in\varphi_i} \left\| \mathbf{y}_i - \mathbf{y}_j \right\|_2^2 \mathbf{W}_{ij} \right\}, \text{ s.t. } \mathbf{YBY}^{\mathrm{T}} = \mathbf{I} \\ &= \arg\min_{\mathrm{Y}} \left\{ \mathrm{tr}\left(\mathbf{YLY}^{\mathrm{T}} \right) \right\}, \text{ s.t. } \mathbf{YBY}^{\mathrm{T}} = \mathbf{I},\end{aligned} \quad , \quad (1)$$

where $\mathrm{L} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ is the Laplacian matrix defined as $\mathrm{L} = \mathrm{D} - \mathrm{W}$ and $\forall \mathrm{i}\ \mathrm{D}_{\mathrm{ii}} = \sum_{\mathrm{j}\neq\mathrm{i}} \mathrm{W}_{\mathrm{ij}}$ and B is a constant matrix defined by the formulation of each manifold learning method. LML methods can be mainly characterized by the construction of the affinity matrix **W**, as described below.

In the following, three popular LML methods – namely LE, LLE and LTSA – are introduced in details according to the aforementioned unified framework of the LML algorithm:

LE: The basic principle is to compute the affinity matrix for each data point in the original high dimensional space using the Gaussian function as [9]

$$\mathbf{W}_{ij}^{LE} = \begin{cases} \exp\left(-\left\|\mathbf{x}_i - \mathbf{x}_j\right\|_2^2 \big/ 2\sigma^2\right) & if\ \ j \in \varphi_i \\ 0 & otherwise, \end{cases} \tag{2}$$

The constant matrix **B** is defined as **B**=**D**. The low dimensional representations can be obtained by solving the optimization Eq. (1).

LE is a very typical graph-based embedding method, which has been proven in Ref. [9] to be simple to implement and robust against outliers and noise. However, its limitation is also obvious [21], namely, local manifold structure is artificially designed by exploiting approximately pairwise distances with heat kernel, which brings relatively weak representation of local manifold without considering the property of local neighbors.

LLE: It represents the underlying local manifold structure by exploiting the local symmetries of linear reconstructions [5] between each data point and its neighbors in the high-dimensional space and then computes the low-dimensional embedding coordinates that preserve the reconstruction coefficients. The reconstruction coefficients, denoted as $A \in \mathbb{R}^{N\times N}$, are obtained by the minimization

$$\hat{\mathbf{A}} = \arg\min_{\mathbf{A}} \left\{ \sum_{i=1}^{N} \left\| \mathbf{x}_i - \sum_{j\in\varphi_j} \mathbf{A}_{ij}\mathbf{x}_j \right\|_2^2 \right\}, \text{ s.t. } \sum_{j\in\varphi_j} \mathbf{A}_{ij} = 1, \tag{3}$$

where $\mathbf{A}_{ij}$ denotes the reconstruction weight between $\mathbf{x}_i$ and $\mathbf{x}_j$, if the j-th data point is one of the k neighbors of the i-th data point ($j \in \varphi_i$); otherwise $\mathbf{A}_{ij} = 0$. Particularly, the constraint of the sum-to-one shown in Eq. (3) is used on the rows of the reconstruction coefficients A to obey the important local symmetries to be invariant to rotations, rescalings and translations of any target data point and its neighbors [5]. The low dimensional coordinates are obtained by minimizing the embedding cost function as

$$\begin{gathered} \hat{\mathbf{Y}} = \arg\min_{\mathbf{Y}} \left\{ \sum_{i=1}^{N} \left\| \mathbf{y}_i - \sum\nolimits_{j\in\varphi_i} \mathbf{A}_{ij}\mathbf{y}_j \right\|_2^2 \right\}, \\ \text{s.t.} \sum_{i=1}^{N} \mathbf{y}_i = 0,\ \frac{1}{N}\sum_{i=1}^{N} \mathbf{y}_i\mathbf{y}_i^{\mathrm{T}} = \mathbf{I} \end{gathered}, \tag{4}$$

From the viewpoint of the graph-embedding framework, LLE can also be induced as the graph-embedding problem; therefore Eq. (4) can be rewritten in the form of Eq. (1) as

$$\begin{aligned} \hat{\mathbf{Y}} &= \arg\min_{\mathbf{Y}} \left\{ \sum_{i=1}^{N} \left\| \mathbf{y}_i - \sum\nolimits_{j\in\varphi_i} \mathbf{A}_{ij}\mathbf{y}_j \right\|_2^2 \right\}, \text{ s.t. } \mathbf{YBY}^{\mathrm{T}} = \mathbf{I} \\ &= \arg\min_{\mathbf{Y}} \left\{ \sum_{i=1}^{N} \sum_{j\in\varphi_i} \left\| \mathbf{y}_i - \mathbf{y}_j \right\|_2^2 \mathbf{W}_{ij}^{LLE} \right\}, \text{ s.t. } \mathbf{YBY}^{\mathrm{T}} = \mathbf{I} \\ &= \arg\min_{\mathbf{Y}} \left\{ \mathrm{tr}\left(\mathbf{YL}^{LLE}\mathbf{Y}^{\mathrm{T}}\right) \right\}, \text{ s.t. } \mathbf{YBY}^{\mathrm{T}} = \mathbf{I}, \end{aligned} \tag{5}$$

where the affinity matrix ($\mathbf{W}^{LLE}$) can be computed by the following equation [20]

$$\mathbf{W}_{ij}^{LLE} = \begin{cases} \mathbf{A}_{ij} + \mathbf{A}_{ji} - \mathbf{A}_{ij}\mathbf{A}_{ji} & if\ \ j \in \varphi_i \\ 0 & otherwise, \end{cases} \tag{6}$$

and Laplacian matrix of LLE can be given by $\mathbf{L}^{LLE} = \mathbf{D} - \mathbf{W}^{LLE} = (\mathbf{I} - \mathbf{A})^{\mathrm{T}}(\mathbf{I} - \mathbf{A})$ [5] and D is a diagonal matrix defined by $\forall i\ D_{ii} = \sum_{j\neq i} \mathbf{W}_{ij}$. **B** is defined as $\mathbf{B} = \mathbf{I}$.

With local regression technique [22], the property of local data is fully taken into consideration in LLE, which means that local manifold structure can be effectively learned from local data. It is natural that it is able to improve the representation ability of local manifold. That is not to say, however, that robust local manifold representation can be obtained using LLE, since LLE is very sensitive to data distribution [23], variability [24], as well as collinearity.

LTSA: Similar to LLE, LTSA attempts to mine the underlying local manifold structure assuming local linearity. The core idea of LTSA is to utilize a local tangent space to represent a local manifold structure via a linear mapping such as PCA. Therefore, it can be solved naturally as a graph-embedding problem, and the affinity matrix can be defined as $\mathbf{W}^{LTSA} = \mathbf{D} - \mathbf{L}^{LTSA}$, more specifically formulated as follows [14]

$$\mathbf{W}_{ij}^{LTSA} = \begin{cases} \dfrac{1}{k} + \dfrac{1}{k-1}\boldsymbol{\theta}_i^{\mathrm{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\theta}_j & if\ \ j \in \varphi_i \\ 0 & otherwise, \end{cases} \tag{7}$$

where $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_j$ are the local tangent coordinates of $\mathbf{x}_i$ and $\mathbf{x}_j$, and $\boldsymbol{\Lambda}$ stands for the leading d eigenvalues of the covariance matrix of φ_i and k is the number of neighbors for $\mathbf{x}_i$. The low dimensional embedding is calculated by the following minimization

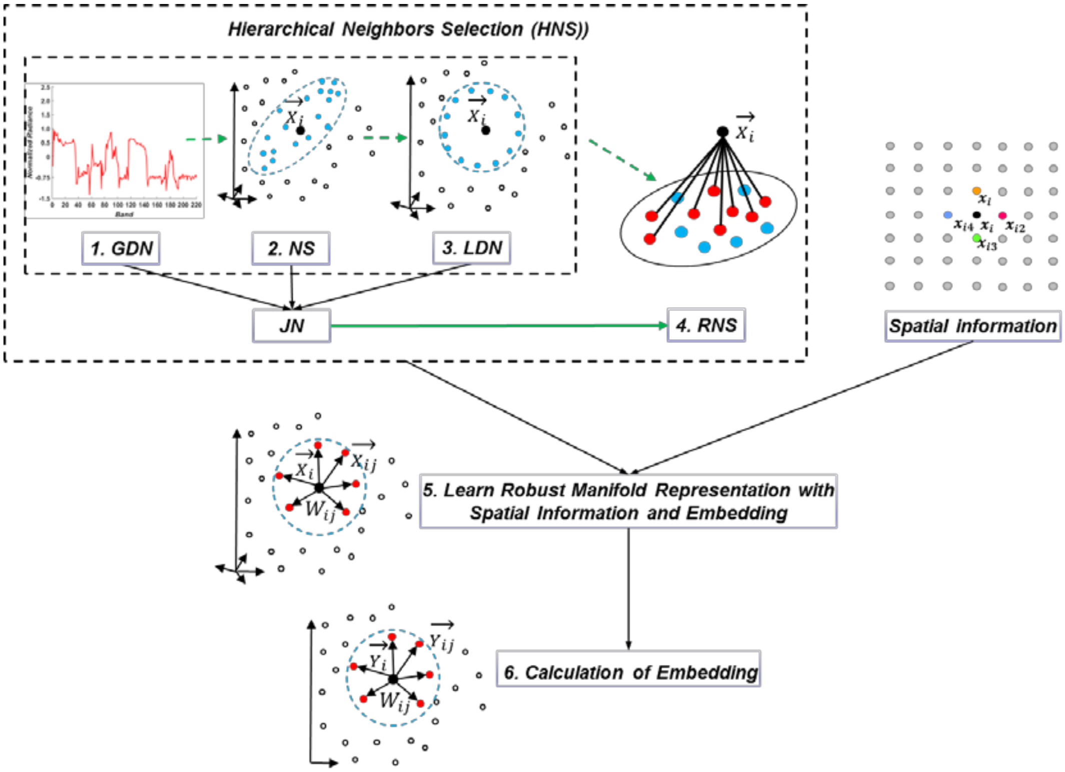

Hierarchical Neighbors Selection (HNS))
1. GDN
2. NS
3. LDN
JN
4. RNS
Spatial information
x_i
x_{i4} x_i x_{i2}
x_{i3}
$\vec{x}_{ij}$
W_{ij}
5. Learn Robust Manifold Representation with Spatial Information and Embedding
6. Calculation of Embedding
$\vec{y}_i$
$\vec{y}_{ij}$

Figure 2. The holistic diagram of the proposed method.

$$\begin{aligned}\hat{\mathbf{Y}} &= \arg\min_{\mathbf{Y}} \left\{\mathrm{tr}\left(\mathbf{Y}\mathbf{L}^{LTSA}\mathbf{Y}^{\mathrm{T}}\right)\right\}, \; s.t.\; \mathbf{YBY}^{\mathrm{T}} = \mathbf{I} \\ &= \arg\min_{\mathbf{Y}} \left\{\sum_{i=1}^{N}\sum_{j\in\varphi_i}\left\|\mathbf{y}_i - \mathbf{y}_j\right\|_2^2 \mathbf{W}_{ij}^{LTSA}\right\}, \; s.t.\; \mathbf{YBY}^{\mathrm{T}} = \mathbf{I} \qquad (8) \\ &= \arg\min_{\mathbf{Y}} \left\{\sum_{i=1}^{N}\left\|\mathbf{y}_i\mathbf{H} - \mathbf{T}_i\boldsymbol{\theta}_i\right\|_2^2\right\}, \; s.t.\; \mathbf{YBY}^{\mathrm{T}} = \mathbf{I}\,,\end{aligned}$$

where $\mathbf{H} = \mathbf{I} - \mathbf{e}\mathbf{e}^{\mathrm{T}}/k$ is the centering matrix, and $\mathbf{e}$ is a uniform vector with the size of $k\times 1$. $\mathbf{T}_i$ is a local transformation matrix with linearity and $\mathbf{B}$ is defined as $\mathbf{B} = \mathbf{I}$.

Typically, a concept of local tangent space is proposed in LTSA to linearly and approximately estimate the local manifold structure, which is able to better capture the intrinsic structure of the underlying manifold [10]. However, such approximated estimation of local manifold structure is possibly inaccurate, particularly in non-uniform distributed data [25], due to those data in local manifold space without lying in, or closing to, a linear subspace. Also, although the performance of LTSA can improve the local manifold representation compared to LLE to some extent, it still fails when taking the data variability (e.g., noise) into consideration [26]. Furthermore, unlike LLE which is able to fully consider geometric structure information of target point and its neighbors by linear regression with constraint, while LTSA explores a linear mapping (e.g. PCA) to find the principle information to depict local manifold structure, accordingly resulting in inevitable loss of useful information (e.g. geometric structure, local minutiae).

In summary, among the three LML methods, one advantage of LLE and LTSA over LE is that by using LLE or LTSA we can obtain a potentially better performance in dimensionality reduction due to their reasonably linear representation in local manifold space. But the drawback of LLE and LTSA is that neither is highly robust against complex data variability, e.g. caused by noise, illumination, or non-uniform data distribution. Therefore, how to robustly learn local manifold representation is an unsolved problem in LML. As a promising LML framework, LLE has been successfully applied in many fields and has obtained some amazing experimental results due to effectively and reasonably local linear assumption, for example in hyperspectral data processing [3][13][14][16][17][22]. However, sensitivity to variability and collinearity when calculating the local linear representation are hindering the advancement of LLE towards robustness and high performance. Therefore, in the next section we emphatically introduce the proposed novel methodology based on LLE in an attempt to address the two issues mentioned above.

III. Robust Local Manifold Representation

In this section, a novel LML methodology is introduced in detail in order to learn a robust local manifold representation (RLMR), mainly including the design of HNS and the integration of spatial contextual information. Figure 2 shows the holistic diagram of the proposed methodology that mainly comprises the six steps given below, where the first four

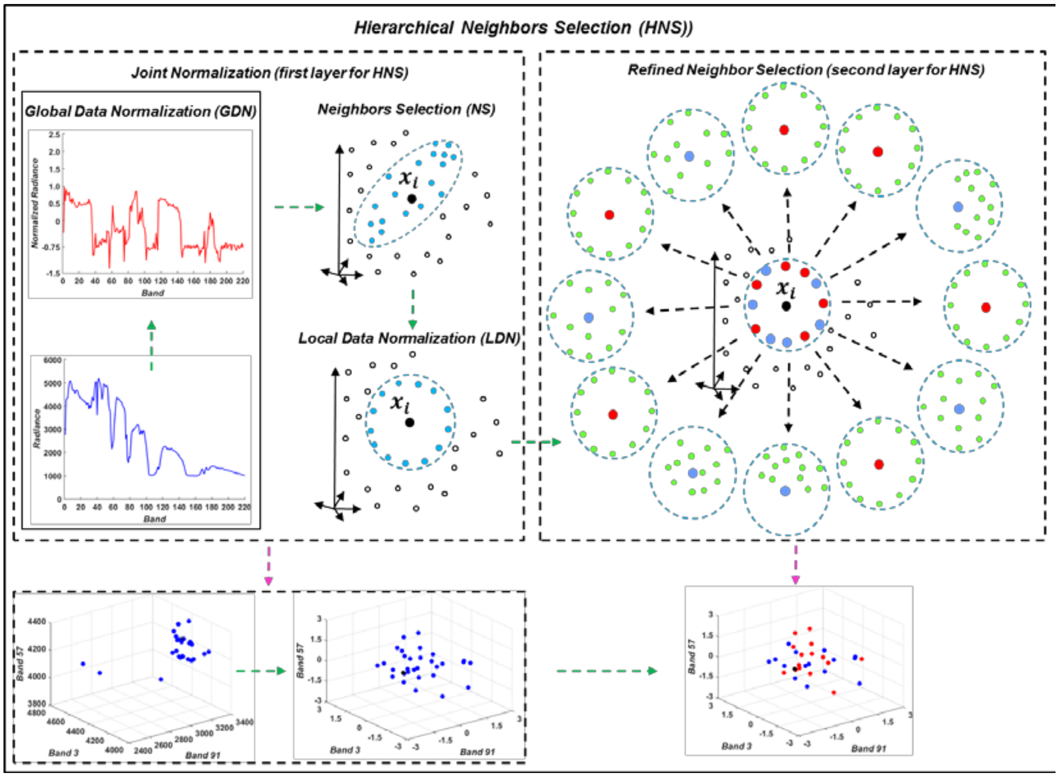

Hierarchical Neighbors Selection (HNS))
Joint Normalization (first layer for HNS)
Global Data Normalization (GDN)
Normalized Radiance
Band
Radiance
Band
Neighbors Selection (NS)
x_i
Local Data Normalization (LDN)
x_i
Refined Neighbor Selection (second layer for HNS)
x_i
Band 57
Band 3
Band 91
Band 57
Band 3
Band 91
Band 57
Band 3
Band 91

Figure 3. The detailed diagram of HNS.

correspond to HNS and the fifth is the integration of spatial information.

Step 1. *Global data normalization (GDN)* is performed to deal with spectral variability modeled by scaling and shifting.

Step 2. *Neighbor selection (NS)* coarsely selects local neighbors of the target pixel.

Step 3. *Local data normalization (LDN)* is applied to make local data distribution more uniform and isotropic and further eliminate locally spectral variability.

Step 4. *Refined Neighbor selection (RNS)* aims at mitigating collinearity in local manifold space, making it possible to obtain a relatively accurate and intrinsic structure of underlying manifold.

Step 5. *Computation of reconstruction weights with contextual information* jointly embeds spectral and spatial information for a robust calculation of the reconstruction weights.

Step 6. *Calculation of embedding* obtains the low-dimensional feature representation by embedding robust local manifold properties into the low-dimensional space.

A. Hierarchical Neighbors Selection (HNS)

Figure 3 shows the detailed diagram of HNS, which is composed of joint normalization (JN) and RNS.

1) JN: Data normalization is widely used in data preprocessing procedure, including hyperspectral data analysis [27][28]. It aims at reducing the effect of numerous variations and improving the performance of subsequent algorithms. Generally, data normalization includes GDN and LDN [29]. The purpose of GDN is to mitigate illumination variations and modify the global data distribution so that it is more uniform and isotropic [30][31], enabling them to be measured in the same, or similar, level or unit. Unlike GDN, LDN tends to uniformize the mean and variance of the local neighborhood, which is particularly useful for non-uniform distributed data [32][33]. Owing to the merits of GDN and LDN, JN is an appropriate approach to effectively address the issues of spectral variability and non-uniform data distribution, which can be implemented step-by-step via the following formulations:

(1) GDN: it performs the following computations:

$$\mathbf{x}_i^{ns} = \frac{\mathbf{x}_i^{o} - c_i^{o}}{s_i^{o}}, \tag{9}$$

$$\mathbf{x}_i^{g} = (\mathbf{x}_i^{ns} - \mathbf{c}^{ns})./\,\mathbf{s}^{ns}, \tag{10}$$

where "./" means the element-wise division, $\mathbf{x}_i^{o} \in \mathbb{R}^{D\times 1}$ is the i-th original spectral signature, c_i^{o} and s_i^{o} are the mean value and variance corresponding to $\mathbf{x}_i^{o}$, respectively. $\mathbf{x}_i^{ns} \in \mathbb{R}^{D\times 1}$ stands for the normalized spectral signature. $\mathbf{X}^{ns} \in \mathbb{R}^{D\times N}$ represents all normalized spectral signatures made up of $\mathbf{x}_i^{ns}$, $\mathbf{c}^{ns} \in \mathbb{R}^{D\times 1}$ and $\mathbf{s}^{ns} \in \mathbb{R}^{D\times 1}$ correspond to the mean value and variance of $\mathbf{X}^{ns}$, respectively. $\mathbf{x}_i^{g} \in \mathbb{R}^{D\times 1}$ stands for the normalized spectral signature of global data normalization.

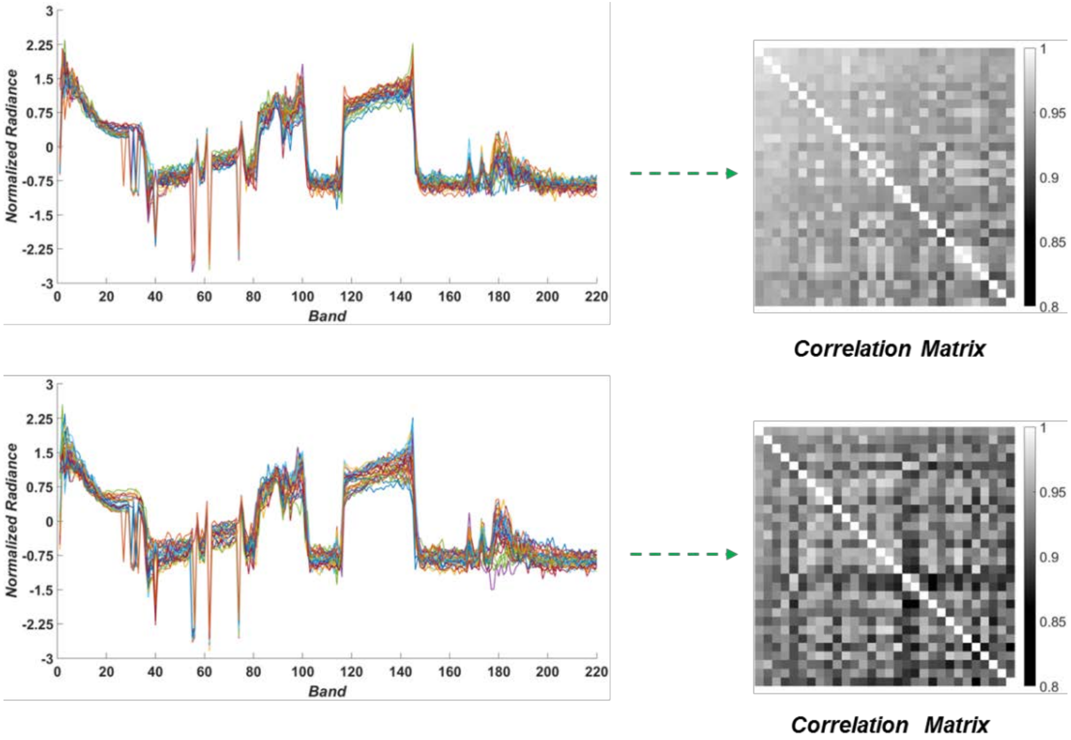


Figure 4. An example to explain and analyze the collinearity in local neighbors (top) without RNS and (bottom) with RNS.

The normalization obtained by performing Eq. (9) can mitigate the effects of spectral variability that can be explained by scaling and shifting, while Eq. (10) makes the global data distribution more uniform and isotropic and puts the same weight on all the spectral bands as shown in the top-left of Figure 3.

(2) LDN: After selecting coarse neighbors for each data point using the Euclidean distance, LDN is exploited to make data distribution more uniform and isotropic in the local manifold space, which can be formulated as

$$\mathbf{x}_{ij}^{l}=(\mathbf{x}_{ij}^{g}-\mathbf{c}_{i}^{g})./\mathbf{s}_{i}^{g},\ j=1,2,\ldots,K\ , \qquad (11)$$

where "./" means the element-wise division, $\mathbf{X}_i^g \in \mathbb{R}^{D\times K}$ consists of the globally normalized spectral features of K nearest coarse neighbors of i-th spectral feature, and $\mathbf{x}_{ij}^g$ $\in \mathbb{R}^{D\times 1}$ is the j-th column vector of $\mathbf{X}_i^g$. $\mathbf{c}_i^g \in \mathbb{R}^{D\times 1}$ and $\mathbf{s}_i^g$ $\in \mathbb{R}^{D\times 1}$ represent the mean value and variance of $\mathbf{X}_i^g$. $\mathbf{X}_i^l=[\mathbf{x}_{i1}^l,\ldots,\mathbf{x}_{ij}^l,\ldots,\mathbf{x}_{jK}^l]\in \mathbb{R}^{D\times K}$represents the final normalized spectral features for i-th data point and its neighbors by JN. An example of local data distribution is shown in the bottom-left of Figure 3. We can see that the data distribution becomes more uniform and isotropic by means of LDN reducing the effects of non-uniform data distribution.

2) RNS: After running JN, we obtain the rough results of neighbor selection where the influence of spectral variability has been removed to a large extent, but multicollinearity still exists among neighbors. As we mentioned above, multicollinearity would lead to inaccurate estimation of affinity matrix, thereby degrade the quality of local manifold structure. As a result, the strategy of refined neighbor selection (RNS) followed by JN is performed against multicollinearity as the second layer of HNS. RNS, which is inspired by local manifold alignment, is proposed to reduce information redundancy [34] in the coarse neighborhood as illustrated in the right of Figure 3. RNS can mitigate the effects of collinearity in the next step, i.e., the calculation of reconstruction weights, while preserving local manifold properties. In detail, LFS is divided into two parts.

First, inspired by Ref. [35][36], we construct the local structure feature $\mathbf{F}_p^{local}$ for the data point p in the feature space using its neighbor's information $\mathbf{X}_p^l=\left[\mathbf{x}_{p1}^l,\ldots,\mathbf{x}_{pj}^l,\ldots,\mathbf{x}_{pK}^l\right]$ $\in \mathbb{R}^{D\times K}$. $\mathbf{F}_p^{local}$ can be formed by the distance property between the feature of p with those of its neighbors using a Gaussian function:

$$F_{pj}^{local}=\exp\left(-\left\|\mathbf{x}_p^l-\mathbf{x}_{pj}^l\right\|_2^2\right) \qquad (12)$$

$$\mathbf{F}_p^{local}=\left[F_{p1}^{local},\ldots,F_{pj}^{local},\ldots,F_{pK}^{local}\right] \qquad (13)$$

The second part is to screen out new local neighbors that have similar data distribution using Kullback–Leibler divergence (KLD). KLD has been justified to effectively measure the similarity of hyperspectral data distribution [37]. The difference of local features $\mathbf{d}^f=\left[d_1^f,\ldots,d_q^f,\ldots,d_K^f\right]\in \mathbb{R}^{1\times K}$ between the point p and its neighbor q can be measured as:

$$d_q^f = KLD\left(\mathbf{F}_p^{local} \,\|\, \mathbf{F}_q^{local}\right) + \alpha KLD\left(\mathbf{F}_q^{local} \,\|\, \mathbf{F}_p^{local}\right), \quad (14)$$

$$KLD\left(\mathbf{F}_p^{local} \,\|\, \mathbf{F}_q^{local}\right) = \sum_{j=1}^{K} F_{pj}^{local} \times \log_2\left(\frac{F_{pj}^{local}}{F_{qj}^{local}}\right), \quad (15)$$

$$KLD\left(\mathbf{F}_q^{local} \,\|\, \mathbf{F}_p^{local}\right) = \sum_{j=1}^{K} F_{qj}^{local} \times \log_2\left(\frac{F_{qj}^{local}}{F_{pj}^{local}}\right), \quad (16)$$

where $\mathbf{F}_p^{local} \in \mathbb{R}^{1\times K}$ and $\mathbf{F}_q^{local} \in \mathbb{R}^{1\times K}$ stand for the local structure features of p and q in the spectral domain, respectively, and α is a penalty parameter balancing the two terms described in Eq. (15) and Eq. (16). Neighbors with the k smallest $\mathbf{d}^f$ value are chosen from the coarse neighbors as the new neighbors of the data point *p*, namely $\mathbf{X}_p^{nl} = \left[\mathbf{x}_{p1}^{nl},\ldots,\mathbf{x}_{pj}^{nl},\ldots,\mathbf{x}_{pk}^{nl}\right] \in \mathbb{R}^{D\times k}$. k is the final number of neighbors for each point, and we make the value of K equal to twofold k.

An example showing the effect of RNS is given in Figure 4, where correlations between the target pixel and its neighbors are shown with and without using RNS. To be specific, given any target pixel, k neighbors need to be selected without RNS, while for RNS, $2k$ neighbors should be selected at first and then k neighbors are refined from $2k$ neighbors. Therefore, the same number of neighbors k can be obtained with RNS and without RNS, respectively. The left of Figure 4 shows spectral signatures of neighbors from two different strategies (with RNS and without RNS) respectively. Although it is not so obvious, it stills emerges the slight difference that spectral signatures without RNS are more intensive than that with RNS, which means that those without RNS are easier to generate singularity when computing the affine matrix (weight matrix). The right of Figure 4 gives relatively obvious results regarding the reduction of collinearity. We can see that the values of correlation matrix with RNS are lower than those without RNS, which demonstrates that the linear correlations observed in the correlation matrix (when value is equal to 1, that means the correlation is up to maximum, and vice versa.) are effectively reduced after using RNS.

B. *Local Manifold Representation with Spatial Contextual Information*

To further improve the robustness of the calculation of reconstruction weights, the spatial information is incorporated into linear reconstructions. We assume that spatially neighboring spectral pixels can be explained by the same or similar reconstruction weights [38], if spatially neighboring pixels include similar spectral components. The calculation of reconstruction weights with spatial contextual information can be formulated based on Eq. (1) by adding the constraint that the reconstruction weights of the target pixel are approximately equal to the average of those of its neighboring pixels, as shown in Eq. (17)

$$\begin{gathered}\mathbf{a}_i^0 = \arg\min_{\mathbf{w}_i^0}\left\{\sum\nolimits_{s=0}^{4}\left\|\mathbf{x}_{is}^{nl} - \mathbf{X}_i^{nl}\mathbf{a}_i^s\right\|_2^2\right\}, \\ \text{s.t.}\ \left\|\mathbf{X}_i^{nl}\left(4\mathbf{a}_i^0 - \sum\nolimits_{s=1}^{4}\mathbf{a}_i^s\right)\right\|_2^2 \le \eta\,,\ \left(\mathbf{a}_i^s\right)^{\mathrm{T}}\mathbf{a}_i^s = 1\,,\ s = 0,1,\ldots,4\end{gathered}\ , \quad (17)$$

where $\mathbf{X}_i^{nl} = \left[\mathbf{x}_{i1}^{nl},\ldots,\mathbf{x}_{ij}^{nl},\ldots,\mathbf{x}_{ik}^{nl}\right] \in \mathbb{R}^{D\times k}$ is the k-nearest neighbors selected by HNS. $\mathbf{x}_{is}^{nl}$, $s = 0,1,\ldots,4$ are the target spectral pixel and its four spatial neighbors, respectively, as an example shown in Figure 5. Correspondingly, $\mathbf{a}_i^s \in \mathbb{R}^{k\times 1}$, $s = 0,1,\ldots,4$ are their reconstruction weights. η is a tiny real number that represents the limit of error.

We can regard Eq. (17) as a joint optimization problem. In this case, the objective function of Eq. (17) can be rewritten as

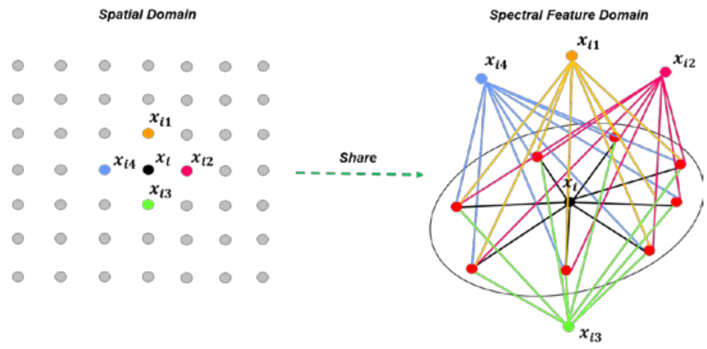


Figure 5. The diagram for Spatial-Spectral combination in hyperspectral DR.

$$\mathbf{a}_i^0 = \arg\min_{\mathbf{a}_i^0}\left\{\left\|\hat{\mathbf{X}}_i^{nl} - \mathbf{L}\hat{\mathbf{A}}_i\right\|_F^2\right\},\ \ \text{s.t.}\ \ \mathbf{C}\hat{\mathbf{A}}_i = [1\ \ 1\ \ 1\ \ 1\ \ 1]^{\mathrm{T}}$$

$$\mathbf{L} = \begin{bmatrix} 4\beta\mathbf{X}_i^{nl} & -\beta\mathbf{X}_i^{nl} & -\beta\mathbf{X}_i^{nl} & -\beta\mathbf{X}_i^{nl} & -\beta\mathbf{X}_i^{nl} \\ \mathbf{X}_i^{nl} & & & & \\ & \mathbf{X}_i^{nl} & & & \\ & & \mathbf{X}_i^{nl} & & \\ & & & \mathbf{X}_i^{nl} & \\ & & & & \mathbf{X}_i^{nl} \end{bmatrix},$$

$$\hat{\mathbf{A}}_i = \begin{bmatrix} \mathbf{a}_i^0 \\ \mathbf{a}_i^1 \\ \mathbf{a}_i^2 \\ \mathbf{a}_i^3 \\ \mathbf{a}_i^4 \end{bmatrix},\quad \hat{\mathbf{X}}_i^{nl} = \begin{bmatrix} \mathbf{0} \\ \mathbf{x}_{i0}^{nl} \\ \mathbf{x}_{i1}^{nl} \\ \mathbf{x}_{i2}^{nl} \\ \mathbf{x}_{i3}^{nl} \\ \mathbf{x}_{i4}^{nl} \end{bmatrix},\quad \mathbf{C} = \begin{bmatrix} \mathbf{e} & & & & \\ & \mathbf{e} & & & \\ & & \mathbf{e} & & \\ & & & \mathbf{e} & \\ & & & & \mathbf{e} \end{bmatrix}, \quad (18)$$

where the sizes of $\mathbf{L}$, $\hat{\mathbf{X}}_i^{nl}$, $\hat{\mathbf{A}}_i$, $\mathbf{C}$ are $6D \times 5k$, $6D \times 1$, $5k \times 1$ and $5 \times 5k$, respectively. And $\mathbf{e} \in \mathbb{R}^{1\times k}$ is the unit vector with a size of $1\times k$, and β is a penalty parameter to balance the importance between error item and constraint item in Eq. (18).

In order to solve Eq. (18), it can be further relaxed by means of Lagrange multipliers as represented by

$$\mathbf{a}_i^0 = \arg\min_{\mathbf{a}_i^0}\left\{\left\|\hat{\mathbf{X}}_i^{nl} - \mathbf{L}\hat{\mathbf{A}}_i\right\|_F^2 + \lambda\left\|\mathbf{C}\hat{\mathbf{A}}_i - \hat{\mathbf{e}}\right\|_2^2\right\}, \qquad (19)$$

where λ is also a penalty parameter, and here let it be 1 for simplicity as well as $\hat{\mathbf{e}} = [1 \quad 1 \quad 1 \quad 1 \quad 1]^{\mathrm{T}} \in \mathbb{R}^{5\times 1}$. The solution in Eq. (19) can be analytically derived [39] by matrix derivation operation as

$$\mathbf{a}_i^0 = \left(\mathbf{L}^{\mathrm{T}}\mathbf{L} + \lambda\mathbf{C}^{\mathrm{T}}\mathbf{C}\right)^{-1}\left(\mathbf{L}^{\mathrm{T}}\hat{\mathbf{X}}_i^{nl} + \lambda\mathbf{C}^{\mathrm{T}}\hat{\mathbf{e}}\right), \qquad (20)$$

Therefore, $\mathbf{a}_i^0$ is the weight vector for *i*-th pixel by using RLMR. Following the framework shown in the Figure 2, the result of dimensionality reduction can be obtained by calculating the embedding using Eq. (1).

IV. Experiment

In this section, we explore classification as a potential application and quantitatively evaluate the performance of DR algorithms using overall classification accuracy. The main focus of this paper is to learn a more robust and discriminative feature representation, rather than how to develop a more advanced classifier. Therefore, we use two common classifiers, namely the nearest neighbor (NN) algorithm based on the Euclidean distance and linear SVM.

A. Hyperspectral datasets

The experiments are carried out using two benchmark hyperspectral datasets.

1) Indian Pine AVIRIS image: The first image-set was acquired by NASA's AVIRIS sensor over the Indian Pine test site in Northwest Indiana with the size of $145\times145\times220$ and 10 nm spectral resolutions over the range of 400-2500 nm, mainly including several kinds of vegetation. More specific classes and the number of samples can be found in Table 1.

2) 2013 IEEE GRSS Data Fusion contest image: The second image-set was provided for the 2013 IEEE GRSS Data Fusion contest acquired by the ITRES-CASI 1500 sensor with the size of $349\times1905\times144$ in the range of 380-1050 nm, which includes more varied categories.

B. Results of Indian Pine AVIRIS image

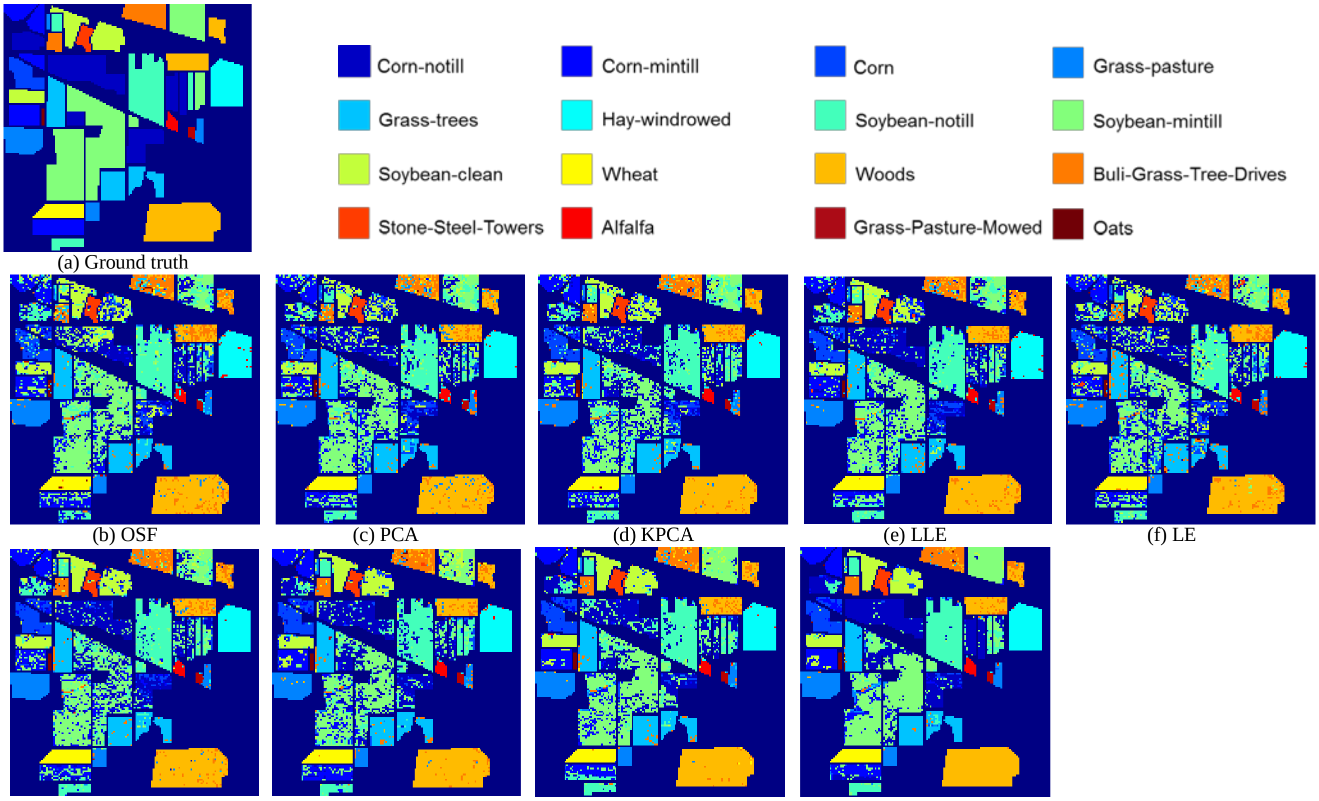
For the first image-set, we adopted two sampling strategies to select training samples and test samples: random sampling and region-based sampling. Random sampling is a common way for the validation of hyperspectral classification. In contrast, classification using region-based sampling is more practical and challenging due to high correlation and limited variability of training samples, and thus an effective way to investigate the performance of the proposed method. We randomly assigned around 5% of total samples as cross-validation samples and then divided the rest into two parts: training samples (5% of total samples), by random sampling or region-based sampling, and test samples (90% of total samples). Moreover, ten replications were performed for selecting training and test samples based on the two aforementioned sampling strategies. The specific number of cross-validation, training, and test samples is listed in Table 1 [40]. We compare the classification results on dimensionality-reduced data using the proposed method with those using some benchmark DR methods (PCA, KPCA [41], LLE, LE, and LTSA) and original spectral features (OSF). Three step-by-step methods, i.e., JN, HNS, and RLMR, are used for the proposed methods to investigate the effects of JN, LFS, and the integration of spatial information.

Table 1 The number of training samples and test samples for each class

NO.	Class Name	Total	Cross-validation	Training	Testing
1	Corn-Notill	1434	50	50	1334
2	Corn-Mintill	834	50	50	734
3	Corn	234	50	50	134
4	Grass-Pasture	497	50	50	397
5	Grass-Trees	747	50	50	647
6	Hay-Windrowed	489	50	50	389
7	Soybean-Notill	968	50	50	868
8	Soybean-Mintill	2468	50	50	2368
9	Soybean-Clean	614	50	50	514
10	Wheat	212	50	50	112
11	Woods	1294	50	50	1194
12	Bldg-Gra-Tr-Driv	380	50	50	280
13	Stone-Stel-Towers	95	15	15	65
14	Alfalfa	54	10	10	34
15	Grass-Past-Mowed	26	5	5	16
16	Oats	20	5	5	10

1) Performance Comparison and Analysis between RLMR and Classical DR methods

Initially, we conducted a five-fold cross-validation on training samples in order to select the optimal parameter combination. Table 2 shows the classification accuracies obtained by using the nine methods with optimal parameters (*d*, *k*). It should be noted that two kinds of classification accuracy are applied here, including overall accuracy (total classification accuracy of all classes) and average accuracy (the average of classification accuracy of each class), to evaluate the performance of the listed methods. The proposed methods outperform the other methods both with random sampling and region-based sampling. Compared to OSF, JN, HNS, and RLMR increase the overall accuracy by 8.25%, 12.71%, and 21.1%, respectively, with random sampling, and 7.42%, 8.83%, and 10.46%, respectively, with region-based sampling. For the average accuracy, on the other hand, the corresponding increases are respectively 10.2%, 12.89%, 18.11% with random sampling, and 9.68%, 10.95%, 11.54% with region-based sampling. The classification maps are shown in Figure 6 and Figure 7. It can be seen that the classification maps of JN, HNS, and RLMR include less salt-and-pepper errors. In particular, those of RLMR are smoother in the local spatial region, resulting from the embedding of spatial information. These results demonstrate the effectiveness of all three technical components of the RLMR, i.e., JN, RNS, and the integration of spatial information, and imply that they successfully contribute to extracting robust and discriminative

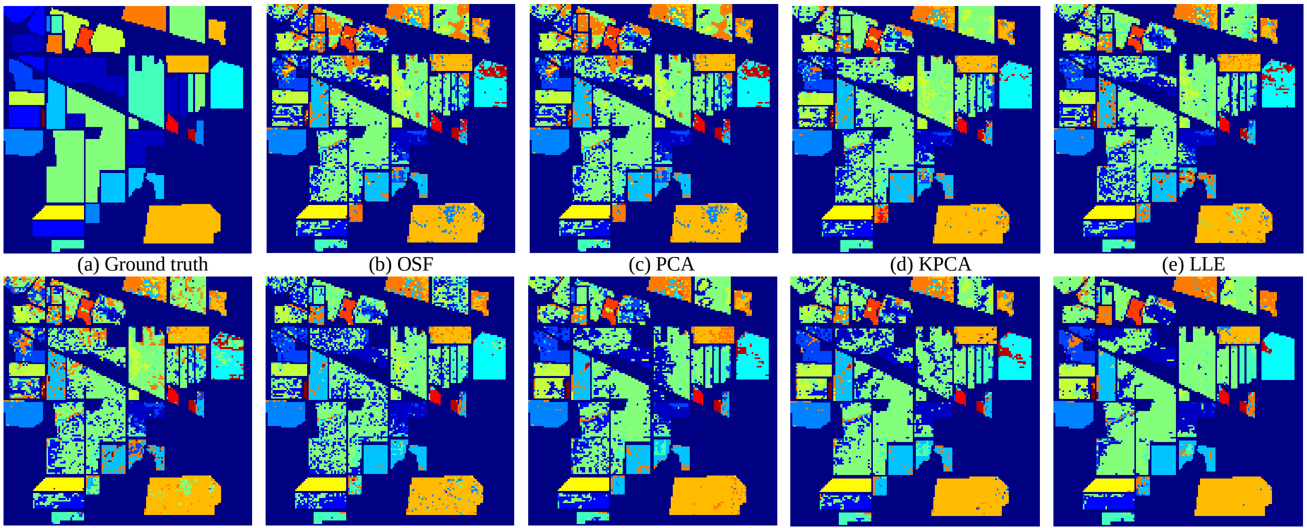


(a) Ground truth

(b) OSF (c) PCA (d) KPCA (e) LLE (f) LE

(g) LTSA (h) JN (i) HNS (j) RLMR

Figure 6 Classification maps for the different methods via NN using the random sampling strategy corresponding to the parameters in Table 2. (a)-(j) are the results for ground truth, OSF, PCA, KPCA, LLE, LE, LTSA, JN, HNS, RLMR, respectively.



(a) Ground truth (b) OSF (c) PCA (d) KPCA (e) LLE

(f) LE (g) LTSA (h) JN (i) HNS (j) RLMR

Figure 7 Classification maps for the different methods via NN using the region-based sampling strategy corresponding to the parameters in Table 2. (a)-(j) are the results for ground truth, OSF, PCA, KPCA, LLE, LE, LTSA, JN, HNS, RLMR, respectively.

low-dimensional feature representations. In contrast, the classification accuracies of the classical LML methods (e.g., LLE, LTSA) are holistically higher than those obtained by using OSF and PCA, and yet lower than the results of our proposed methods due to the sensitivity of variability with respect to LLE and the unavoidable loss of information with respect to LTSA. As for the performance of LE, it is even inferior to the performances of OSF and PCA, and considerably lower than LLE and LTSA, as discussed in Section 2. This indicates that the performance of these methods is unstable in DR due to challenges involved in neighbor selection and affinity calculations.

As shown in Table 2 and Figure 6 and Figure 7, the classification results change substantially when using the two different sampling strategies. The classification accuracy of region-based sampling is much lower than those of random sampling, whereas our methods JN, HNSP, and RLMR are still superior to other methods, even though their classification performance sharply degrades as expected. It should be noted that, as a result of its full use of spatial information, the proposed RLMR leads to a smooth classification map. However, the rate of misclassification using region-based

TABLE 2 CLASSIFICATION ACCURACIES USING OPTIMAL PARAMETER COMBINATION VIA NN FOR DIFFERENT DR METHODS IN INDIAN PINE DATASET

Method	Optimal parameter combination	Classification accuracy			
		Random sampling		Region-based sampling	
		Overall accuracy	Average accuracy	Overall accuracy	Average accuracy
OSF	/	64.74%	72.72%	44.78%	56.67%
PCA	*d*=50	64.62%	72.66%	44.74%	56.64%
KPCA	*d*=50, *v*=10	66.95%	76.03%	48.79%	61.25%
LLE	*d*=60, *k*=40	68.49%	75.51%	47.45%	59.55%
LE	*d*=60, *k* =7	59.57%	68.19%	40.92%	52.73%
LTSA	*d*=60, *k*=70	71.22%	81.12%	51.63%	66.09%
JN	*d*=70, *k*=40	72.99%	82.92%	52.20%	66.35%
HNS	*d*=70, *k*=40	77.45%	85.61%	53.61%	67.62%
RLMR	*d*=50, *k*=80	**85.84%**	**90.83%**	**55.24%**	**68.21%**

sampling for training data is still so high that many integrated regions are misclassified completely. This is caused by limited training samples, as shown in Figure 7.

In order to effectively support the conclusion obtained by nearest neighbor classifier, an advanced and common classifier –SVM [44] is also applied for classification under the same condition. Classification accuracies obtained via SVM and corresponding optimal parameters for nine methods are listed in the Table 3. Figure 7 and Figure 7 shows classification maps for the different methods using the random sampling and region-based sampling strategies respectively. Note that SVM is usually categorized by three versions: linear SVM, polynomial SVM and kernel SVM. In this paper, a linear version of SVM is selected for classification task in order to clarify that the contribution of nonlinear properties is from the reduced feature extracted from manifold learning rather than kernel-based SVM.

In addition, we can observe that the performance of JN, HNS and RLMR is progressively which can be contributed by the used of normalization, RNS and spatial information, respectively. However, it is still lack of an explanation and proof that how important or effective RNS is. Consequently, an additional experiment is performed to compare the performance with RNS and without RNS, listed in Table 4. We can clearly see that the classification accuracies of those methods with RNS are stably higher than those without RNS as well as the proposed method JN+RNS (HNS) obtains the best performance.

2) *Sensitivity Analysis of Parameters and Robustness against Noise*

a. Sensitivity Analysis of Parameters

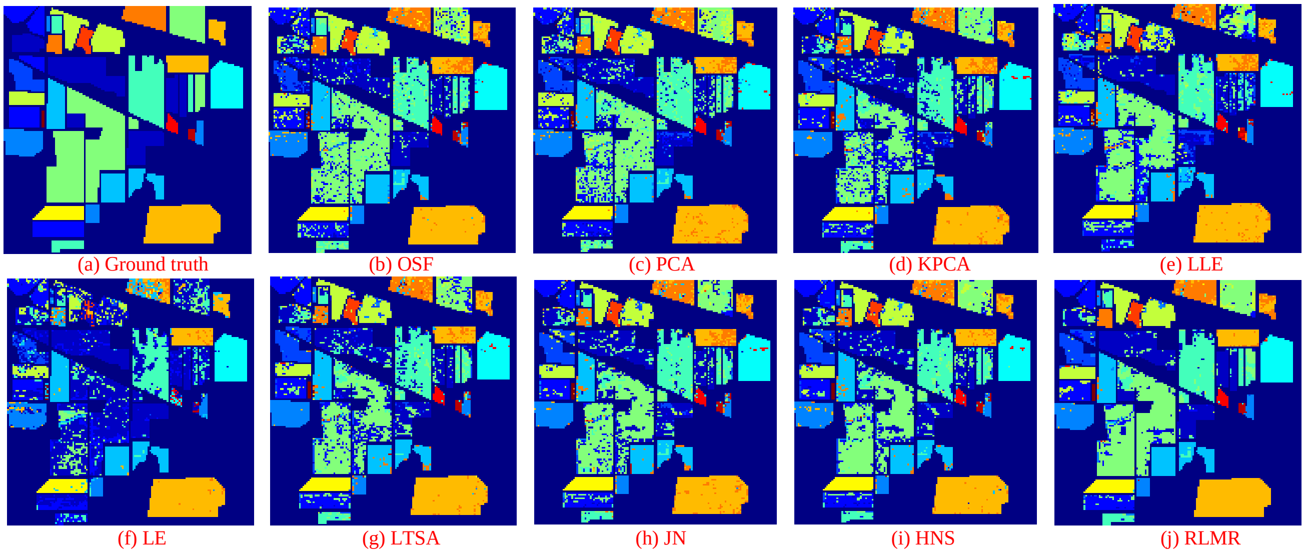
The sensitivity of parameters is examined by varying the number of neighbors (*k*) and the size of reduced dimensionality (*d*) for local manifold learning methods, and the variance (*v*) of kernel for KPCA. As shown in Figure 10 and Figure 11, the performance of the LML methods is less sensitive to the parameters. In general, as observed from the data dimensionality point of view, classification accuracy increases with increasing dimensionality, to a certain extent, and then holds steady. When the reduced dimensionality *d* reaches approximately 50, the results are basically stable for those ML based methods, while the number of neighbors *k* is around 60 when accuracy reaches the nearly optimum level. As the number of neighbors gradually increases, the corresponding classification accuracy progressively increases to a peak (e.g., *k* is equal to around 50) and then dramatically drops. A large number of neighbors may obscure the local structure, whereas a small number of neighbors may not

TABLE 3 CLASSIFICATION ACCURACIES USING OPTIMAL PARAMETER COMBINATION VIA SVM FOR DIFFERENT DR METHODS IN INDIAN PINE DATASET

Method	Optimal parameter combination	Classification accuracy			
		Random sampling		Region-based sampling	
		Overall accuracy	Average accuracy	Overall accuracy	Average accuracy
OSF	/	73.86%	76.04%	47.39%	61.87%
PCA	*d*=30	70.60%	79.50%	47.82%	58.38%
KPCA	*d*=60, *v*=10	72.16%	80.88%	50.36%	63.52%
LLE	*d*=40, *k*=50	71.47%	72.51%	47.23%	62.49%
LE	*d*=80, *k* =3	56.93%	65.06%	36.59%	52.85%
LTSA	*d*=40, *k*=70	75.49%	84.93%	52.79%	64.51%
JN	*d*=90, *k*=60	76.52%	83.03%	52.83%	66.95%
HNS	*d*=100, *k*=50	78.75%	85.04%	54.73%	68.03%
RLMR	*d*=40, *k*=90	**87.06%**	**90.93%**	**56.92%**	**69.24%**

TABLE 4 PERFORMANCE COMPARISON: CLASSIFICATION ACCURACIES USING OPTIMAL PARAMETER COMBINATION FOR DIFFERENT DR METHODS IN INDIAN PINE DATASET

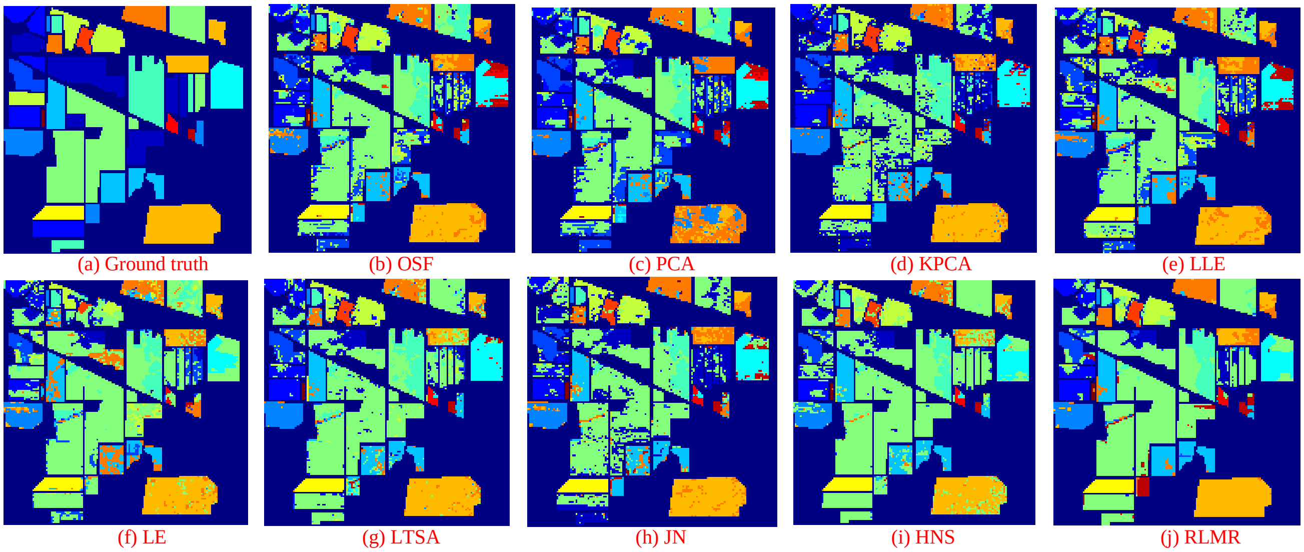
Method	Optimal parameter combination	Classification accuracy	
		Random sampling	Region-based sampling
EU+LLE	*d*=60, *k*=40	68.49%	47.45%
EU+RNS	*d*=90, *k*=50	70.24%	48.85%
SAM+LLE	*d*=60, *k*=80	70.85%	48.97%
SAM+RNS	*d*=70, *k*=50	72.67%	49.50%
JN	*d*=70, *k*=40	72.99%	52.20%
JN+RNS (HNS)	*d*=70, *k*=40	**77.45%**	**53.61%**



(a) Ground truth (b) OSF (c) PCA (d) KPCA (e) LLE

(f) LE (g) LTSA (h) JN (i) HNS (j) RLMR

Figure 8 Classification maps for the different methods via SVM using the random sampling strategy corresponding to the parameters in Table 3. (a)-(j) are the results for ground truth, OSF, PCA, KPCA, LLE, LE, LTSA, JN, HNS, RLMR, respectively.



(a) Ground truth (b) OSF (c) PCA (d) KPCA (e) LLE

(f) LE (g) LTSA (h) JN (i) HNS (j) RLMR

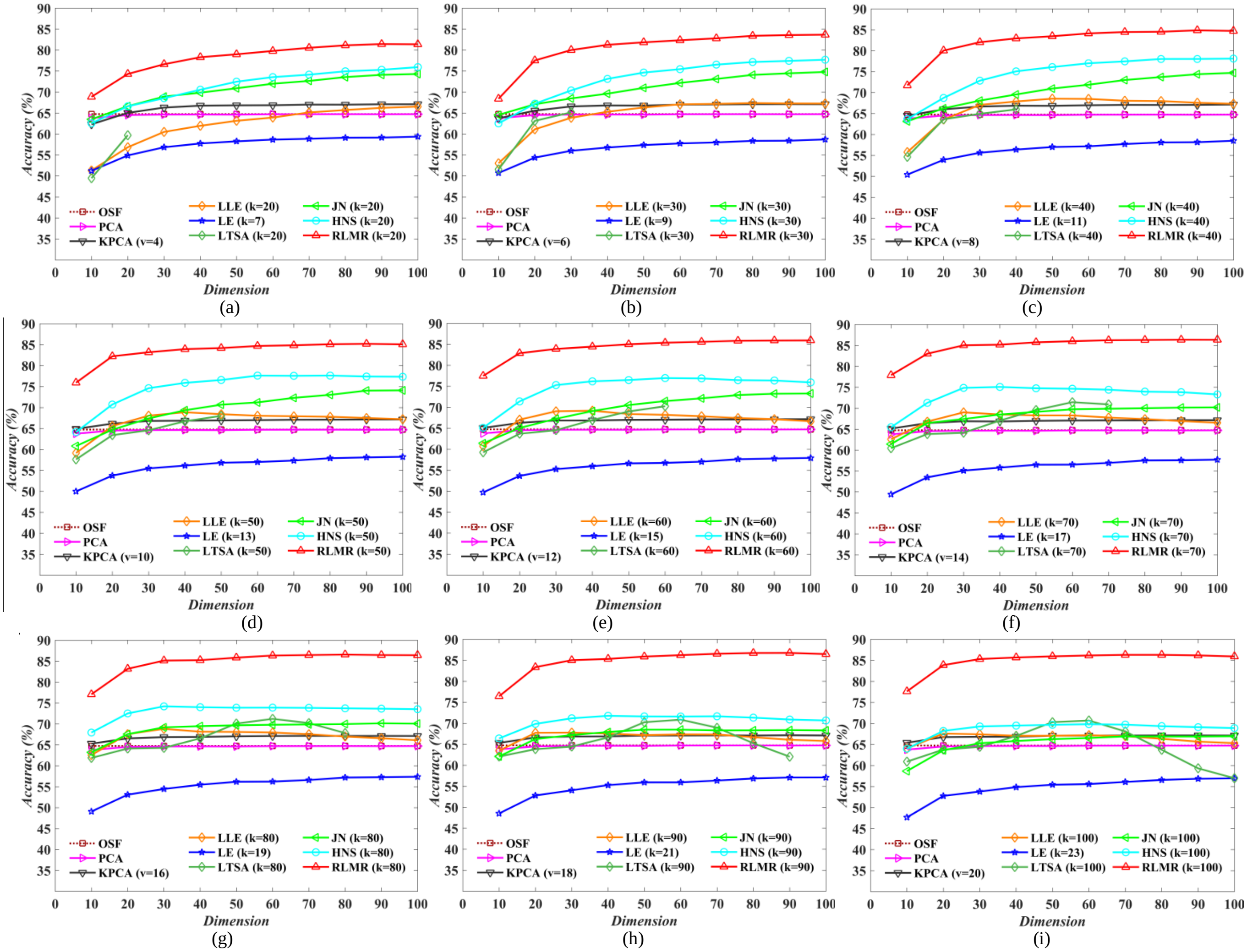
Figure 9 Classification maps for the different methods via SVM using the region-based sampling strategy corresponding to the parameters in Table 3. (a)-(j) are the results for ground truth, OSF, PCA, KPCA, LLE, LE, LTSA, JN, HNS, RLMR, respectively.

sufficiently represent the local structure, causing the degradation of the DR performance. Proper parameters are determined from Figure 10 and Figure 11, which are basically consistent with parameter selection defined via cross-validation shown in Table 2, where the LML methods are used for classification.

However, it is worth noting that due to robustness of our proposed method (RLMR), its results remain stable with the increase in the number of neighbors k and reduced dimensionality d. Conversely, the performances of JN and HNS are progressively degrading with the change of parameters; particularly in a situation with a large k, the classification accuracies even degrade to a level similar to classical LML methods.

Unlike manifold learning methods, the size of reduced dimensionality (d) is the only parameter for PCA, and a limited number of d, around 30, is sufficient to obtain the best classification accuracy. Compared to PCA, KPCA shows a better performance owing to its advantage to capture nonlinear properties of the data; however, the parameter selection of kernel is important.

Except for the two parameters: the number of neighbors (k) and the size of reduced dimensionality (d) , there are still several parameters in the proposed method, including α in RNS (Eq.(14)), penalty parameter λ (Eq.(19)) and the number of spatial neighbors (Eq. (17)) . The parameter α is to balance similarity generated by KLD from the point of view of data distribution. KLD between target point and its neighbors consists of two parts: one is the similarity of data distribution from target point to its neighbors (TPN) and another is the similarity of data distribution form neighbors to target point (NTP). Obviously, the similarity of TPN should be more important than that of NTP, which means the parameter α should be less than 1 as shown in Figure 12(a), therefore this optimal value is 0.2 corresponding to the best classification accuracy. Accordingly, the λ is set to 1 as shown in Figure

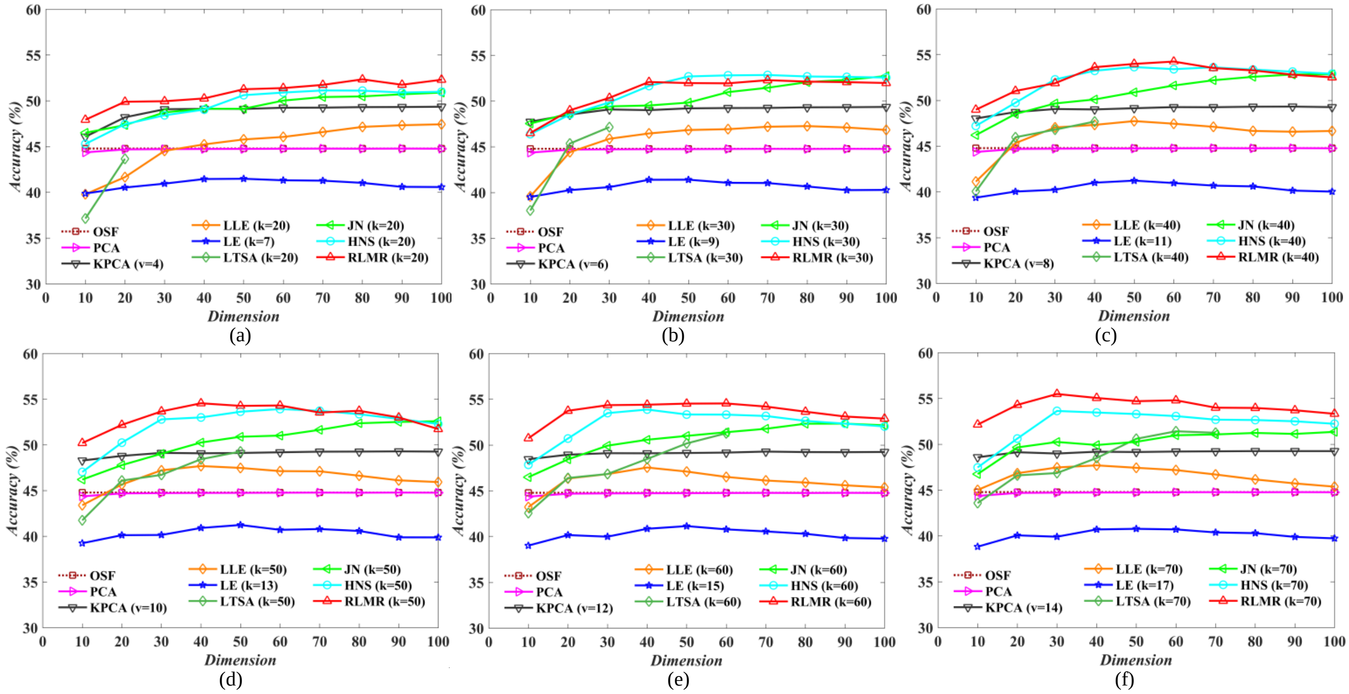


(a) (b) (c)

(d) (e) (f)

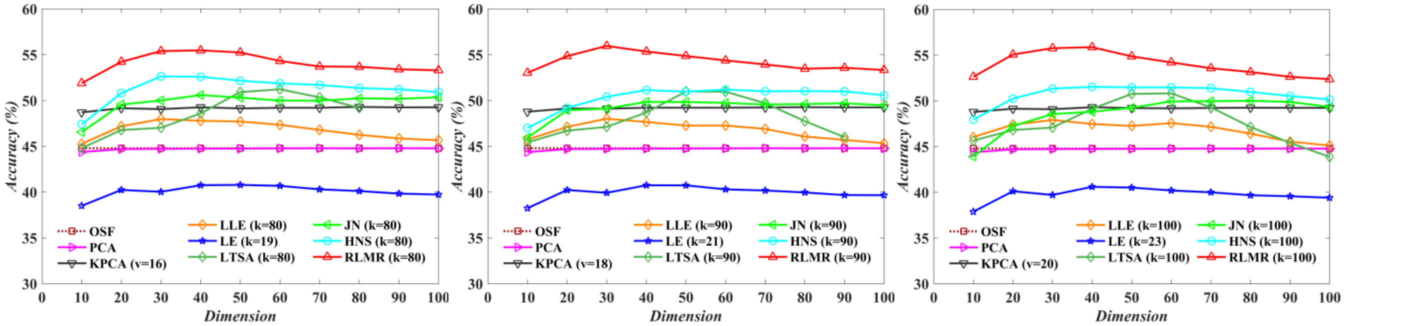
(g) (h) (i)

Figure 10 Performance comparison: Classification accuracy as a function of data dimension using random sampling. (a)-(i) are the results using a different number of neighbors, respectively.



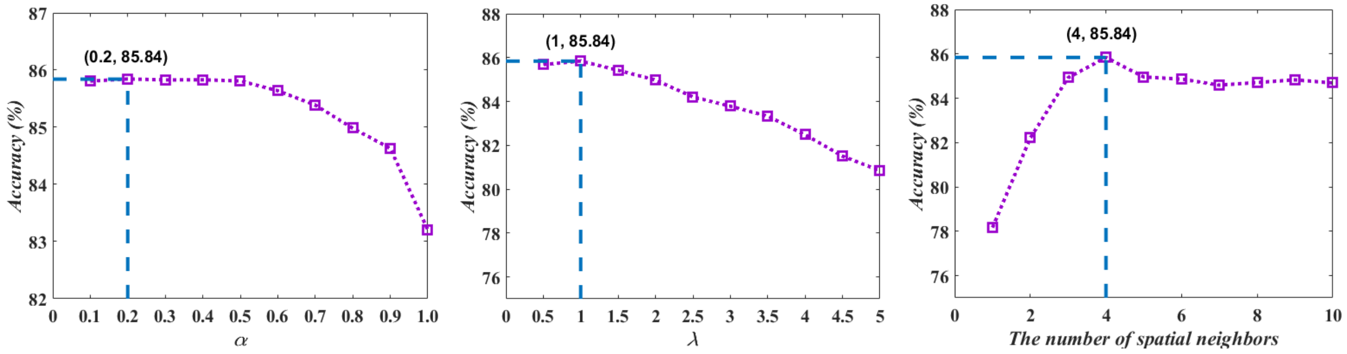
(a) (b) (c)

(d) (e) (f)



(g) (h) (i)

Figure 11 Performance comparison: Classification accuracy as a function of data dimension using region-based sampling. (a)-(i) are the results using a different number of neighbors, respectively.



(a) (b) (c)

Figure 12 The optimal parameter analysis corresponding to the best classification accuracy on the Indian Pine dataset

12(b) in order to balance embedding term and spatial constraint term in Eq. (19). While for the parameter - the number of spatial neighbors, over-larger or over-smaller one would result in over-used or under-used of spatial contextual information. As a result, the value of this parameter should be selected eclectically and in terms of the best classification accuracy observed from Figure 12(c), it is set as 4.

b. Robustness Analysis

In order to validate the robustness of RLMR, a further experiment is performed, which adds noise with a different signal-to-noise-ratio (SNR) into the AVIRIS Indian Pine image. The Gaussian noises are added to the image band by band with the same SNR. Classification was performed with various SNRs to investigate the robustness of the DR algorithms against noise. Figure 13 shows the classification accuracies under the two sampling strategies. As the SNR decreases, the performance of JN, HNS, and RLMR are comparatively stable and superior compared to those of classical ML methods, PCA, KPCA, and OSF. This demonstrates the robustness of the proposed method against noise and implies its effectiveness for low SRN hyperspectral images.

C. Results of 2013 IEEE GRSS Data Fusion contest image

Similarly, we obtained the classification accuracies for the nine methods under the optimal parameters tuned by five-fold cross-validation via NN and SVM classifiers using the given training samples in DFC, as listed in Table 5 and Table 6. As

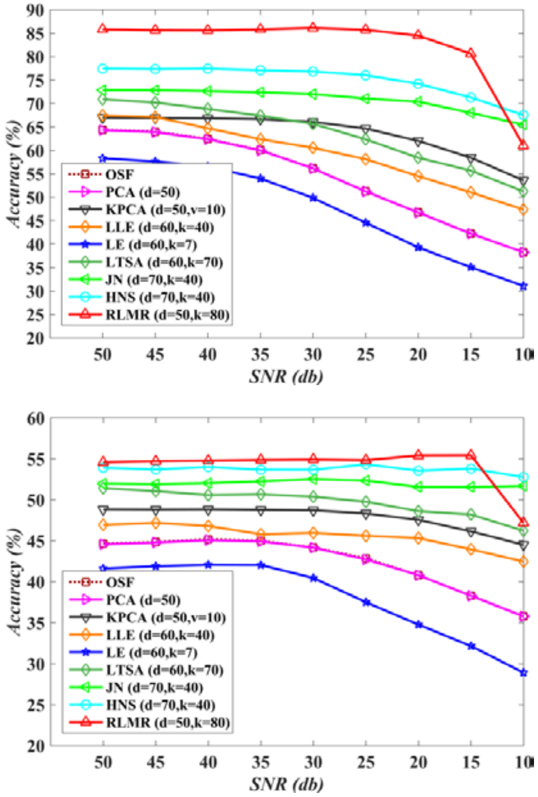


Figure 13 Classification accuracy using the parameters in Table 2 for the different DR methods under the two sampling strategies on the Indian Pine dataset with different SNRs.

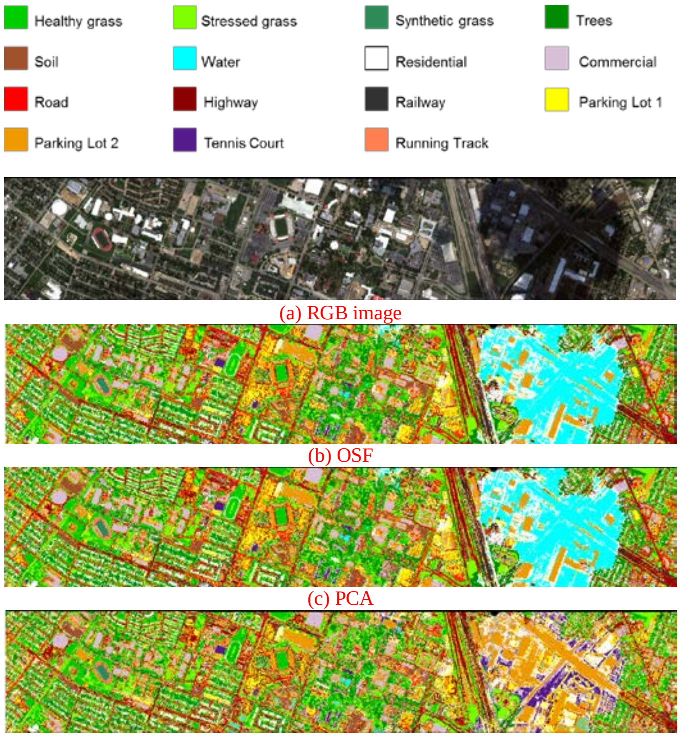
TABLE 5 CLASSIFICATION ACCURACIES USING OPTIMAL PARAMETER COMBINATION VIA NN FOR DIFFERENT DR METHODS IN DFC

Method	Optimal parameter combination	Classification accuracy	
		Overall accuracy	Average accuracy
OSF	/	72.83%	76.16%
PCA	*d*=50	72.85%	76.19%
KPCA	*d*=50, *v*=10	73.80%	77.79%
LLE	*d*=40, *k*=50	74.23%	77.49%
LE	*d*=60, *k* =20	66.70%	70.66%
LTSA	*d*=40, *k*=50	75.40%	78.75%
JN	*d*=60, *k*=50	77.45%	80.69%
HNS	*d*=80, *k*=70	78.52%	81.75%
RLMR	*d*=70, *k*=50	**80.87%**	**82.77%**

TABLE 6 CLASSIFICATION ACCURACIES USING OPTIMAL PARAMETER COMBINATION VIA SVM FOR DIFFERENT DR METHODS IN DFC

Method	Optimal parameter combination	Classification accuracy	
		Overall accuracy	Average accuracy
OSF	/	74.68%	77.84%
PCA	*d*=30	74.78%	77.79%
KPCA	*d*=30, *v*=10	75.12%	78,14%
LLE	*d*=60, *k*=40	75.33%	78.03%
LE	*d*=20, *k* =30	70.71%	72.98%
LTSA	*d*=30, *k*=50	76.04%	79.18%
JN	*d*=70, *k*=60	77.86%	80.12%
HNS	*d*=90, *k*=60	78.98%	82.01%
RLMR	*d*=90, *k*=100	**81.13%**	**82.79%**

can be seen in Table 5 and Table 6, RLMR outperforms the other methods in DFC dataset. This demonstrates that the proposed novel ML method can indeed obtain the good feature representation, thereby further improving the classification accuracy. What's more, similar results contributing similar conclusions are obtained, even though using different classifiers, e.g., the nearest neighbor algorithm, linear SVM, resulting in the effectiveness and robustness of the proposed method.

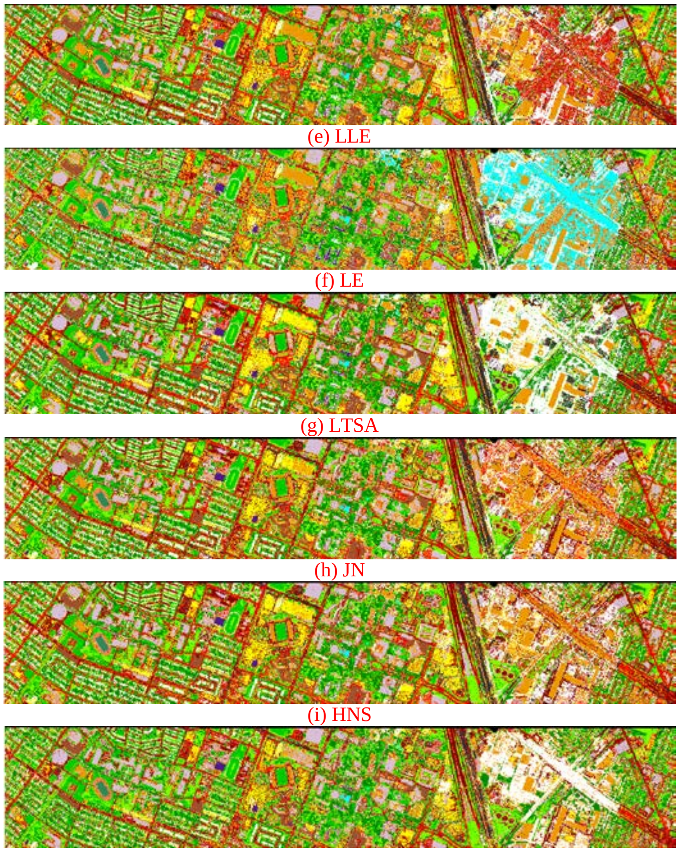


(a) RGB image

(b) OSF

(c) PCA

(d) KPCA



(e) LLE

(f) LE

(g) LTSA

(h) JN

(i) HNS

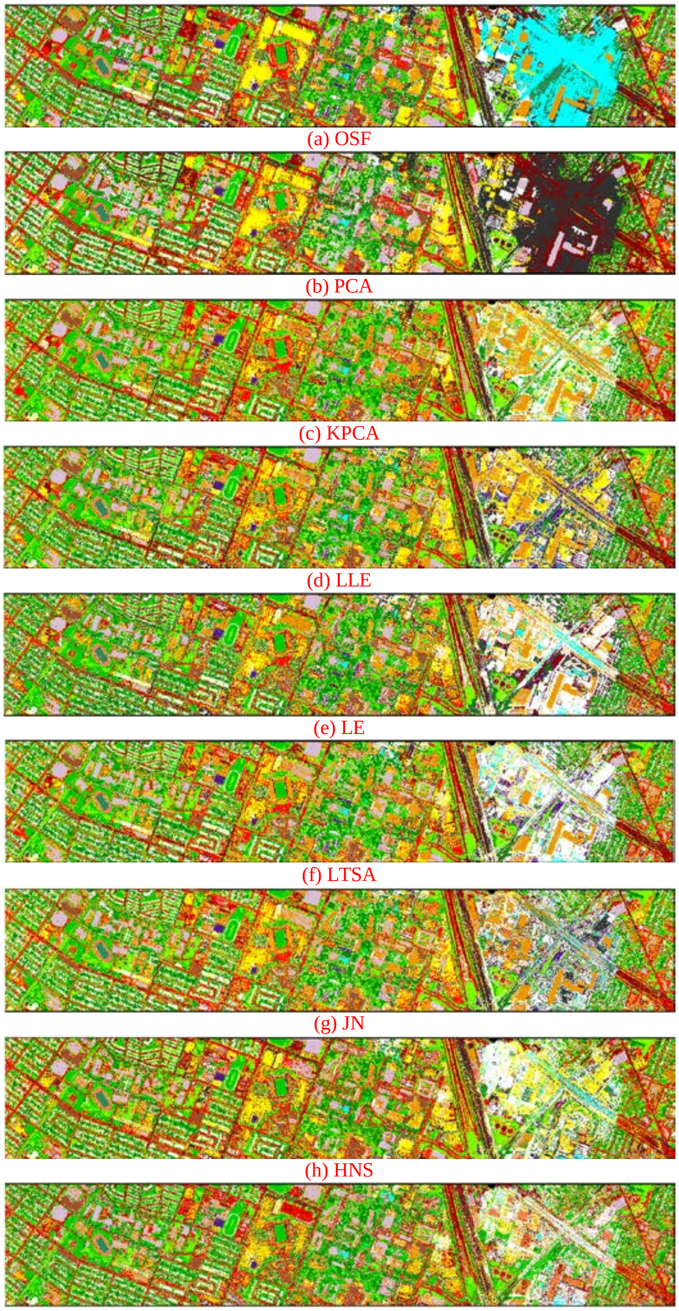
(j) RLMR

Figure 14 Classification maps on the DFC dataset via NN classifier using the different DR methods with the optimal parameters tuned as given in Table 4. (a) is an RGB image from the original hyperspectral image. (b)-(j) are the results using OSF, PCA, KPCA, LLE, LE, LTSA, JN, HNSP, RLMR, respectively.

For simplicity, a general framework for the out-of-samples extension of ML proposed by Bengio [42][43] is used in this paper in order to obtain the full classification map. The out-of-samples extension can be separated into two parts: first, an appropriate kernel function should be constructed (Here, a Gaussian kernel is chosen) ; next, Nystrom formulation should be applied for the generalization of a new data point. Classification maps for different DR methods using the aforementioned optimal parameters are given in Figure 14 and Figure 15, respectively corresponding to NN and SVM classifiers. As shown in Figure 14 (a), the east side of the scene is covered with shadows of clouds, resulting in the performance degradation of those previous DR methods – such as in Figure 14 (b-g) and Figure 15 (a-f) – while our proposed methods are rather robust against this variability observed in Figure 14 (h-j) and Figure 15 (g-j).

V. CONCLUSION

In this work, a novel local manifold learning methodology – RLMR – is developed for hyperspectral dimensionality reduction in order to tackle two challenges of LML, involving: 1) neighbor selection due to complex spectral variability (e.g., noise, illumination, non-uniform data distribution) and 2) the computation of affinity weights due to collinearity. The



(a) OSF

(b) PCA

(c) KPCA

(d) LLE

(e) LE

(f) LTSA

(g) JN

(h) HNS

(i) RLMR

Figure 15 Classification maps on the DFC dataset via SVM classifier using the different DR methods with the optimal parameters tuned as given in Table 5. (a)-(i) are the results using OSF, PCA, KPCA, LLE, LE, LTSA, JN, HNSP, RLMR, respectively.

proposed method is based on JN, RNS, and the integration of spatial information. It was validated via classification using two benchmark hyperspectral datasets. Compared to other state-of-the-art methods, the proposed method achieves better performance in terms of classification accuracy. RLMR has a more robust and stable performance than the other methods due to JN, RNS, and the embedding of spatial information, as shown in a series of experiments. In the future, we will further focus on how to more effectively embed the spatial information into dimensionality reduction framework. Additionally, the application of manifold learning methods to large-scale data should be given more attention in the future.

References

[1] Q. Zhang, R. Souvenir, R. Pless, “On manifold structure of cardiac mri data: Application to segmentation,” in IEEE Computer Society Conference on Computer Vison and Pattern Recognition (CVPR), 2006, pp. 1092-1098.

[2] J. Yang, D. Zhang, J. Yang, B. Niu, “Globally maximizing, locally minimization: unsupervised discriminant projection with application to face and palm biometrics,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 4, pp. 650-664, 2007.

[3] D.Lunga, S. Prasad, M. M. Crawford, O. Ersoy, “Manifold-learning-based feature extraction for classification of hyperspectral data: A review of advances in manifold learning, ” IEEE Signal Processing, vol. 31, no.1, pp. 55-66, Jan. 2014.

[4] D. Tosato, M. Farenzena, M. Spera, V. Murino, M. Cristani, “Multi-class classification on Riemannian manifolds for video surveillance, ” in Springer on Europe Conference on Computer vision (ECCV), 2010, pp. 378-391.

[5] L. K. Saul and S. T. Roweis, “Thing globally, fit locally: Unsupervised learning of low dimensional manifolds,” J. Mach. Learn. Res., vol. 4, pp.119-155, Jun. 2003.

[6] I. T. Jolliffe, Principle Component Analysis. New York, NY, USA: Springer-Verleg, 1986.

[7] J. B. Tenenbaum, V. de Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” Science, vol. 290, pp. 2319–2323, Dec. 2000.

[8] S. Roweis, L. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” Science, vol. 290, no. 5550, pp.2323-2326, 2000.

[9] M. Belkin and P. Niyogi, “Laplacian eigenmaps for dimensionality reduction and data representation,” Neural Comput., vol. 15, no. 6, pp.1373-1396, Mar. 2003.

[10] Z. Y. Zhang, and H. Y. L. Zha, “Principal manifolds and nonlinear dimension reduction via local tangent space alignment,” SIAM J. Sci. Comput., vol. 26, no. 1, pp. 313-338, Dec. 2004.

[11] C. M. Bachmann, T. L. Ainsworth, and R. A. Fusina, “Exploiting manifold geometry in hyperspectral imagery,” IEEE Trans. Geosci. Remote Sensing, vol. 43, no. 3, pp. 441–454, 2005.

[12] J. He, L. Zhang, Q. Wang, and Z. Li, “Using diffusion geometric coordinates for hyperspectral imagery representation, ” IEEE Trans. Geosci. Remote Sensing, vol. 6, no. 4, pp.767-771, Jan. 2009.

[13] L. Ma, M. M. Crawford, J. W. Tian, “Local manifold learning-based-k-nearest-neighbor for hyperspectral image classification,” IEEE Trans. Geosci. Remote Sensing, vol. 48, no. 11, pp.4099-4109, Nov. 2010.

[14] L. Ma, M. M. Crawfor, X. Yang, Y. Guo, “Local-manifold-leanring-based graph construction for semisupervised hyperspectral image classification,” IEEE Trans. Geosci. Remote Sensing, vol. 53, no. 5, pp.2832-2844, May. 2015.

[15] H. Huang, H. Huo, T. Fang, “Hierarchical manifold learning with application to supervised classification for high-resolution remotely sensed images,” IEEE Trans. Geosci. Remote Sensing, vol. 52, no. 3, pp.1677-1692, Mar. 2013.

[16] Y. Tan, H. Yuan, L. Li, “Manifold-based sparse representation for hyperspectral image classification,” IEEE Trans. Geosci. Remote Sensing, vol. 52, no. 12, pp. 7606-7618, Dec. 2014.

[17] L. Ma, M. M. Crawford, and J. W. Tian, “Anomaly detection for hyperspectral images based on robust locally linear embedding, ” J. Infrared Millimeter Terahertz Waves, vol.31, no.6, pp. 753-763, 2010.

[18] L. Zhang, L. Zhang, D. Tao, X. Huang, “Sparse transfer manifold embedding for hyperspectral target detection,” IEEE Trans. Geosci. Remote Sensing, vol. 52, no. 2, pp.1030-1042, Feb. 2014.

[19] H. L. Yang, M. M. Crawford, “Spectral and spatial proximity-based manifold alignment for multitemporal hyperspectral image classification,” IEEE Trans. Geosci. Remote Sensing, vol. 54, no. 1, pp.51-64, Jan. 2016.

[20] S. Yan, X. Dong, H. J. Zhang, Q. Yang, and S. Lin, “Graph embedding and extensions: A general framework for dimensionality reduction,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. `, pp. 40-51, Jan, 2007.

[21] S. Gerver, T. Tasdizen, R. Whitaker, “Robust non-linear dimensionality reduction using successive 1-dimensional Laplacian eigenmaps,” In Proceedings of the 24th international conference on machine learning (ICML), pp. 281-288, Jun 2007.

[22] H. Chang, D. Y. Yeung, “Robust locally linear embedding,” Pattern Recog., vol. 39, no. 6, pp. 1053-1065, Jun. 2006.

[23] Y. Goldberg, Y. A. Ritov, “LDR-LLE: LLE with low-dimensional neighborhood representation,” Advances in Visual Computing, vol. 5359, pp.43-54, Springer Berlin Heidelberg, Dec 2008.

[24] J. A. Lee, M. Verleysen, “Nonlinear dimensionality reduction,” Spring Science & Business Media, Oct. 2007.

[25] P. Zhang, H. Qiao, B. Zhang, “An improved local tangent space alignment method for manifold learning,” Pattern Lett., vol. 32, no. 2, pp. 181-189, Jan. 2011.

[26] Y. Zhan, J. Yin, “Robust local tangent space alignment,” Neural Information Processing, vol. 5863, pp.293-301, Springer Berlin Heidelberg, Dec. 2009.

[27] S. T. Monteiro, K. Uto, Y. Kosugi, K. Oda, Y. Lino, G. Saito, “Hyperspectral image classification of grass species in northeast Japan,” In Proc. IEEE Int. Symp. Geosci. Remote Sens., Jul. 2008, pp.IV-399.

[28] A. A. Nielsen, “The regularized iteratively reweighted MAD method for change detection in multi- and hyperspectral data,” IEEE Trans. Image Process, vol. 16, no. 2, pp. 463-478, Jan. 2007.

[29] S. Lyu, E. P. Simoncelli, “Nonlinear image representation using divisive normalization, ” in IEEE Computer Society Conference on Computer Vison and Pattern Recognition (CVPR), Jun. 2008, pp. 1-8.

[30] R. C. Gonzalez, R. E. Woods, “Digital image processing,” Prentice Hall, pp. 85-86, 2007.

[31] S.O. Los, P.R.J. North, W.M.F. Gery, M.J. Barnsley, “A method to convert AVHRR normalized difference vegetation index time series to a standard viewing and illumination geometry,” Remote Sens. Of Environment, vol. 99, no.4, pp. 400-411, Dec. 2005.

[32] D. Sage: Local normalization filter to reduce the effect of non-uniform illumination (March 2011): http://bigwww.epfl.ch/sage/soft/localnormalization/.

[33] S. Azadi, J. Maitin-Shepard, P. Abbeel, “Optimization-based artifact correction for electron microscopy image stacks,” in Springer on Europe Conference on Computer vision (ECCV), 2014, pp. 219-235.

[34] D. Wang, F. Nie, H. Huang, “Feature selection via global redundancy minimization,” IEEE Trans. Knowledge and Data Engineering, vol. 27, no. 10, Sep. 2015.

[35] Y. Pei, F. Huang, F. Shi, H. Zhi, “Unsupervised image matching based on manifold alignment,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 8, pp.1658-1664, Jun. 2012.

[36] C. Wang, J. Lai J. Zhu, ”Graph-based Multiprototype Competitive Learning and its applications”, IEEE transactions on Systems, Man, and Cybernetics-Part C, vol. 42, no. 6, pp.934-946, Dec. 2012.

[37] M. Jia, M. Gong, E. Zhang, Y. Li, L. Jiao, “Hyperspectral image classification based on nonlocal means with a novel class-relativity measurement, ” IEEE Geosci. Remote Sensing Lett., vol. 11, no. 7, Jul. 2014.

[38] Y. Chen, N. M. Nasrabadi, T. D. Tran, “Hyperspectral image classification using dictionary-based sparse representation, ” IEEE Trans. Geosci. Remote Sensing, vol. 49, no. 10, pp. 3973-3985, Otc. 2011.

[39] L. Zhang, M. Yang, X. Feng, “Sparse representation or collaborative representation: which helps face recognition?,” In IEEE International Conference on Computer Vision (ICCV), Nov. 2011, pp. 471-478.

[40] P. Ghamisi, J. A. Benediktsson, M.O. Ulfarsson, “Spectral-Spatial classification of hyperspectral images based on hidden markov random fields, ” IEEE Trans. Geosci. Remote Sensing, vol. 52 no. 5, pp. 2565-2574, Feb. 2014.

[41] B. Scholkopf, A. J. Smola, K. R. Muller, “Nonlinear component analysis as a kernel eigenvalue problem,” Neural. Comput., vol. 10, no. 5, pp. 583-588, 1998.

[42] Y. Bengio, O. Delalleau, N. Le Roux, “Learning eigenfunctions links spectral embedding and kernel PCA,” Neural Comput., vol. 16, no.10, pp. 2197-2219, 2004.

[43] Y. Bengio, J. F. Paiement, P. Vincent. “Out-of-sample extensions for LLE, Isomap, MDS, eigenmaps, and spectral clustering,” Advances in Neural Information Processing Systems (NIPS), Cambridge, MA, 2003, pp. 177-184.

[44] J. A. Benediktsson and P. Ghamisi, Spectral-Spatial Classification of Hyperspectral Remote Sensing Images, Artech House Publishers, INC, Boston, USA.