

Incremental Dictionary Learning-Driven Tensor Low-Rank and Sparse Representation for Hyperspectral Image Classification

Zhaohui Xue[✉], Member, IEEE, Xiangyu Nie, and Mengxue Zhang[✉]

Abstract— Low-rank and sparse representation (LRSR) has gained popularity in hyperspectral image (HSI) classification. However, existing LRSR models usually treat HSI as a 2-D matrix, which may destroy the original 3-D intrinsic structure of HSI. Moreover, the dictionary consisting of only training samples lacks completeness and may be suboptimal for representation. To overcome the above issues, we propose an incremental dictionary learning-driven tensor LRSR (TLRSR-IDL) model for HSI classification. First, we represent HSI as a third-order tensor to retain its original 3-D intrinsic structure by using the TLRSR model, which also combines both sparsity and low rankness to maintain global and local data structures. Second, we design an optimal reconstruction within regularized neighborhood (ORRN) method to exploit spectral–spatial information by avoiding the interference of heterogeneous samples in the neighborhood. Finally, an incremental dictionary learning (IDL) scheme is designed to iteratively introduce augmented samples into the dictionary, and the final classification map is produced by feeding back the last round of the incremental dictionary into the TLRSR-IDL model. The main innovative contribution lies in that the proposed IDL scheme can leverage supervised and unsupervised information, which greatly enhances traditional LRSR and TLRSR models. Experimental results based on three popular hyperspectral datasets demonstrate that the proposed method outperforms other related counterparts in terms of classification accuracy and generalization performance, with overall accuracy (OA) improvements of 0.97%–16.83%, 1.25%–6.89%, and 0.85%–6.67% for Indian Pines, Pavia University, and Salinas, respectively.

Index Terms— Hyperspectral image (HSI), incremental dictionary learning (IDL), spectral–spatial classification, tensor low-rank and sparse representation (TLRSR).

NOMENCLATURE

\mathcal{H}	Hyperspectral dataset.
\mathcal{H}'	Denoised \mathcal{H} .
\mathcal{D}	Initial dictionary.
$\mathcal{D}^{(k)}$	k th incremental dictionary.

Manuscript received 30 June 2022; revised 1 November 2022; accepted 17 November 2022. Date of publication 21 November 2022; date of current version 1 December 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 41971279 and Grant 42271324 and in part by the Natural Science Foundation of Jiangsu Province under Grant BK20221506. (*Corresponding author: Zhaohui Xue.*)

Zhaohui Xue and Xiangyu Nie are with the School of Earth Sciences and Engineering, Hohai University, Nanjing 211100, China, and also with the Jiangsu Province Engineering Research Center of Water Resources and Environment Assessment Using Remote Sensing, Hohai University, Nanjing 211100, China (e-mail: zhaohui.xue@hhu.edu.cn).

Mengxue Zhang is with the Image and Signal Processing (ISP) Group, University of València, 46980 València, Spain.

Digital Object Identifier 10.1109/TGRS.2022.3223792

\mathcal{X}	Coefficient item corresponding to \mathcal{D} .
$\mathcal{X}^{(k)}$	Coefficient item corresponding to $\mathcal{D}^{(k)}$.
\mathcal{E}	Noise item corresponding to \mathcal{D} .
$\mathcal{E}^{(k)}$	Noise item corresponding to $\mathcal{D}^{(k)}$.
$W_S = T \times T$	Window size is utilized in ORRN.
$W_S^{(k)} = T^{(k)} \times T^{(k)}$	Window size is utilized in the k th IDL.
\mathcal{Y}_i	i th neighborhood sample set based on W_S .
$\mathcal{Y}_i^{(k)}$	i th neighborhood sample set based on $W_S^{(k)}$.
α	Sparse regularization parameter.
β	Noise regularization parameter.
S	Similarity threshold.
C	Number of classes.
I_1, I_2, I_3	Number of bands, columns, and rows, respectively.
N	Number of test samples.
K	Maximum number of iterations.

I. INTRODUCTION

HYPERSPECTRAL remote sensing is capable of providing hundreds of continuous wavelengths to represent land surface objects with extremely refined spectra, making it possible to identify and distinguish various objects accurately [1]. In this context, hyperspectral image (HSI) classification has gained great popularity and played an important role in precision agriculture, forestry monitoring, mineral surveys, environmental assessment, and so on [2].

In the past few decades, HSI classification has been greatly promoted by advanced machine learning and pattern recognition methods [3], [4], [5], [6]. In particular, representation-based methods [7] have attracted great interest due to the emerging paradigm of compressed sensing. Typical prototypes of representation-based classification methods include sparse representation (SR) [8], collaborative representation (CR) [9], and low-rank representation (LRR) [10]. As a seminal work, Chen et al. [8] introduced SR into the HSI classification scenario for the first time, and they also proposed a joint SR (JSR) method to incorporate spectral–spatial information. Then,

kernel SR was proposed to enhance the HSI classification performance [11]. Later, further explorations based on SR for HSI classification included class-dependent SR [12], morphological component analysis [13], graph embedding [14], discriminative SR [15], and graph regularization [16], [17]. Recent advanced progress has focused on SR incorporated with self-paced learning [18], nonlocal weighting [19], maximum likelihood estimation [20], manifold-based sparse discriminant learning [21], and multifeature/multiscale learning [22], [23], [24]. Although SR-based methods perform well in classification, such methods have a common drawback of ignoring the global data structure.

Different from SR, LRR seeks the lowest rank representation of the given data through a complete dictionary or the given data, and LRR-based methods have been widely used in HSI processing [25], [26], [27], [28]. To effectively mitigate the “Hughes phenomenon” of HSI classification, Zhai et al. [29] utilized a novel Laplacian-regularized LRR method for band selection. Similarly, An et al. [30] proposed a new dimensionality reduction method where LRR was utilized to maintain the global structure and tensor representation was utilized to preserve the spatial neighborhood information. Since ignoring the semantic information around pixels in the LRR method may lead to the “salt and pepper” problem in HSI classification, Wang et al. [31] proposed a self-supervised LRR method combined with superpixel segmentation to effectively alleviate this problem. To overcome the “ill-posed” classification issue with a small training set, Feng et al. [32] proposed a geometric low-rank Laplacian-regularized semisupervised classifier.

LRR-based methods lack an effective description of the local data structure, and a more advocated model is low-rank and sparse representation (LRSR) [33]. In LRSR, sparsity and low rankness are combined to maintain global and local data structures simultaneously [34]. Substantial studies based on LRSR have been exploited for HSI spectral–spatial classification [35], [36], [37], graph-based discriminant analysis [38], [39], fusion of hyperspectral and LiDAR/multispectral data [40], band selection [41], superresolution [42], [43], and anomaly detection [44], [45]. Nevertheless, traditional LRSR methods need to expand HSI into a 2-D matrix and then find the optimal solution, which not only seriously destroys the original 3-D intrinsic structure of HSI but also loses part of the discriminative information.

Recently, to overcome the drawbacks of LRSR, tensor-based LRSR (TLRSR) methods have attracted much more attention [46], [47], [48]. In order to improve the spatial resolution of HSI, Dian et al. [49] proposed a low tensor-train rank representation (LTTR) method for HSI super-resolution. Based on the high-spatial-resolution multispectral image (MSI) and low-spatial-resolution HSI, LTTR can learn the correlations among the spatial, spectral, and nonlocal modes of the nonlocal similar high-spatial-resolution HSI cubes. Later, in [50], a novel unified global and local fusion framework was presented for HSI and MSI fusion, called graph Laplacian-guided coupled tensor decomposition (gLGCTD). The gLGCTD method can preserve the intrinsic spatial–spectral information by coupled Tucker decomposition and exploits the latent submanifold

structures encoded by graph Laplacian regularization. For removing the noise in HSI, Xue et al. [51] designed a nonlocal low-rank regularized CANDECOMP/PARAFAC tensor decomposition (NLR-CPTD) method based on global correlation across the spectrum and nonlocal self-similarity characteristics of HSI, which has the ability to avoid tensor rank estimation bias and improve denoising performance. In [52], a general 3-D $\ell_{2,1}$ spatial–spectral total variation ($\ell_{2,1}$ -SSTV) regularization was first developed to globally represent the sparse prior in the gradient domain of HSI and then combined with the local low-rank tensor (LTR) model to enhance spatial information and denoising. Furthermore, Zheng et al. [53] proposed a sparse and low-rank constrained tensor factorization (SPLRTF) algorithm for HSI unmixing, which is based on the matrix–vector nonnegative tensor factorization (MV-NTF) framework. The main idea of SPLRTF is to flexibly express the low-rank characteristics of the abundance tensors by low-rank constraint and combine sparse constraint to ensure the accuracy of abundance maps. However, the above methods are mainly developed based on tensor robust principal component analysis (T-RPCA), and they cannot be used to cluster tensor data [54]. Therefore, it is difficult to directly apply them to classification tasks. Although TLRSR models based on subdictionary reconstruction has the ability to cluster data, the involved dictionary is usually built from the training samples, which are fixed and may be suboptimal for representation.

To address the above issues, this article presents a novel incremental dictionary learning-driven TLRSR (TLRSR-IDL) model for HSI classification, where HSI is regarded as a third-order tensor to retain the original 3-D intrinsic structure while maintaining global and local data structures through low-rank and sparse constraints. Moreover, an optimal reconstruction within regularized neighborhood (ORRN) method is designed to exploit spectral–spatial information. In the ORRN, we combine Euclidean and cosine distance metrics to perform neighborhood regularization on the dynamic window-based sample set and determine the label of the central test sample by minimizing the reconstruction error within the regularized neighborhood. More importantly, we propose an incremental dictionary learning (IDL) scheme to enhance the representation ability of the initial dictionary by leveraging supervised and unsupervised information in an iterative learning fashion, where a regularized neighborhood with decreasing size is constructed and further utilized to find augmented samples and continuously augment the dictionary.

The main contributions of our work can be summarized as follows.

- 1) We propose a TLRSR-IDL model for HSI classification, which can maintain the original 3-D intrinsic structure as well as the global and local data structure. To the best of our knowledge, it is unique in the literature.
- 2) We design an ORRN method for exploiting spectral–spatial information, which can effectively introduce contextual information and avoid the interference of heterogeneous samples on the classification performance. In addition, the dynamic window strategy can

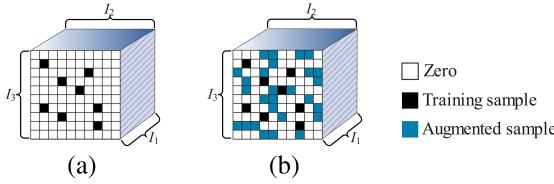


Fig. 1. Graphical illustration of the dictionary. (a) Initial dictionary. (b) Incremental dictionary.

continuously enhance the consistency of the neighborhood sample set.

- 3) We present an IDL scheme to iteratively enhance the completeness and representation ability of the initial dictionary. The proposed IDL scheme can leverage supervised and unsupervised information, which greatly enhances traditional LRSR and TLRSR models.

The main differences between the proposed TLRSR-IDL method and other LRSR or TLRSR methods include the following.

- 1) Traditional LRSR models [35], [36], [37] are based on 2D matrix operations by expanding HSI into a 2-D matrix, which destroys the inherent 3-D spectral-spatial structure of HSI. On the contrary, the proposed method TLRSR-IDL treats HSI as a 3-D data cube in the whole process, which effectively preserves the intrinsic structure and spectral-spatial information of HSI.
- 2) Existing TLRSR models have been widely used in super-resolution [49], [50], denoising [51], and unmixing [53] of HSI, but they are rarely used for classification since such models cannot cluster tensor data, and they often need to combine with additional classifiers. Differently, TLRSR-IDL can directly determine the class of unlabeled samples according to the designed ORRN.
- 3) The dictionary in current LRSR- or TLRSR-related models is usually fixed, which is incompleteness and suboptimal for representation. In contrast, we present an IDL strategy to combine supervised and unsupervised information in an iterative learning fashion to continuously expand the dictionary.

II. PROPOSED METHOD

Let $\mathcal{H} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ be a hyperspectral dataset, where I_1 represents the number of bands and I_2 and I_3 represent the number of columns and rows, respectively. Let $\mathcal{D} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ indicate the initial dictionary built by training samples, and the remaining positions are filled with zeros. Let $\mathcal{D}^{(k)} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ represent the k th incremental dictionary, where $k \in [1, K]$ and K is the maximum number of iterations. \mathcal{D} and $\mathcal{D}^{(k)}$ are constructed, as shown in Fig. 1. Let $W_S = T \times T$ (where $T \in \{3, 5, \dots\}$) denote the neighborhood window size utilized to determine the class label in the ORRN, and let $W_S^{(k)} = T^{(k)} \times T^{(k)}$ denote the neighborhood window size used to find augmented samples in the k th IDL, where $T^{(k)} = \max\{T + 2[\text{ceil}(K/2)] - k, 3\}$ and $\text{ceil}(\cdot)$ is the round-up function. The neighborhood sample set of the i th test sample is denoted as $\mathcal{Y}_i = \{y_{i1}, y_{i2}, \dots, y_{iW_S}\}$, where

$i \in [1, N]$ with N the number of test samples, y_{i1} is the test sample located in the center of the window, and the remaining neighborhood samples are sorted in column-first order (e.g., y_{i2} and y_{i3} are the first two samples of the first column). Let $\mathcal{Y}_i^{(k)} = \{y_{i1}^{(k)}, y_{i2}^{(k)}, \dots, y_{iW_S^{(k)}}^{(k)}\}$ represent the neighborhood sample set of the i th test sample constructed according to $W_S^{(k)}$. For clarity, the above and other important notations are reported in the Nomenclature. Furthermore, some necessary tensor operation definitions and lemmas utilized to solve the TLRSR-IDL model are illustrated in Appendix A.

A graphical illustration of the proposed TLRSR-IDL model for HSI classification is shown in Fig. 2. First, we solve the TLRSR (TLRSR-IDL, when $k > 1$) model to represent hyperspectral data, where HSI is regarded as a third-order tensor, and we restore HSI based on each subdictionary belonging to different classes. Second, we construct the neighborhood sample set based on a dynamic window and filter out the heterogeneous samples through neighborhood regularization. The label of the central test sample is then determined by minimizing the reconstruction error within the regularized neighborhood. Finally, we iteratively update the dictionary based on the IDL scheme to explore augmented samples, with the final classification map produced by feeding the last incremental dictionary round into the model.

A. TLRSR-IDL Model Definition

In the TLRSR model, the objective function is formulated as

$$\min_{\mathcal{X}} \text{rank}(\mathcal{X}) + \alpha \|\mathcal{X}\|_0, \quad \text{s.t. } \mathcal{H} = \mathcal{D} * \mathcal{X} \quad (1)$$

where α is a positive regularization parameter balancing sparsity and low rankness and $\text{rank}(\cdot)$ means low-rank operation. Tensor ℓ_0 -norm ($\|\cdot\|_0$) represents the number of all nonzero elements in the input tensor.

Due to low-rank operations and the ℓ_0 -norm, (1) is nonconvex and difficult to optimize. Therefore, we use tensor nuclear norm ($\|\cdot\|_*$) (see Definition 11) and tensor ℓ_1 -norm ($\|\cdot\|_1$) for convex optimization, which can be reformulated as

$$\min_{\mathcal{X}} \|\mathcal{X}\|_* + \alpha \|\mathcal{X}\|_1, \quad \text{s.t. } \mathcal{H} = \mathcal{D} * \mathcal{X}. \quad (2)$$

HSI is usually noisy in real applications. Therefore, a robust TLRSR model considering noise reduction takes the form

$$\min_{\mathcal{X}} \|\mathcal{X}\|_* + \alpha \|\mathcal{X}\|_1 + \beta \|\mathcal{E}\|_{2,1}, \quad \text{s.t. } \mathcal{H} = \mathcal{D} * \mathcal{X} + \mathcal{E} \quad (3)$$

where β is a positive regularization parameter utilized to balance the noise term, $\|\cdot\|_{2,1}$ represents the tensor $\ell_{2,1}$ -norm, and $\mathcal{E} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is the noise term. Note that we impose the tensor ℓ_1 -norm constraint to maintain the local structural information of the data. In addition, a noise term constrained by the tensor $\ell_{2,1}$ -norm is utilized to remove the noise of HSI and improve the robustness of the model.

After the k th iteration, the current dictionary is replaced by the new incremental dictionary $\mathcal{D}^{(k)}$, and the objective function in TLRSR-IDL is

$$\begin{aligned} \min_{\mathcal{X}^{(k)}, \mathcal{E}^{(k)}} & \|\mathcal{X}^{(k)}\|_* + \alpha \|\mathcal{X}^{(k)}\|_1 + \beta \|\mathcal{E}^{(k)}\|_{2,1} \\ \text{s.t. } & \mathcal{H} = \mathcal{D}^{(k)} * \mathcal{X}^{(k)} + \mathcal{E}^{(k)}. \end{aligned} \quad (4)$$

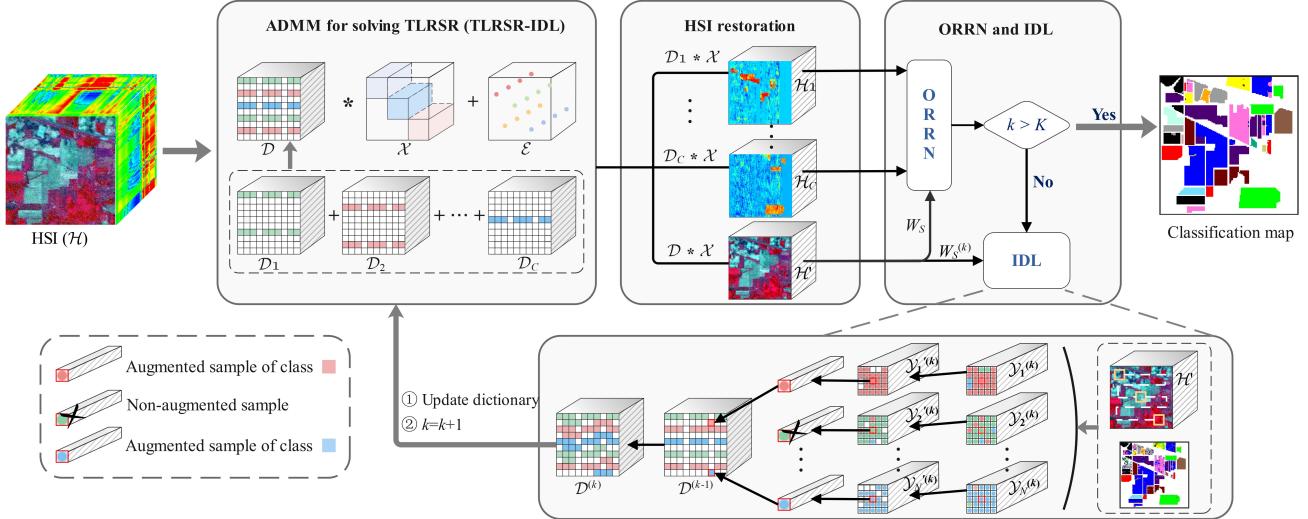


Fig. 2. Graphical illustration of the proposed TLRSR-IDL model for HSI classification.

B. TLRSR-IDL for HSI Classification

1) *HSI Restoration Based on the Subdictionary*: We first randomly select a certain number of labeled samples from each class of \mathcal{H} to build the initial dictionary \mathcal{D} . Then, the coefficient term \mathcal{X} and the noise term \mathcal{E} can be obtained by solving (3). Since $\mathcal{D} = \mathcal{D}_1 + \mathcal{D}_2 + \dots + \mathcal{D}_C$, equation $\mathcal{H} = \mathcal{D} * \mathcal{X} + \mathcal{E}$ can be expanded as

$$\mathcal{H} = (\mathcal{D}_1 + \mathcal{D}_2 + \dots + \mathcal{D}_C) * \mathcal{X} + \mathcal{E} \quad (5)$$

where C is the number of classes. Let $\mathcal{D}_c \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ ($c \in [1, C]$) represent the subdictionary containing only dictionary samples of class c , i.e., the number of subdictionaries is equal to the number of classes, and \mathcal{H}_c can be restored based on \mathcal{D}_c as

$$\mathcal{H}_c = \mathcal{D}_c * \mathcal{X}. \quad (6)$$

Let \mathcal{H}' represent the denoised HSI, which can be obtained by

$$\mathcal{H}' = \mathcal{D} * \mathcal{X} \quad \text{or} \quad \mathcal{H}' = \mathcal{H} - \mathcal{E}. \quad (7)$$

2) *Optimal Reconstruction Within Regularized Neighborhood*: To make full use of the neighborhood information, the neighborhood sample set \mathcal{Y}_i can be established using the i th test sample y_{i1} as the central sample. During follow-up iterations, the size of W_S is dynamically updated as

$$W_S = \begin{cases} W_S, & W_S < W_S^{(k)} \\ W_S^{(k)}, & \text{otherwise.} \end{cases} \quad (8)$$

Specifically, we take $T = 7$ and $K = 5$ (i.e., $k \in [1, 5]$, $T^k = 13 - 2k$) as an example, and the size of W_S under different iterations is shown in Fig. 3(a). As can be seen, the size of W_S starts to decrease when $k > 3$. The reason for this setting is that the completeness and representation ability of the dictionary improves after several rounds of incremental learning. In this context, we enhance the consistency of samples in the sample set by reducing the size of W_S to

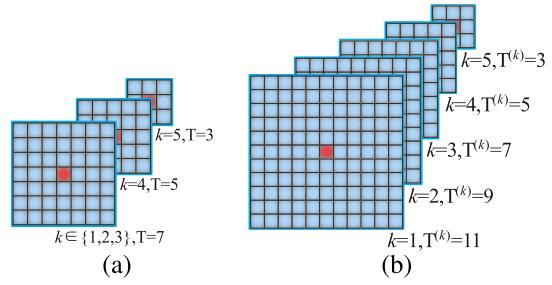


Fig. 3. Dynamic window with sizes of (a) W_S and (b) $W_S^{(k)}$ at different iterations (corresponding to the settings of $T=7$ and $K=5$).

ensure that better performance can be achieved in subsequent classification.

However, the existence of heterogeneous samples cannot be avoided within a square neighborhood window, which will affect the classification accuracy. Inspired by our previous work [55], we design a similarity measure that combines the Euclidean distance and cosine distance for neighborhood regularization, which takes the form

$$S_{ij} = \frac{\langle y_{i1}, y_{ij} \rangle}{\|y_{i1}\|_1 \times \|y_{ij}\|_1} \exp \left\{ - \left\| \frac{y_{i1}}{\|y_{i1}\|_1} - \frac{y_{ij}}{\|y_{ij}\|_1} \right\|_2 \right\} \quad (9)$$

where $j = 2, 3, \dots, W_S$ and $\exp(\cdot)$ represents the exponential function. Then, S_{ij} is designed to measure the similarity between y_{i1} and y_{ij} , and a larger value of S_{ij} indicates a higher similarity. Assume that the optimal threshold of similarity is S , and we remove y_{ij} from \mathcal{Y}_i when $S_{ij} < S$. After neighborhood regularization, the remaining samples are resorted as $\mathcal{Y}'_i = \{y'_{i1}, y'_{i2}, \dots, y'_{iW'_S}\}$, where $y'_{i1} = y_{i1}$ and $W'_S \leq W_S$.

Denote the minimum reconstruction error between \mathcal{Y}'_i and \mathcal{Y}'_i^c as r_i^c , where \mathcal{Y}'_i^c represents the neighborhood sample set based on W_S and \mathcal{H}_c . Then, r_i^c can be formulated as

$$r_i^c = \min \left\{ \left\| \frac{y_{i1}^c}{\|y_{i1}^c\|_1} - \frac{y'_{i1}}{\|y'_{i1}\|_1} \right\|_1, \left\| \frac{y_{i2}^c}{\|y_{i2}^c\|_1} - \frac{y'_{i2}}{\|y'_{i2}\|_1} \right\|_1, \dots, \left\| \frac{y_{iW'_S}^c}{\|y_{iW'_S}^c\|_1} - \frac{y'_{iW'_S}}{\|y'_{iW'_S}\|_1} \right\|_1 \right\} \quad (10)$$

where y_{ij}^c and y'_{ij} ($j \in [1, W_s^i]$) are located in the same spatial position on \mathcal{Y}_i^c and \mathcal{Y}'_i .

Subsequently, the class label of the i th test sample can be obtained by comparing the value of r_i^c ($c \in 1, 2, \dots, C$)

$$\text{class}(y'_{i1}) = \arg \min_{c=1,2,\dots,C} r_i^c. \quad (11)$$

3) *Incremental Dictionary Learning*: The neighborhood sample set $\mathcal{Y}_i^{(1)}$ can be constructed based on \mathcal{H}' and $W_S^{(1)}$. Different from \mathcal{Y}_i , $\mathcal{Y}_i^{(1)}$ is utilized to determine the augmented sample. Similarly, we need to perform the neighborhood regularization operation on $\mathcal{Y}_i^{(1)}$, which is regularized by (9) and resorted as $\mathcal{Y}_i'^{(1)} = \{y_{i1}^{(1)}, y_{i2}^{(1)}, \dots, y_{iW_S^{(1)}}^{(1)}\}$. If all samples in $\mathcal{Y}_i'^{(1)}$ satisfy (12), the central sample $y_{i1}^{(1)}$ is regarded as an augmented sample whose class label can be determined as $\text{class}(y_{i1}^{(1)})$

$$\text{class}\left(y_{i1}^{(1)}\right) = \text{class}\left(y_{i2}^{(1)}\right) \cdots = \text{class}\left(y_{iW_S^{(1)}}^{(1)}\right). \quad (12)$$

After obtaining an augmented sample, we add it to the initial dictionary \mathcal{D} according to its spatial position and traverse all test samples to obtain the first round of incremental dictionary $\mathcal{D}^{(1)}$. Next, we further improve the completeness and representation ability of the incremental dictionary through iterative learning. Taking the k th iteration as an example, the complete IDL fashion can be summarized as follows.

- 1) The TLRSR-IDL model in (4) is solved according to $\mathcal{D}^{(k-1)}$. The restored image $\mathcal{H}_c^{(k-1)}$ and the denoised image $\mathcal{H}'^{(k-1)}$ can be obtained by (5)–(7).
- 2) W_S is updated by (8), and the neighborhood sample set \mathcal{Y}_i is constructed from W_S and $\mathcal{H}'^{(k-1)}$. Furthermore, \mathcal{Y}_i regularized by (9) is resorted as \mathcal{Y}'_i .
- 3) The class labels are reassigned to the test samples according to (10) and (11).
- 4) The neighborhood sample set $\mathcal{Y}_i^{(k)}$ is regularized by (9) and resorted as $\mathcal{Y}_i'^{(k)}$, where $\mathcal{Y}_i'^{(k)}$ is constructed from $W_S^{(k)}$ and $\mathcal{H}'^{(k-1)}$. Then, augmented samples are obtained by (12), and a new incremental dictionary $\mathcal{D}^{(k)}$ can be obtained.

In summary, the pseudocode of IDL is outlined in Algorithm 1.

In the IDL scheme, we combine both supervised and unsupervised information to learn augmented samples. In detail, for the i th neighborhood sample set $\mathcal{Y}_i^{(k)}$, the supervised information refers to the class labels obtained in the ORRN step, which may not be consistent within a fixed window. The unsupervised information refers to the result $\mathcal{Y}_i'^{(k)}$ obtained by the neighborhood regularization of $\mathcal{Y}_i^{(k)}$. The essence of neighborhood regularization is to divide the samples in $\mathcal{Y}_i^{(k)}$ into two classes. The first class is the samples with high similarity to the central sample, which are retained. The second class is the samples with low similarity to the central sample, which are removed. At this point, if all the retained samples have the same class label, it means that the supervised and unsupervised results are in agreement. Then, the central sample will be introduced into the dictionary as an augmented sample.

As shown in Fig. 3(b), in the iterative process of IDL, the size of $W_S^{(k)}$ decreases as the number of iterations k

Algorithm 1 IDL

Input: \mathcal{H} , $\mathcal{D}^{(k-1)}$, T , and iteration k

1. Solve the model in Eq. (4) based on $\mathcal{D}^{(k-1)}$.
2. Calculate $\mathcal{H}_c^{(k-1)}$ and $\mathcal{H}'^{(k-1)}$ by Eqs. (5)–(7).
3. Update W_S by Eq. (8) and regularize \mathcal{Y}_i by Eq. (9).
4. Obtain class labels by Eqs. (10)–(11).
5. Regularize $\mathcal{Y}_i'^{(k)}$ by Eq. (9).
6. Predict augmented samples by Eq. (12).

Output: $\mathcal{D}^{(k)}$

increases. The reason for this design is that in the initial IDL process, the classification accuracy is poor since the current dictionary is suboptimal. Therefore, using a larger neighborhood size can probably avoid introducing incorrect samples. Then, the classification results can be improved as the dictionary is augmented. In this case, more reliable samples can be augmented by shrinking the neighborhood size, and the completeness and representation ability of the dictionary can be gradually enhanced.

C. Alternating Direction Method of Multipliers for TLRSR-IDL Model Solving

We utilize the alternating direction method of multipliers (ADMM) [56] to solve the TLRSR-IDL model in (4). Moreover, $\mathcal{D}^{(k)}$ is replaced with \mathcal{D} when solving the TLRSR model in (3).

To make the low-rank term and the sparse term separable, we introduce two new supplementary variables \mathcal{X}_1 and \mathcal{X}_2 ; then, the objective function in (4) is converted to equal optimization problems as

$$\begin{aligned} & \min_{\mathcal{X}^{(k)}, \mathcal{E}^{(k)}} \|\mathcal{X}_1\|_* + \alpha \|\mathcal{X}_2\|_1 + \beta \|\mathcal{E}^{(k)}\|_{2,1} \\ & \text{s.t. } \mathcal{X}_1 = \mathcal{X}^{(k)}, \quad \mathcal{X}_2 = \mathcal{X}^{(k)}, \quad \mathcal{H} = \mathcal{D}^{(k)} * \mathcal{X}^{(k)} + \mathcal{E}^{(k)}. \end{aligned} \quad (13)$$

The corresponding augmented Lagrange function of (13) is given by

$$\begin{aligned} & L(\mathcal{X}^{(k)}, \mathcal{E}^{(k)}, \mathcal{X}_1, \mathcal{X}_2, \mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3) \\ & = \|\mathcal{X}_1\|_* + \alpha \|\mathcal{X}_2\|_1 + \beta \|\mathcal{E}^{(k)}\|_{2,1} \\ & \quad + \langle \mathcal{Q}_1, \mathcal{H} - \mathcal{D}^{(k)} * \mathcal{X}^{(k)} - \mathcal{E}^{(k)} \rangle \\ & \quad + \langle \mathcal{Q}_2, \mathcal{X}_1 - \mathcal{X}^{(k)} \rangle + \langle \mathcal{Q}_3, \mathcal{X}_2 - \mathcal{X}^{(k)} \rangle \\ & \quad + \frac{\mu}{2} \left(\|\mathcal{H} - \mathcal{D}^{(k)} * \mathcal{X}^{(k)} - \mathcal{E}^{(k)}\|_F^2 \right. \\ & \quad \left. + \|\mathcal{X}_1 - \mathcal{X}^{(k)}\|_F^2 + \|\mathcal{X}_2 - \mathcal{X}^{(k)}\|_F^2 \right) \end{aligned} \quad (14)$$

where \mathcal{Q}_1 , \mathcal{Q}_2 , and \mathcal{Q}_3 are the Lagrange multipliers and μ is a positive penalty parameter. For tensors \mathcal{A} and \mathcal{B} in $\mathbb{R}^{I_1 \times I_2 \times I_3}$, $\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i=1}^{I_3} \langle \mathcal{A}^{(i)}, \mathcal{B}^{(i)} \rangle$ represents the tensor inner product.

The ADMM method is utilized to solve (14), and the detailed updating rules are presented in Appendix B. The pseudocode of using the ADMM method to solve TLRSR-IDL is outlined in Algorithm 2.

Algorithm 2 ADMM for Solving TLRSR-IDL

Input: \mathcal{H} , $\mathcal{D}^{(k)}$, parameters α and β , max-iter
Initialize: $\mathcal{X}^{(k)} = \mathcal{X}_1 = \mathcal{X}_2 = 0$, $\mathcal{E}^{(k)} = 0$, $\mathcal{Q}_1 = \mathcal{Q}_2 = \mathcal{Q}_3 = 0$, $\mu = 10^{-4}$, $\mu_{max} = 10^2$
While not max-iter **do**
 Update \mathcal{X}_1 by Eqs. (24)-(27)
 Update $\mathcal{X}^{(k)}$ by Eqs. (28)-(30)
 Update \mathcal{X}_2 by Eq. (31)
 Update $\mathcal{E}^{(k)}$ by Eqs. (32)-(33)
 Update $\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3$ and μ
 $\mathcal{Q}_1 = \mathcal{Q}_1 + \mu(\mathcal{H} - \mathcal{D}^{(k)} * \mathcal{X}^{(k)} - \mathcal{E}^{(k)})$
 $\mathcal{Q}_2 = \mathcal{Q}_2 + \mu(\mathcal{X}_1 - \mathcal{X}^{(k)})$
 $\mathcal{Q}_3 = \mathcal{Q}_3 + \mu(\mathcal{X}_2 - \mathcal{X}^{(k)})$
 $\mu = \min(1.2\mu, \mu_{max})$
Output: $\mathcal{X}^{(k)}, \mathcal{E}^{(k)}$

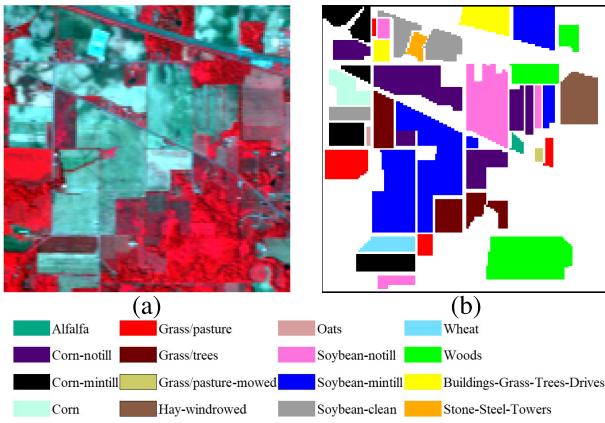


Fig. 4. Indian Pines datasets. (a) False-color composite image (R: 50, G: 27, and B: 17). (b) Ground-truth map.

III. EXPERIMENTAL RESULTS

A. Hyperspectral Datasets

1) *Indian Pines*: The dataset was gathered by an Airborne Visible Infrared Imaging Spectrometer (AVIRIS) sensor over Northwestern Indiana's Indian Pines test site in 1992. The dataset consists of 145×145 pixels and 224 spectral reflectance bands in the wavelength range from 0.4 to $2.5 \mu\text{m}$ with a spatial resolution of 20 m. There are 200 radiance channels used in the experiments after removing 20 noisy and water-absorbed bands. The false-color composite and the ground-truth maps are shown in Fig. 4. The ground-truth map includes 16 classes. The numbers of training and test samples utilized in the experiment are listed in Table I.

2) *Pavia University*: The dataset was acquired by the Reflective Optics Spectrographic Imagine System (ROSIS) sensor over the urban area of the University of Pavia, Northern Italy, in 2002. The dataset consists of 610×340 pixels and 115 spectral reflectance bands in the wavelength range from 0.43 to $0.86 \mu\text{m}$ with a spatial resolution of 1.3 m. There are 103 radiance channels used in the experiments after removing 12 noisy bands. The false-color composite and the ground-truth maps are shown in Fig. 5. The ground-truth map includes nine classes. The numbers of training and test samples utilized in the experiment are listed in Table II.

3) *Salinas*: The dataset was obtained by an AVIRIS sensor over Salinas Valley, CA, USA. The dataset consists of

TABLE I
NUMBER OF TRAINING AND TEST SAMPLES USED FOR INDIAN PINES

Class	Name	Training	Test	Total
1	Alfalfa	5	41	46
2	Corn-notill	143	1,285	1,428
3	Corn-mintill	83	747	830
4	Corn	24	213	237
5	Grass/pasture	49	434	483
6	Grass/trees	73	657	730
7	Grass/pasture-mowed	3	25	28
8	Hay-windrowed	48	430	478
9	Oats	2	18	20
10	Soybean-notill	98	874	972
11	Soybean-mintill	246	2,209	2,455
12	Soybean-clean	60	533	593
13	Wheat	21	184	205
14	Woods	127	1,138	1,265
15	Buildings-Grass-Trees-Drives	39	347	386
16	Stone-Steel-Towers	10	83	93
		Total	1,031	9,218
				1,0249

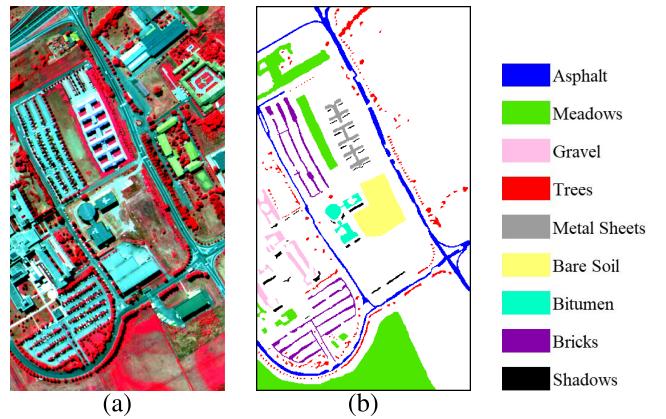


Fig. 5. Pavia University datasets. (a) False-color composite image (R: 102, G: 56, and B: 31). (b) Ground-truth map.

TABLE II
NUMBER OF TRAINING AND TEST SAMPLES USED FOR PAVIA UNIVERSITY

Class	Name	Training	Test	Total
1	Asphalt	332	6,299	6,631
2	Meadows	933	17,716	18,649
3	Gravel	105	1,994	2,099
4	Trees	153	2,911	3,064
5	Metal Sheets	68	1,277	1,345
6	Bare Soil	252	4,777	5,029
7	Bitumen	67	1,263	1,330
8	Bricks	185	3,497	3,682
9	Shadows	48	899	947
		Total	2,143	40,633
				42,776

512 \times 217 pixels and 224 spectral reflectance bands in the wavelength range from 0.4 to $2.5 \mu\text{m}$ with a spatial resolution of 3.7 m. There are 204 spectral bands retained after discarding the 20 water absorption bands. The false-color composite and the ground-truth maps are shown in Fig. 6. The ground-truth map includes 16 classes. The numbers of training and test samples utilized in the experiment are listed in Table III.

B. Experimental Settings

In this section, we introduce the comparison methods, corresponding parameter settings, quantitative metrics, and experimental environment.

- To verify the performance of the proposed method, we select eight methods for comparison, including support vector machines (SVMs) [57], SVM with composite

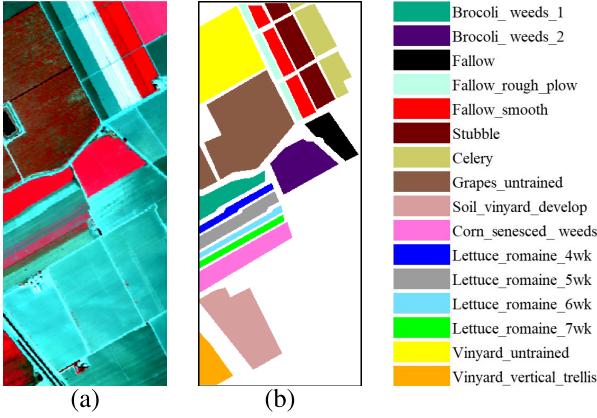


Fig. 6. Salinas datasets. (a) False-color composite image (R: 50, G: 27, and B: 17). (b) Ground-truth map.

TABLE III
NUMBER OF TRAINING AND TEST SAMPLES USED FOR SALINAS

Class	Name	Training	Test	Total
1	Brocoli-weeds-1	101	1,908	2,009
2	Brocoli-weeds-2	187	3,539	3,726
3	Fallow	99	1,877	1,976
4	Fallow-rough-plow	70	1,324	1,394
5	Fallow-smooth	134	2,544	2,678
6	Stubble	197	3,762	3,959
7	Celery	178	3,401	3,579
8	Grapes-untrained	563	10,708	11,271
9	Soil-vinyard-develop	310	5,893	6,203
10	Corn-senesced-weeds	163	3,115	3,278
11	Lettuce-romaine-4wk	54	1,014	1,068
12	Lettuce-romaine-5wk	97	1,830	1927
13	Lettuce-romaine-6wk	46	870	916
14	Lettuce-romaine-7wk	54	1,016	1,070
15	Vinyard-untrained	364	6,904	7,268
16	Vinyard-vertical-trellis	91	1,716	1,807
	Total	2,708	51,421	54,129

kernel (SVMCK) [58], JSR [8], kernel JSR (KJSR) [11], spatial-aware CR (SaCR) [59], structure-aware collaborative representation with Tikhonov regularization (SaCRT) [60], locality and structure regularized LRR (LSLRR) [36], and generalized tensor regression based on spatially derived features (GTR-SD) [61]. The parameters of SVM and SVMCK are obtained by cross validation, and the remaining considered methods are set with reference to the original paper. The specific parameter settings are shown in Table IV, where T is the sparsity level parameter, γ is the parameter of the RBF kernel, and $\lambda, \lambda_1, \lambda_2$, and λ_3 are the specific parameters of each model.

- 2) For the proposed TLRSR-IDL method, we experientially set the maximum number of iterations $K = 5$, and $W_S = W_S^{(3)}$ can be derived from the definitions of W_S and $W_S^{(k)}$. Therefore, we only need to determine the initial value of W_S , and the value of $W_S^{(k)}$ can be uniquely determined in the k th iteration.
- 3) Three quantitative metrics, including overall accuracy (OA), average accuracy (AA), and kappa coefficient (κ), are utilized to evaluate the classification performance of different methods. All experimental results are averaged

over ten independent runs, and the training set for each run is obtained by randomly sampling from each class.

- 4) All the implementations were carried out using MATLAB R2020b on a desktop PC equipped with an Intel Core i7-11700F CPU (at 2.5 GHz) and 32 GB of RAM. The source code will be available online (<https://github.com/ZhaohuiXue/TLRSR-IDL>).

C. Parameter Sensitiveness

In our proposed method, there are four parameters that need to be tuned, including the sparse regularization parameter α , the noise regularization parameter β , the initial size of the neighborhood window W_S , and the similarity threshold S .

- 1) We tune the parameters α and β , whose values are chosen from $\{0.01, 0.1, 1, 10, 100\}$ and $\{0.02, 0.2, 2, 20, 200\}$, respectively. The evolution of OA as a function of various combinations of the two parameters is shown in Fig. 7. During the experiment, the other two parameters are fixed to $\{W_S = 7 \times 7, S = 0.85\}$, $\{W_S = 19 \times 19, S = 0.75\}$, and $\{W_S = 9 \times 9, S = 0.85\}$ for Indian Pines, Pavia University, and Salinas, respectively. It can be clearly seen that the proposed TLRSR-IDL method is more sensitive to parameter β . Especially for the Pavia University dataset, when β is increased from 0.02 to 0.2, the classification accuracy greatly improves. In addition, it can be obtained that the optimal values of α and β are $\{0.1, 2\}$, $\{0.1, 200\}$, and $\{1, 2\}$ for the three datasets.
- 2) We tune the parameter W_S , whose value is taken from 3×3 to 27×27 . The evolution of OA as a function of W_S is shown in Fig. 8(a). In this set of experiments, the other three parameters are fixed to $\{\alpha = 0.1, \beta = 2, \text{ and } S = 0.85\}$, $\{\alpha = 0.1, \beta = 200, \text{ and } S = 0.75\}$, and $\{\alpha = 1, \beta = 2, \text{ and } S = 0.85\}$ for the three datasets. As shown in Fig. 8(a), with the increase in W_S , the OA of the three datasets shows a trend of increasing at first and then decreasing, and the maximum value is obtained when the size of W_S is 7×7 , 23×23 , and 9×9 . The parameter is determined by the spatial resolution and the homogeneity of the scene. Therefore, the optimal value of W_S in Pavia University is clearly greater than that of Indian Pines and Salinas.
- 3) We tune the similarity threshold S under the premise of setting the remaining three parameters as $\{\alpha = 0.1, \beta = 2, \text{ and } W_S = 7 \times 7\}$, $\{\alpha = 0.1, \beta = 200, \text{ and } W_S = 23 \times 23\}$, and $\{\alpha = 1, \beta = 2, \text{ and } W_S = 9 \times 9\}$. The value of S is chosen from an arithmetic sequence of 0.625–0.95, and the difference between two adjacent values is 0.025. The corresponding experimental results are shown in Fig. 8(b), where the optimal similarity thresholds corresponding to the three datasets are 0.9, 0.75, and 0.85. Since Pavia University is collected by different sensors, the number of bands is less than that of Indian Pines and Salinas. The similarity threshold strongly depends on the number of bands in the dataset; therefore, the optimal value of S for Pavia University is significantly smaller than that for the other two datasets.

TABLE IV
PARAMETER SETTINGS OF DIFFERENT RELATED METHODS FOR THE THREE DATASETS

Methods	Indian Pines	Pavia University	Salinas
JSR [8]	$\{T = 20, W_S = 5 \times 5\}$	$\{T = 25, W_S = 11 \times 11\}$	$\{T = 20, W_S = 11 \times 11\}$
KJSR [11]	$\{T = 20, W_S = 5 \times 5, \gamma = 1\}$	$\{T = 25, W_S = 9 \times 9, \gamma = 1\}$	$\{T = 20, W_S = 9 \times 9, \gamma = 1\}$
SaCR [59]	$\{\lambda_1 = 10^{-6}, \lambda_2 = 1\}$	$\{\lambda_1 = 10^{-2}, \lambda_2 = 10^6\}$	$\{\lambda_1 = 10^{-2}, \lambda_2 = 1\}$
SaCRT [60]	$\{\lambda_1 = 5, \lambda_2 = 10^{-3}, \lambda_3 = 10^{-3}\}$	$\{\lambda_1 = 10, \lambda_2 = 10^{-4}, \lambda_3 = 10^{-3}\}$	$\{\lambda_1 = 10, \lambda_2 = 10^{-4}, \lambda_3 = 10^{-3}\}$
LSLRR [36]	$\{\lambda = 20, \alpha = 0.8, \beta = 0.6\}$	$\{\lambda = 10, \alpha = 0.3, \beta = 1.2\}$	$\{\lambda = 10, \alpha = 1, \beta = 0.4\}$
GTR-SD [61]	$\{L_w = 5, L_h = 5\}$	$\{L_w = 9, L_h = 9\}$	$\{L_w = 7, L_h = 7\}$

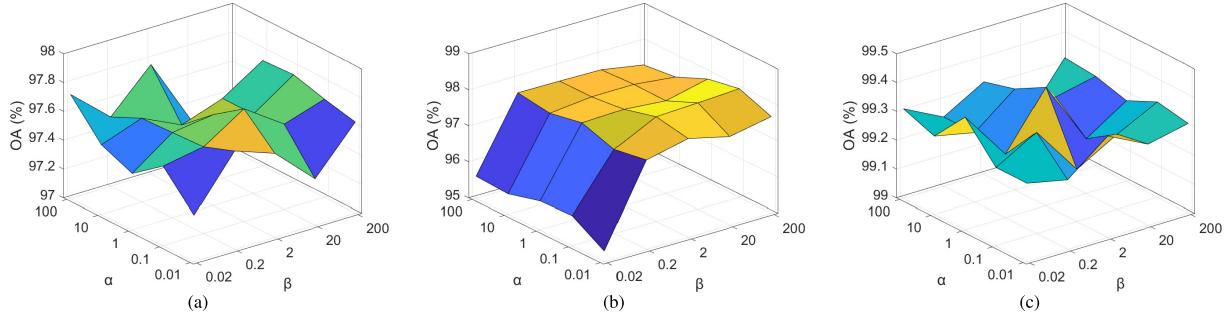


Fig. 7. Impact of α and β on the classification accuracy for different datasets. (a) Indian Pines. (b) Pavia University. (c) Salinas.

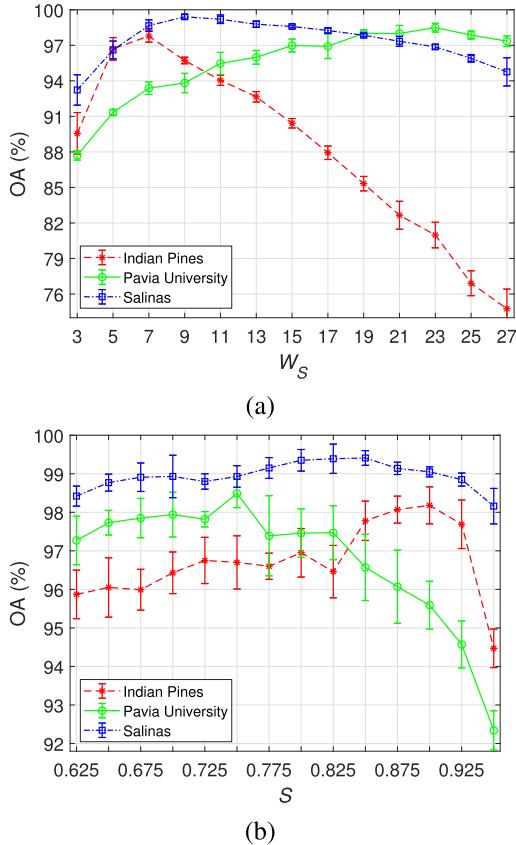


Fig. 8. Impact of (a) W_S and (b) S on classification accuracy for different datasets.

D. Classification Results

In this set of experiments, we compare the proposed method with eight other considered methods and give the classification accuracies and maps obtained by different methods for the three datasets.

1) *Indian Pines*: The classification accuracies and κ coefficients on this dataset utilizing various methods are listed in Table V. According to the results, our method achieves the highest OA of 98.18%, which is approximately 0.97%–16.83% higher than other methods. For AA and κ , the TLRSR-IDL method increases the accuracies by 2.03%–31.7% and 1.10%–19.34%, respectively, compared with the other eight methods. For class-specific accuracies, TLRSR-IDL yields the best accuracy in 12 classes, and five of them are 100%. It is interesting to note that SaCR and LSLRR also yield superior performance, with OA exceeding 95%. The classification maps are shown in Fig. 9, from which we can observe that due to a lack of spatial information, the classification map obtained by SVM has the worst smoothness. In contrast, the proposed TLRSR-IDL method obtains higher accuracy and a smoother classification map, especially for the classes *Alfalfa*, *Grass/trees*, and *Oats*.

2) *Pavia University*: As listed in Table VI, the proposed method TLRSR-IDL performs much better than the other methods with the highest accuracies. For the OA, AA, and κ coefficients, TLRSR-IDL yields 1.25%–6.89%, 1.49%–15.85%, and 1.58%–9.19% improvements compared to the others, respectively. For class-specific accuracies, both our method and SaCRT achieve higher accuracy for the three classes. It is worth noting that although the accuracy of the proposed method is not the highest for some classes, the gap with the highest accuracy is very small. For example, for classes *Meadows* and *Metal Sheets*, the accuracy difference is less than 0.1%. In addition, both SaCR and LSLRR show competitive performance, with OA reaching over 97%. Classification maps obtained by different methods are shown in Fig. 10, where TLRSR-IDL provides a more accurate and smooth classification map, which can be verified in classifying the classes *Gravel*, *Bitumen*, and *Bricks*.

3) *Salinas*: The classification accuracies for each of the 16 classes, OA, AA, and κ coefficient on the test set utilizing

TABLE V
CLASSIFICATION ACCURACIES OBTAINED BY DIFFERENT METHODS FOR INDIAN PINES (10% FOR TRAINING)

Class	SVM [57]	SVMCK [58]	JSR [8]	KJSR [11]	SaCR [59]	SaCRT [60]	LSLRR [36]	GTR-SD [61]	TLRSR-IDL (Ours)
1	11.46±12.44	82.62±10.10	87.8±12.44	74.60±22.02	95.61±3.24	84.88±13.75	7.80±0.73	43.41±12.47	100.0±0.00
2	76.93±1.87	90.17±2.31	88.41±2.15	88.80±1.90	95.58±0.95	88.54±1.34	90.95±2.37	95.77±1.24	97.21±1.23
3	64.70±2.63	95.15±1.54	84.47±3.82	92.11±4.14	93.20±1.63	87.20±2.88	92.66±2.48	90.17±2.31	96.83±1.51
4	58.87±8.19	83.66±4.34	79.67±6.06	92.00±4.47	92.25±4.93	86.38±4.65	97.00±1.63	81.69±4.16	97.56±5.66
5	88.62±1.59	93.57±2.57	94.51±2.96	92.85±1.34	94.32±2.35	95.99±2.29	94.33±1.41	95.44±1.54	93.59±3.67
6	94.89±2.03	99.23±0.38	98.63±0.89	98.74±0.27	99.76±0.16	98.84±0.61	98.95±0.36	99.41±0.31	100.0±0.00
7	6.40±11.81	49.20±29.21	96.00±3.77	36.50±18.76	97.60±2.65	96.00±8.00	97.20±1.83	62.00±7.85	100.0±0.00
8	99.12±0.92	99.75±0.25	99.40±0.79	99.93±0.11	100.0±0.00	98.93±0.71	100.0±0.00	99.98±0.02	100.0±0.00
9	0.00±0.00	55.56±9.63	81.11±13.15	15.88±21.05	91.11±6.19	84.44±8.16	92.22±13.19	22.78±13.25	100.0±0.00
10	68.19±3.47	90.28±1.87	93.86±1.67	91.16±1.76	95.54±1.87	84.67±1.97	93.34±2.42	85.92±4.08	97.63±2.04
11	86.01±1.97	96.43±1.25	95.55±0.80	95.28±1.14	98.71±0.28	97.53±0.81	96.36±1.05	96.15±1.01	98.54±0.77
12	78.67±3.38	89.65±3.53	84.89±3.37	95.62±1.68	97.28±0.70	89.38±3.68	89.53±2.88	89.12±2.46	98.39±1.08
13	95.11±3.10	98.61±1.16	99.29±1.06	99.09±0.58	99.73±0.27	97.83±1.85	99.02±0.53	98.37±0.94	99.95±0.16
14	96.92±0.75	99.16±0.33	99.02±0.44	99.04±0.81	99.96±0.04	98.73±0.78	99.72±0.16	99.63±0.36	99.95±0.04
15	56.71±4.70	96.41±2.50	78.65±5.07	91.37±6.62	94.93±1.87	96.66±1.16	89.48±4.92	93.40±3.44	96.28±4.27
16	85.71±5.50	85.41±6.16	98.21±2.53	95.74±2.91	97.38±1.83	98.80±1.08	95.66±4.48	95.66±4.48	99.64±0.77
OA	81.35±0.71	94.41±0.50	92.69±0.45	94.01±0.67	97.21±0.21	93.63±0.51	95.10±0.44	94.17±0.44	98.18±0.48
AA	66.77±1.33	87.80±3.19	91.22±1.43	84.92±2.51	96.44±0.76	92.80±1.19	95.26±1.05	84.31±1.13	98.47±0.61
$\kappa \times 100$	78.58±0.81	93.63±0.57	91.65±0.52	93.18±0.76	96.82±0.24	92.70±0.59	94.42±0.51	93.34±0.51	97.92±0.55
Time (s)	212	247	56	67	98	92	83	27	129

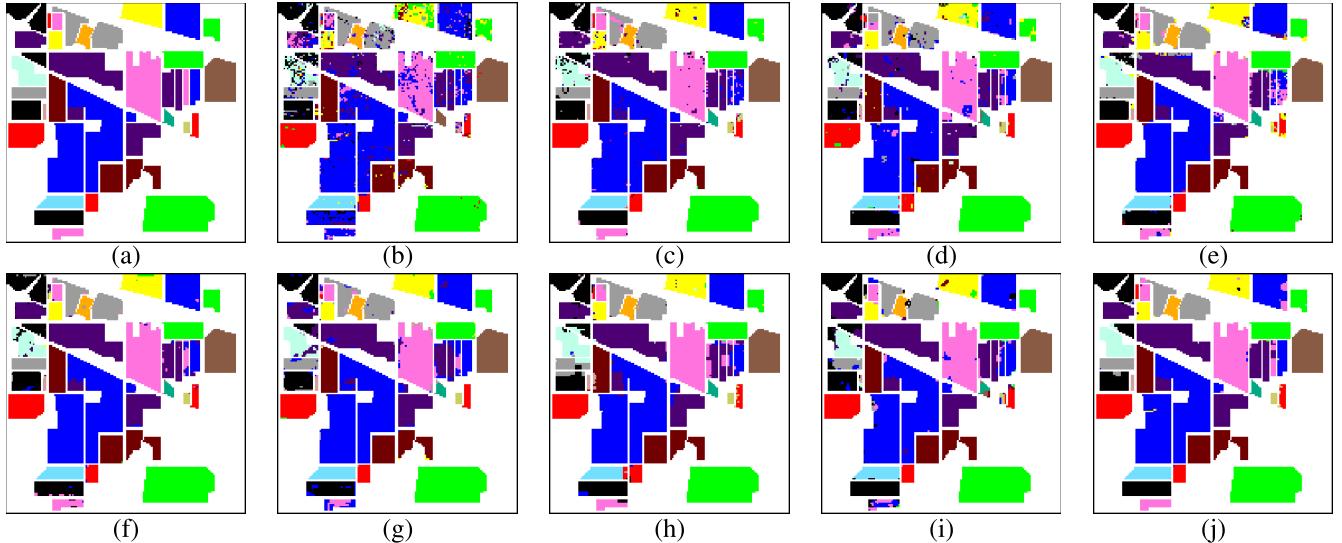


Fig. 9. Classification maps obtained by different methods for Indian Pines. (a) Ground truth. (b) SVM (81.35%). (c) SVMCK (94.41%). (d) JSR (92.69%). (e) KJSR (94.01%). (f) SaCR (97.21%). (g) SaCRT (93.63%). (h) LSLRR (95.10%). (i) GTR-SD (94.17%). (j) TLRSR-IDL (98.18%). The OAs are given in parentheses.

TABLE VI
CLASSIFICATION ACCURACIES OBTAINED BY DIFFERENT METHODS FOR PAVIA UNIVERSITY (5% FOR TRAINING)

Class	SVM [57]	SVMCK [58]	JSR [8]	KJSR [11]	SaCR [59]	SaCRT [60]	LSLRR [36]	GTR-SD [61]	TLRSR-IDL (Ours)
1	93.42±0.60	96.56±0.74	86.53±1.06	90.79±2.72	91.77±0.73	99.08±0.20	97.98±0.95	93.97±0.65	98.11±0.36
2	97.91±0.29	98.39±0.33	99.96±0.07	99.26±0.27	99.82±0.02	100.0±0.00	99.57±0.30	99.92±0.06	99.98±0.05
3	76.56±2.00	96.57±1.33	86.82±2.50	91.96±2.92	95.54±0.67	84.87±2.11	98.23±1.10	76.38±3.35	99.82±0.09
4	92.80±1.14	99.38±0.23	93.83±1.67	91.95±1.07	95.37±0.54	85.68±1.40	94.46±2.12	87.75±1.44	95.19±0.49
5	98.74±0.42	83.19±6.40	100.0±0.00	100.0±0.00	99.77±0.10	100.0±0.00	99.91±0.03	99.91±0.05	99.92±0.02
6	88.70±1.05	97.26±0.60	71.01±3.82	92.33±1.38	99.63±0.06	91.36±0.98	88.86±4.41	85.52±1.13	94.44±2.44
7	84.16±2.05	97.85±0.65	88.10±2.73	82.88±4.59	98.08±0.29	94.30±2.11	99.35±0.77	89.13±3.30	99.79±0.06
8	90.10±0.90	96.66±1.21	95.63±1.47	86.68±2.82	93.73±1.15	88.90±1.87	97.13±1.02	80.67±1.76	98.44±0.49
9	99.37±0.24	85.89±7.36	84.89±1.22	77.98±1.07	86.46±1.66	89.63±1.41	90.95±1.20	23.92±6.44	94.13±1.48
OA	93.68±0.20	96.83±0.20	92.32±1.02	94.21±0.48	97.15±0.09	95.71±0.25	97.24±0.59	91.60±0.38	98.49±0.37
AA	91.31±0.41	94.63±0.87	89.64±1.02	90.42±0.49	95.57±0.16	92.65±0.61	96.27±0.46	81.91±0.88	97.76±0.35
$\kappa \times 100$	91.58±0.27	96.06±0.27	89.62±0.66	92.30±0.61	96.21±0.12	94.26±0.34	96.31±0.80	88.70±0.53	97.89±0.38
Time (s)	195	223	174	240	1,386	1,131	598	82	1,263

various methods are reported in Table VII, and the corresponding classification maps are presented in Fig. 11. Similar to Indian Pines, the proposed method TLRSR-IDL achieves

better performance when compared with other classifiers in nine of 16 classes, and the classification accuracies of six classes are 100%. Again, TLRSR-IDL yields the highest OA,

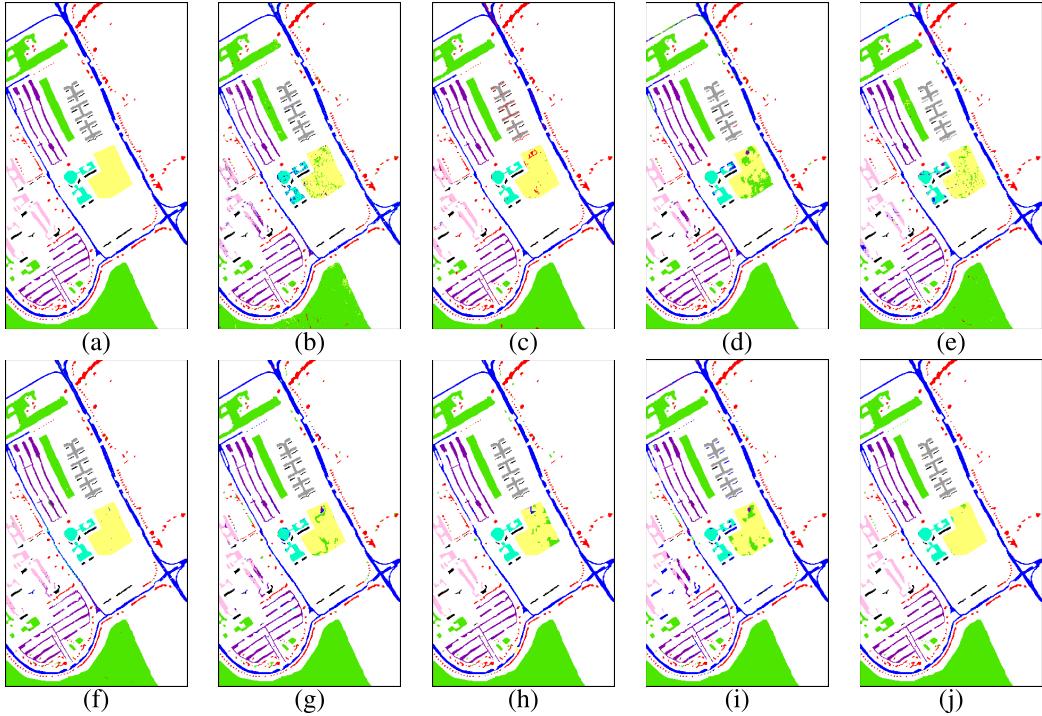


Fig. 10. Classification maps obtained by different methods for Pavia University. (a) Ground truth. (b) SVM (93.68%). (c) SVMCK (96.83%). (d) JSR (92.32%). (e) KJSR (94.21%). (f) SaCR (97.15%). (g) SaCRT (95.71%). (h) LSLRR (97.24%). (i) GTR-SD (91.60%). (j) TLRSR-IDL (98.49%). The OAs are given in parentheses.

TABLE VII
CLASSIFICATION ACCURACIES OBTAINED BY DIFFERENT METHODS FOR SALINAS (5% FOR TRAINING)

Class	SVM [57]	SVMCK [58]	JSR [8]	KJSR [11]	SaCR [59]	SaCRT [60]	LSLRR [36]	GTR-SD [61]	TLRSR-IDL (Ours)
1	99.32±0.37	98.92±0.49	100.0±0.00	99.98±0.03	99.72±0.07	99.85±0.15	99.99±0.02	99.80±0.17	100.0±0.00
2	99.58±0.18	98.09±1.01	99.95±0.05	99.96±0.06	99.97±0.01	99.95±0.03	99.84±0.13	100.0±0.00	100.0±0.00
3	99.30±0.43	99.51±0.29	99.70±0.27	98.19±0.52	99.89±0.07	99.81±0.11	98.82±1.38	100.0±0.00	100.0±0.00
4	98.90±0.78	98.09±2.00	98.26±0.60	98.70±0.59	99.12±0.15	97.85±0.73	90.89±2.09	99.28±0.20	96.84±1.28
5	98.46±0.38	97.85±0.54	99.14±0.41	98.99±1.30	98.79±0.26	99.46±0.38	98.74±0.56	99.47±0.21	98.15±0.67
6	99.61±0.15	98.64±0.59	99.94±0.05	99.99±0.01	99.88±0.05	100.0±0.00	99.92±0.05	100.0±0.00	99.93±0.02
7	99.36±0.20	98.72±0.49	99.97±0.02	99.46±0.20	99.93±0.01	99.93±0.04	99.72±0.04	100.0±0.00	99.81±0.05
8	88.03±1.18	95.44±0.82	94.75±0.81	96.69±0.82	98.35±0.27	98.94±0.26	97.26±0.35	92.55±0.75	98.99±0.82
9	99.60±0.28	99.48±0.32	99.97±0.03	99.99±0.01	100.0±0.00	99.98±0.03	99.96±0.03	100.0±0.00	100.0±0.00
10	94.63±1.16	95.19±1.53	99.16±0.32	98.33±0.46	96.87±0.49	99.80±0.15	98.14±0.44	97.80±0.80	98.81±0.60
11	97.70±1.34	96.33±1.90	100.0±0.00	97.08±3.49	96.59±0.96	98.72±1.15	98.44±0.85	99.87±0.06	100.0±0.00
12	99.54±0.24	99.93±0.11	100.0±0.00	100.00±0.00	99.91±0.04	99.99±0.01	97.84±0.52	100.0±0.00	100.0±0.00
13	98.03±0.99	97.37±0.69	99.98±0.05	99.03±0.87	98.34±0.53	99.95±0.06	99.21±0.38	99.28±0.65	99.98±0.05
14	94.65±1.20	89.86±5.27	99.70±0.54	99.25±0.36	96.11±0.67	99.61±0.32	97.03±1.19	98.86±0.30	99.51±0.50
15	71.18±1.98	92.43±1.28	90.50±2.18	88.83±1.46	93.24±0.74	92.34±1.47	97.80±0.59	78.86±1.19	99.33±0.44
16	97.84±0.62	95.43±4.54	99.91±0.19	99.42±0.34	98.78±0.18	98.75±0.47	99.49±0.38	99.59±0.15	99.62±0.61
OA	92.74±0.12	96.68±0.40	97.46±0.21	97.41±0.16	98.23±0.09	98.56±0.17	98.44±1.19	95.32±0.19	99.41±0.19
AA	95.98±0.19	96.95±0.58	98.81±0.12	98.37±0.24	98.47±0.13	99.06±0.08	98.31±0.59	97.81±0.08	99.44±0.09
$\kappa \times 100$	91.91±0.14	96.30±0.45	97.17±0.24	97.12±0.17	98.03±0.11	98.39±0.19	98.26±0.58	94.78±0.21	99.35±0.19
Time (s)	612	675	417	473	1,527	1,325	1,128	73	976

AA, and κ coefficients. In addition, SaCR, SaCRT, and LSLRR also yield superior performance, with only approximately 1% lower OA than the proposed method. As shown in Fig. 11, it can be seen that each classifier has significant differences in the classification results of *Grapes-untrained* and *Vinyard-untrained*, whereas the proposed method obtains a more accurate and smooth classification map.

E. Generalization Performance

In this experiment, we examine the impacts of the number of training samples on classification performance for various

methods on the three datasets. As shown in Fig. 12, with the increase in training samples, the classification accuracy of each method continuously improves, and TLRSR-IDL obtains the highest OAs in most of the cases for the three datasets, illustrating a better generalization performance. For Indian Pines, we randomly choose 1%–10% of labeled samples per class for training, and the proposed method is significantly better than others when the number of training samples is greater than 5%. For Pavia University and Salinas, the proposed method always has the highest OA when the ratio of training samples ranges from 0.5% to 5%.

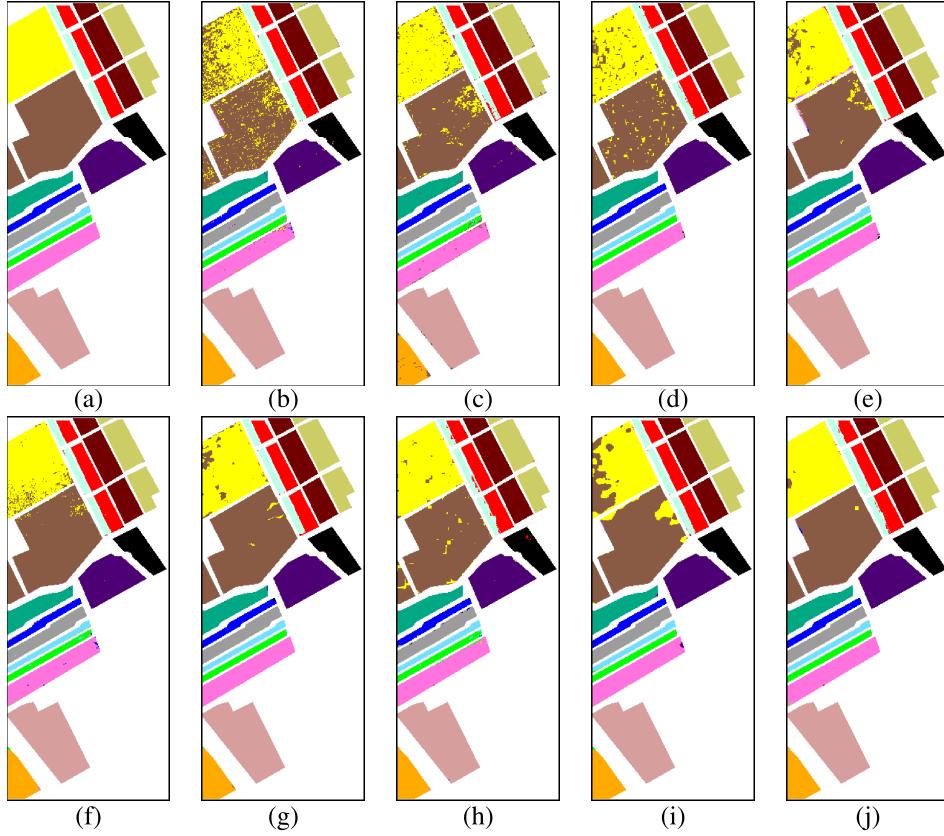


Fig. 11. Classification maps obtained by different methods for Salinas. (a) Ground truth. (b) SVM (92.74%). (c) SVMCK (96.68%). (d) JSR (97.46%). (e) KJSR (97.41%). (f) SaCR (98.23%). (g) SaCRT (98.56%). (h) LSLRR (98.44%). (i) GTR-SD (95.32%). (j) TLRSR-IDL (99.41%). The OAs are given in parentheses.

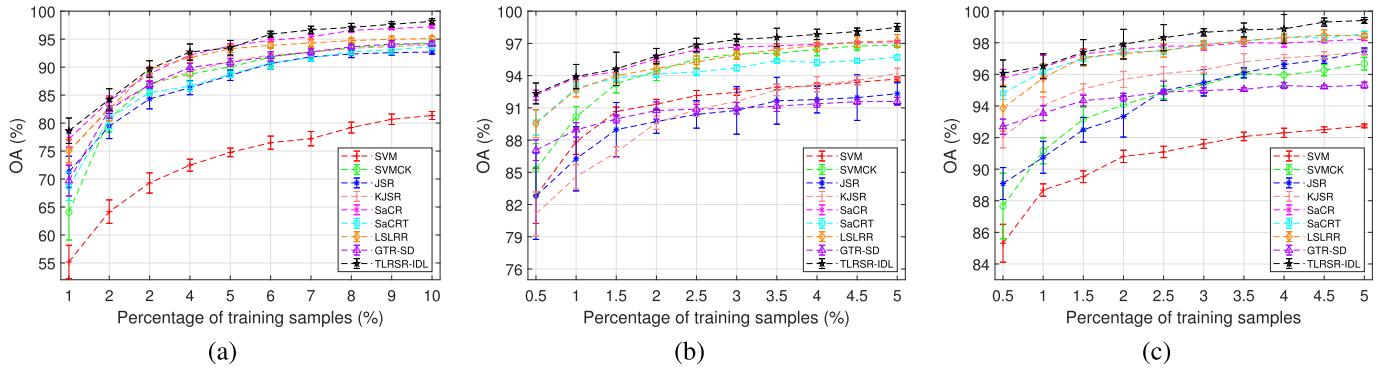


Fig. 12. Evolution of OAs obtained by different methods as a function of different percentages of training samples. (a) Indian Pines. (b) Pavia University. (c) Salinas.

F. Comparison With Deep Learning Methods

In this section, we compare the proposed TLRSR-IDL method with four deep learning-based HSI classification methods, including spectral-spatial residual network (SSRN) [62] based on 3-D convolutional neural network (CNN), deep pyramidal residual network (DPRN) [63] based on 2-D CNN, 3-D octave and 2-D vanilla mixed convolutional network (Oct-MCNN) [64] based on mixed 2-D CNN and 3-D CNN, and spectral-spatial feature tokenization transformer (SSFTT) [65] based on transformer framework. The classification results for the three datasets are listed in Tables VIII–X, and Fig. 13 shows the OA obtained by each method under a different number of training samples.

As shown in Table VIII, the classification performance of TLRSR-IDL for Indian Pines is better than that of the other four deep learning-based methods, which not only achieve the highest OA, AA, and κ but also the best performance on 11 classes. For classes *Grass/pasture-mowed* and *Oats* with very few samples, TLRSR-IDL achieves 100% accuracy, while the rest of the methods perform poorly. Similarly, for Salinas, our method also outperforms the other comparison methods. As shown in Table X, except AA, both OA and κ are the best. For class-specific accuracies, Oct-MCNN is the most competitive method, which has the highest accuracy on seven classes with five classes being 100%, while TLRSR-IDL reaches the highest accuracy on 11 classes with six classes

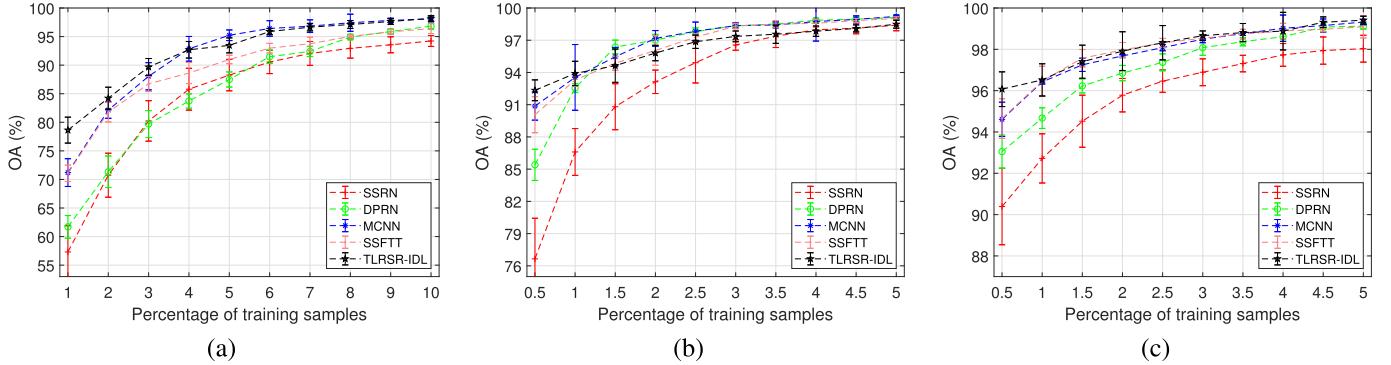


Fig. 13. Evolution of OAs obtained by different deep learning methods and TLRSR-IDL as a function of different percentages of training samples. (a) Indian Pines. (b) Pavia University. (c) Salinas.

TABLE VIII

CLASSIFICATION ACCURACIES OBTAINED BY DIFFERENT DEEP LEARNING METHODS AND TLRSR-IDL FOR INDIAN PINES (10% FOR TRAINING)

Class	SSRN [62]	DPRN [63]	Oct-MCNN [64]	SSFTT [65]	TLRSR-IDL
1	82.82 \pm 10.07	81.95 \pm 17.63	96.10 \pm 10.13	87.07 \pm 10.86	100.00 \pm 0.00
2	91.07 \pm 3.07	96.04 \pm 1.65	96.20 \pm 0.93	95.70 \pm 1.17	97.21 \pm 1.23
3	95.88 \pm 1.50	97.03 \pm 1.74	98.74 \pm 0.96	98.32 \pm 0.67	96.83 \pm 1.51
4	94.58 \pm 3.16	93.43 \pm 3.35	94.04 \pm 3.98	93.38 \pm 3.87	97.56 \pm 5.66
5	93.53 \pm 3.24	95.66 \pm 3.27	98.74 \pm 2.05	92.39 \pm 2.61	93.59 \pm 3.67
6	98.11 \pm 1.57	99.27 \pm 0.74	99.47 \pm 0.30	98.81 \pm 1.58	100.00 \pm 0.00
7	83.75 \pm 20.02	72.40 \pm 29.15	99.20 \pm 2.40	91.20 \pm 5.00	100.00 \pm 0.00
8	98.08 \pm 3.43	99.84 \pm 0.28	99.91 \pm 0.28	99.93 \pm 0.11	100.00 \pm 0.00
9	79.41 \pm 21.57	56.67 \pm 15.28	70.00 \pm 26.32	82.78 \pm 6.78	100.00 \pm 0.00
10	89.31 \pm 3.76	94.78 \pm 2.49	98.65 \pm 0.78	92.06 \pm 2.72	97.63 \pm 2.04
11	95.88 \pm 2.75	97.96 \pm 0.85	98.23 \pm 0.75	97.59 \pm 1.81	98.54 \pm 0.77
12	86.87 \pm 3.02	91.40 \pm 4.25	96.93 \pm 2.01	90.84 \pm 2.84	98.39 \pm 1.08
13	98.63 \pm 1.70	99.41 \pm 1.44	99.46 \pm 0.64	98.70 \pm 1.06	99.95 \pm 0.16
14	98.48 \pm 0.94	99.33 \pm 0.54	99.98 \pm 0.02	99.48 \pm 0.39	99.95 \pm 0.04
15	90.64 \pm 5.67	97.49 \pm 1.34	95.22 \pm 2.94	94.90 \pm 2.39	96.28 \pm 4.27
16	94.81 \pm 5.11	94.17 \pm 7.96	97.26 \pm 3.37	99.05 \pm 1.04	99.64 \pm 0.77
OA	94.23 \pm 0.95	96.83 \pm 0.52	98.10 \pm 0.50	96.40 \pm 0.87	98.18 \pm 0.48
AA	91.99 \pm 1.60	91.68 \pm 2.71	96.13 \pm 2.48	94.51 \pm 1.38	98.47 \pm 0.61
$\kappa \times 100$	93.42 \pm 1.08	96.38 \pm 0.59	97.84 \pm 0.58	95.89 \pm 0.99	97.92 \pm 0.55
Time (s)	267	99	71	27	129

TABLE X

CLASSIFICATION ACCURACIES OBTAINED BY DIFFERENT DEEP LEARNING METHODS AND TLRSR-IDL FOR SALINAS (5% FOR TRAINING)

Class	SSRN [62]	DPRN [63]	Oct-MCNN [64]	SSFTT [65]	TLRSR-IDL
1	99.17 \pm 1.07	99.96 \pm 0.13	100.00 \pm 0.00	99.74 \pm 0.67	100.00 \pm 0.00
2	99.99 \pm 0.02	99.95 \pm 0.11	100.00 \pm 0.00	99.99 \pm 0.01	100.00 \pm 0.00
3	98.31 \pm 2.21	99.95 \pm 0.12	100.00 \pm 0.00	99.61 \pm 0.53	100.00 \pm 0.00
4	99.84 \pm 0.22	98.98 \pm 1.59	99.59 \pm 0.88	99.43 \pm 0.54	98.84 \pm 1.28
5	99.15 \pm 0.95	99.32 \pm 0.70	99.67 \pm 0.46	98.46 \pm 1.07	98.15 \pm 0.67
6	99.98 \pm 0.06	99.95 \pm 0.12	99.99 \pm 0.03	99.66 \pm 0.56	99.93 \pm 0.02
7	99.92 \pm 0.07	99.94 \pm 0.12	100.00 \pm 0.00	99.93 \pm 0.10	99.81 \pm 0.05
8	96.24 \pm 1.88	98.01 \pm 1.00	98.87 \pm 1.02	98.59 \pm 1.99	98.99 \pm 0.82
9	99.80 \pm 0.18	100.00 \pm 0.00	99.99 \pm 0.01	99.95 \pm 0.05	100.00 \pm 0.00
10	98.62 \pm 1.05	99.58 \pm 0.23	99.77 \pm 0.15	98.39 \pm 0.67	98.81 \pm 0.05
11	98.85 \pm 1.07	99.57 \pm 0.69	99.79 \pm 0.13	99.58 \pm 0.50	100.00 \pm 0.00
12	99.99 \pm 0.01	99.90 \pm 0.22	100.00 \pm 0.00	99.64 \pm 0.32	100.00 \pm 0.00
13	99.47 \pm 0.67	99.45 \pm 0.81	99.94 \pm 0.11	98.48 \pm 1.85	99.98 \pm 0.05
14	98.91 \pm 0.64	97.81 \pm 3.58	99.26 \pm 0.49	99.14 \pm 0.41	99.51 \pm 0.50
15	93.60 \pm 4.07	97.88 \pm 1.18	97.34 \pm 1.06	98.13 \pm 2.79	99.33 \pm 0.44
16	99.26 \pm 0.52	99.56 \pm 0.44	99.37 \pm 0.17	98.49 \pm 1.06	99.62 \pm 0.61
OA	98.03 \pm 0.65	99.12 \pm 0.14	99.32 \pm 0.13	99.09 \pm 0.54	99.41 \pm 0.19
AA	98.82 \pm 0.40	99.36 \pm 0.24	99.60 \pm 0.07	99.20 \pm 0.39	99.44 \pm 0.09
$\kappa \times 100$	97.80 \pm 0.72	99.02 \pm 0.15	99.25 \pm 0.15	98.99 \pm 0.60	99.35 \pm 0.19
Time (s)	1,001	107	101	52	976

TABLE IX

CLASSIFICATION ACCURACIES OBTAINED BY DIFFERENT DEEP LEARNING METHODS AND TLRSR-IDL FOR PAVIA UNIVERSITY (5% FOR TRAINING)

Class	SSRN [62]	DPRN [63]	Oct-MCNN [64]	SSFTT [65]	TLRSR-IDL
1	98.51 \pm 0.68	99.13 \pm 0.45	99.13 \pm 0.27	99.37 \pm 0.61	98.11 \pm 0.36
2	98.95 \pm 0.89	99.95 \pm 0.04	99.92 \pm 0.04	99.85 \pm 0.15	99.98 \pm 0.05
3	95.52 \pm 2.56	94.35 \pm 2.37	95.74 \pm 0.93	94.06 \pm 0.82	99.82 \pm 0.09
4	96.93 \pm 1.92	98.51 \pm 0.48	98.73 \pm 0.50	98.62 \pm 0.38	95.19 \pm 0.49
5	99.95 \pm 0.07	99.98 \pm 0.05	100.00 \pm 0.00	99.95 \pm 0.09	99.92 \pm 0.02
6	99.47 \pm 0.56	99.55 \pm 0.41	99.62 \pm 0.36	99.75 \pm 0.21	94.44 \pm 2.44
7	95.53 \pm 2.21	99.36 \pm 0.72	99.55 \pm 0.33	98.77 \pm 0.74	99.79 \pm 0.06
8	96.94 \pm 1.12	96.99 \pm 1.35	97.08 \pm 1.20	96.23 \pm 0.79	98.44 \pm 0.49
9	99.09 \pm 0.90	99.57 \pm 0.71	99.93 \pm 0.05	99.04 \pm 0.72	94.13 \pm 1.48
OA	98.39 \pm 0.51	99.12 \pm 0.18	99.22 \pm 0.12	99.03 \pm 0.15	98.49 \pm 0.37
AA	97.88 \pm 0.50	98.60 \pm 0.33	98.86 \pm 0.16	98.41 \pm 0.17	97.76 \pm 0.35
$\kappa \times 100$	97.86 \pm 0.67	98.83 \pm 0.24	98.97 \pm 0.16	98.72 \pm 0.20	97.89 \pm 0.38
Time (s)	1,533	166	128	64	1,263

being 100%. The classification results of Pavia University are listed in Table IX. For this dataset, deep learning-related classification methods perform better, and the accuracy of our method is only higher than that of SSRN.

From the variation trend of OA with the number of training samples shown in Fig. 13, it is easy to find that the classification performance of four deep learning-based methods is not outstanding with few training samples. For instance, for Indian Pines with 1% training samples, the OA of our method is 78.63%, followed by MCNN whose OA is only 71.2%. For Pavia University and Salinas with 0.5% training

samples, the OAs of our method are 92.37% and 96.07%, which are 1.47%–19.69% and 1.42%–5.68% higher than the remaining four methods, respectively. The main reason for this phenomenon is that deep neural networks have huge trainable parameters. When the number of training samples is insufficient, overfitting will inevitably occur, thus affecting the classification performance. In practical applications, labeling processing is usually time-consuming and labor-intensive, making it difficult to meet the needs of deep learning models. In contrast, the proposed TLRSR-IDL method achieves higher classification accuracy under the premise of fewer training samples, proving its better practicability.

IV. DISCUSSION

A. Effectiveness of IDL

1) *Correctness of IDL*: As reported in Tables XI–XIII, to evaluate the correctness of augmented samples in the IDL scheme, we report the number of augmented samples introduced into the dictionary in different stages for the three datasets. Nevertheless, we spatially visualize the augmented samples that are correctly and wrongly introduced into the dictionary, as shown in Figs. 14–16. According to the results, it is easy to find that there is a positive correlation between the number of learning times and the number of augmented

TABLE XI
NUMBER OF AUGMENTED SAMPLES INTRODUCED
BY IDL FOR INDIAN PINES

Class	D	$D^{(1)}$	$D^{(2)}$	$D^{(3)}$	$D^{(4)}$	$D^{(5)}$
Alfalfa	5	41	42	45	46	43
Corn-notill	143	508	813	1,032	1,190	1,277
Corn-mintill	83	486	598	710	750	765
Corn	24	117	157	197	236	242
Grass/pasture	49	346	359	404	434	436
Grass/trees	73	650	698	716	731	737
Grass/pasture-mowed	3	18	20	23	29	30
Hay-windrowed	48	458	478	478	478	478
Oats	2	2	4	7	11	13
Soybean-notill	98	502	715	827	892	957
Soybean-mintill	246	1,426	2,068	2,244	2,307	2,325
Soybean-clean	60	338	367	450	527	555
Wheat	21	182	192	199	205	205
Woods	127	1,262	1,263	1,264	1,264	1,264
Buildings-Grass-Trees-Drives	39	91	226	291	369	376
Stone-Steel-Towers	10	77	82	85	90	88
Total samples introduced	1,031	6,504	8,082	8,972	9,559	9,791
Remove training samples	0	5,473	7,051	7,941	8,528	8,760
Correctly introduced	-	5,461	7,036	7,927	8,458	8,703
Correctness	-	99.78%	99.79%	99.82%	99.18%	99.35%

TABLE XII
NUMBER OF AUGMENTED SAMPLES INTRODUCED
BY IDL FOR PAVIA UNIVERSITY

Class	D	$D^{(1)}$	$D^{(2)}$	$D^{(3)}$	$D^{(4)}$	$D^{(5)}$
Asphalt	332	4,632	5,748	5,877	5,817	5,846
Meadows	933	16,329	18,164	18,343	18,469	18,587
Gravel	105	1,813	2,026	2,042	2,056	2,069
Trees	153	2,359	2,550	2,592	2,616	2,589
Metal Sheets	68	1,247	1,273	1,291	1,305	1,317
Bare Soil	252	293	3,616	3,191	3,483	3,834
Bitumen	67	1,152	1,134	1,204	1,248	1,272
Bricks	185	2,538	2,842	3,087	3,377	3,512
Shadows	48	683	740	754	768	780
Total samples introduced	2,143	31,046	38,093	38,381	39,139	39,806
Remove training samples	0	28,903	35,950	36,238	36,996	37,663
Correctly introduced	-	28,852	35,892	36,175	36,928	37,580
Correctness	-	99.82%	99.84%	99.83%	99.82%	99.78%

TABLE XIII
NUMBER OF AUGMENTED SAMPLES INTRODUCED BY IDL FOR SALINAS

Class	D	$D^{(1)}$	$D^{(2)}$	$D^{(3)}$	$D^{(4)}$	$D^{(5)}$
Brocoli-weeds-1	101	2,009	2,009	2,009	2,009	2,009
Brocoli-weeds-2	187	3,636	3,671	3,689	3,717	3,726
Fallow	99	1,857	1,976	1,976	1,976	1,976
Fallow-rough-plow	70	311	800	1,040	1,207	1,313
Fallow-smooth	134	1,919	1,959	2,176	2,379	2,536
Stubble	197	3,848	3,902	3,937	3,948	3,938
Celery	178	3,562	3,563	3,565	3,571	3,574
Grapes-untrained	563	1,018	3,363	7,582	10,280	10,976
Soil-vinyard-develop	310	6,022	6,185	6,186	6,204	6,204
Corn-senesced-weeds	163	1,609	2,319	2,788	2,892	3,120
Lettuce-romaine-4wk	54	651	891	1,017	1,061	1,079
Lettuce-romaine-5wk	97	995	1,398	1,584	1,462	1,792
Lettuce-romaine-6wk	46	490	729	818	897	905
Lettuce-romaine-7wk	54	401	701	939	1,060	1,078
Vinyard-untrained	364	1,940	3,811	6,052	6,916	7,249
Vinyard-vertical-trellis	91	1,713	1,763	1,773	1,808	1,809
Total samples introduced	2,708	31,981	39,040	47,131	51,387	53,284
Remove training samples	0	29,273	36,332	44,423	48,679	50,576
Correctly introduced	-	29,261	36,319	44,409	48,655	50,475
Correctness	-	99.96%	99.96%	99.97%	99.95%	99.80%

samples. In addition, as shown in the last row of the three tables, the augmented samples have extremely high reliability.

- For Indian Pines, the number of augmented samples increased from 5461 to 8703 during the five-time iterative learning, and the AA of augmented samples was 99.58%. The corresponding spatial visualization maps of introduced augmented samples are shown in Fig. 14, where square markers denote the wrongly augmented samples and the marker edge colors are set to black (for easy distinction, the marker edge colors are set to red when the marker face colors are black). From Fig. 14(b), we observe that the number of incorrectly augmented samples is very small, and they are mainly in the boundary between classes.

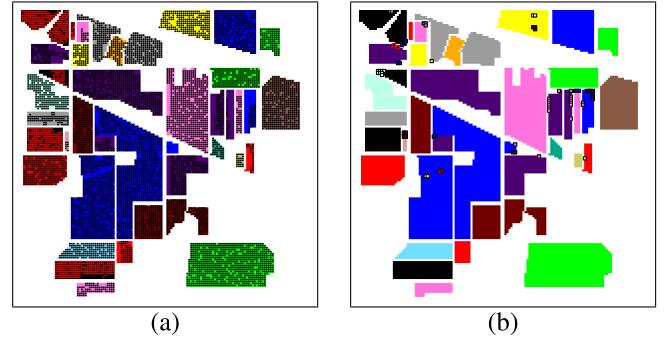


Fig. 14. Spatial visualization of augmented samples by IDL for Indian Pines. (a) Correctly augmented samples. (b) Incorrectly augmented samples. Square markers denote the incorrectly augmented samples, and the marker edge colors are set to black.

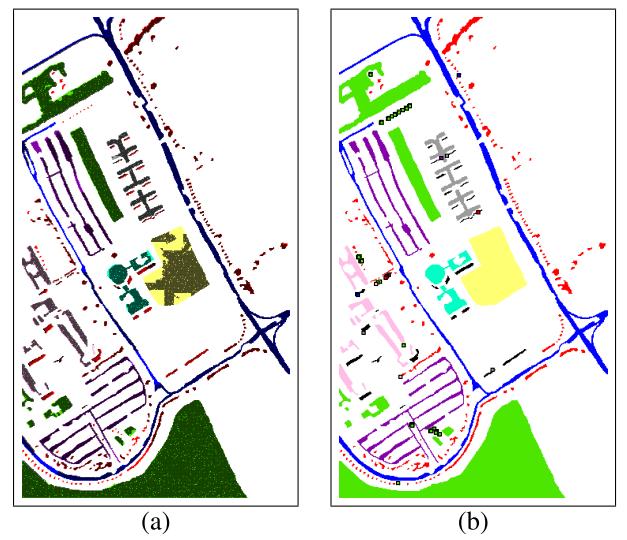


Fig. 15. Spatial visualization of augmented samples by IDL for Pavia University. (a) Correctly augmented samples. (b) Incorrectly augmented samples. Square markers denote the incorrectly augmented samples, and the marker edge colors are set to black.

- For Pavia University, the number of augmented samples increased from 28 852 to 37 580, and the accuracy was always approximately 99.82%. The spatial visualization maps of the introduced augmented samples for Pavia University are shown in Fig. 15.
- For Salinas, the number of incremental samples increased from 29 261 to 50 457. Similar to the other two datasets, the sample-introducing accuracy of the IDL scheme is extremely high, with an AA of 99.93%. The spatial visualization maps of introduced augmented samples for Salinas are shown in Fig. 16, and the wrongly augmented samples are mainly concentrated in the edge regions of classes *Fallow-rough-plow* and *Fallow-smooth*.
- Impact of IDL on Classification Performance:** To examine the impact of IDL on classification performance, we plot the evolution of classification accuracy at different learning times. As shown in Fig. 17, with the increase in learning times, the accuracy shows an almost monotonically increasing trend

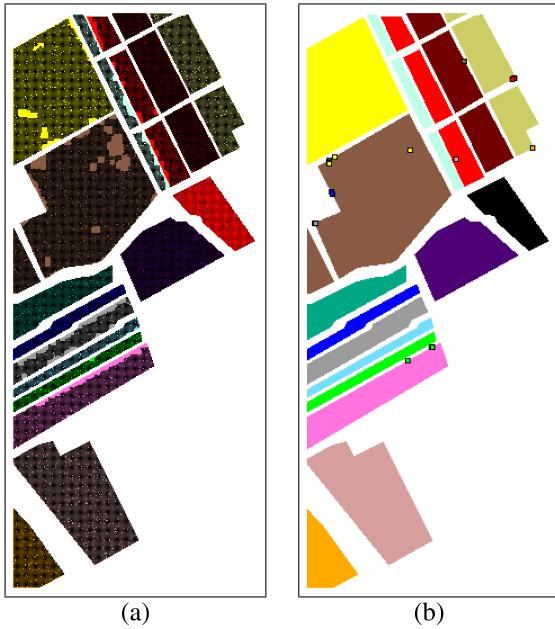


Fig. 16. Spatial visualization of augmented samples by IDL for Salinas. (a) Correctly augmented samples. (b) Incorrectly augmented samples. Square markers denote the incorrectly augmented samples and the marker edge colors are set to black.

in the three datasets because the representation ability of the dictionary set is continuously improved.

- 1) For Indian Pines, the OA based on the initial dictionary is only 91.27%, which is improved by approximately 7% after completing five iterations of IDL learning, and AA and κ are increased by approximately 6% and 8%, respectively.
- 2) For Pavia University, the classification accuracy is significantly improved after completing dictionary iterative learning, with approximately 4%, 4%, and 5% improvements for OA, AA, and κ , respectively.
- 3) For Salinas, after completing five IDL iterations, OA, AA, and κ improve from 90.98%, 95.28%, and 89.99% to 99.41%, 99.44%, and 99.35%, respectively, which proves the effectiveness of our proposed IDL scheme.

B. Ablation Study

To exploit the factors affecting the accuracy of TLRSR-IDL, we design a series of variant methods: 1) NRN for TLRSR with a nonregularized neighborhood (TLRSR-NRN); 2) NRN-IDL for TLRSR with a nonregularized neighborhood and IDL (TLRSR-NRN-IDL); 3) Euc-IDL for TLRSR with a regularized neighborhood constructed by Euclidean distance and IDL (TLRSR-Euc-IDL); and 4) Cos-IDL for TLRSR with a regularized neighborhood constructed by cosine distance and IDL (TLRSR-Cos-IDL).

Fig. 18 shows the accuracies obtained by different variants for the three datasets. By comparison, TLRSR-NRN has the worst performance in different datasets due to the lack of neighborhood regularization and the lack of assistance of the incremental dictionary. Compared with TLRSR-NRN, the accuracies obtained by TLRSR-NRN-IDL have been greatly improved by approximately 5%, 4%, and 7% for the three

datasets. TLRSR-Euc-IDL and TLRSR-Cos-IDL achieve better performance due to the combination of an incremental dictionary and neighborhood regularization. For the proposed method, the highest OA, AA, and κ are obtained since it jointly considers the two distance metrics and incremental dictionary. In general, both neighborhood regularization and incremental dictionary have significant effects on classification performance, which once again proves the effectiveness of the proposed method.

C. Complexity Analysis

There are four variables in the objective equation of TLRSR-IDL that need to be optimized, which are \mathcal{X}_1 , \mathcal{X} , \mathcal{X}_2 , and \mathcal{E} . As shown in Algorithm 2, we iteratively update each variable. First, when the variable \mathcal{X}_1 is updated, it is necessary to perform I_3 times of singular value decomposition (SVD) on matrices of size $r_D \times I_2$. The corresponding computational complexity is $\mathcal{O}(r_D I_2 I_3)$, where $r_D = \text{rank}_t(\mathcal{D})$. Second, in the process of updating the variable \mathcal{X} , the computational complexity brought by the tensor-product operation is $\mathcal{O}(r_D(I_1 + I_2)I_3 \log(I_3) + r_D I_1 I_2 I_3)$. Third, the computational complexity of updating variables \mathcal{X}_2 and \mathcal{E} is $\mathcal{O}(r_D I_2 I_3)$ and $\mathcal{O}(r_D I_1 I_2 I_3)$, respectively. Finally, denote the max-iter as T , and then, the total computational complexity is $\mathcal{O}(Tr_D(I_1 + I_2)I_3 \log(I_3) + Tr_D I_1 I_2 I_3)$.

D. Feature Representation Analysis

As shown in Fig. 19, we verify the feature expression ability of the TLRSR-IDL model by taking the Indian Pines dataset as an example. In order to obtain the feature maps of different classes, we first solve the TLRSR-IDL model based on dictionaries with different IDL iterations to obtain the corresponding representation coefficients. Then, HSI is restored based on each subdictionary. Finally, the restored HSI is compressed by principal component analysis (PCA), and the top three principal components are selected to composite the feature maps.

Experimental results show that the feature expression ability of the TLRSR model without IDL iteration is the worst. As shown in Fig. 19(a), it is difficult to discriminate between different land cover types based on the corresponding feature maps, especially for *Corn-mintill* and *Grass/trees*. After one iteration of IDL, as shown in Fig. 19(b), the feature expression ability of the model has been significantly improved, but the outlines of some classes are still blurred, such as *Grass/trees*. Subsequently, as shown in Fig. 19(c) and (d), with the increase of IDL iterations, the overall purity of the obtained feature maps is continuously improved, and the contours are gradually sharper. On the one hand, this experiment proves that the proposed TLRSR-IDL method can effectively express the spectral-spatial features of HSI. On the other hand, it is intuitively verified that the IDL model can iteratively enhance the representation ability of dictionary.

E. Differences Between Deep Learning and the Proposed Method

It is worth noting that deep learning methods are fundamentally different from the proposed TLRSR-IDL method.

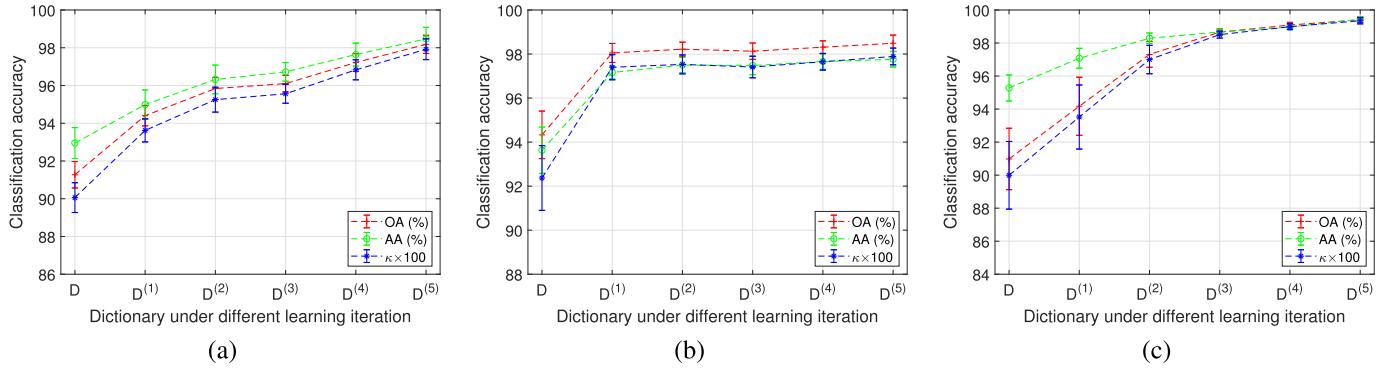


Fig. 17. Effect of IDL under different learning iterations on classification accuracy. (a) Indian Pines. (b) Pavia University. (c) Salinas.

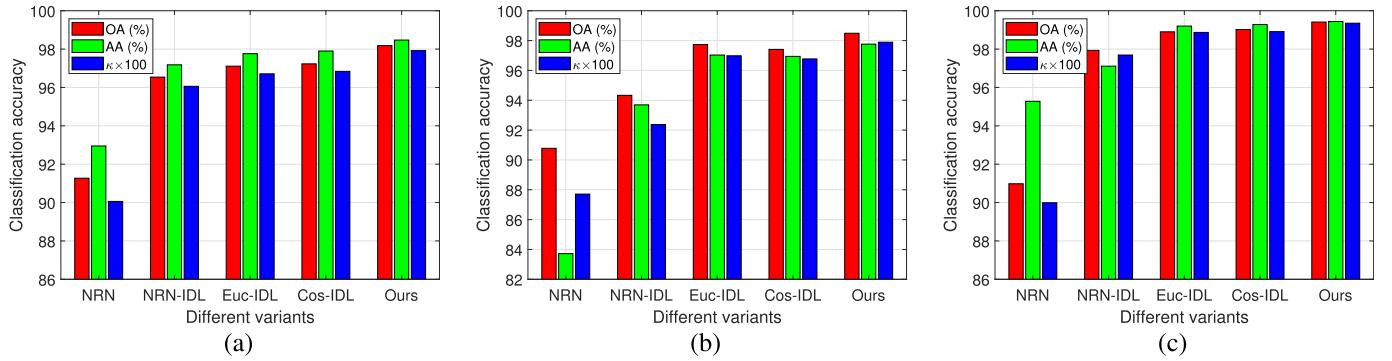


Fig. 18. Comparison of different variants of TLRSR-IDL in terms of classification accuracy. (a) Indian Pines. (b) Pavia University. (c) Salinas. NRN for TLRSR with nonregularized neighborhood, NRN-IDL for TLRSR with nonregularized neighborhood and IDL, Euc-IDL for TLRSR with regularized neighborhood constructed by Euclidean distance and IDL, and Cos-IDL for TLRSR with regularized neighborhood constructed by cosine distance and IDL.

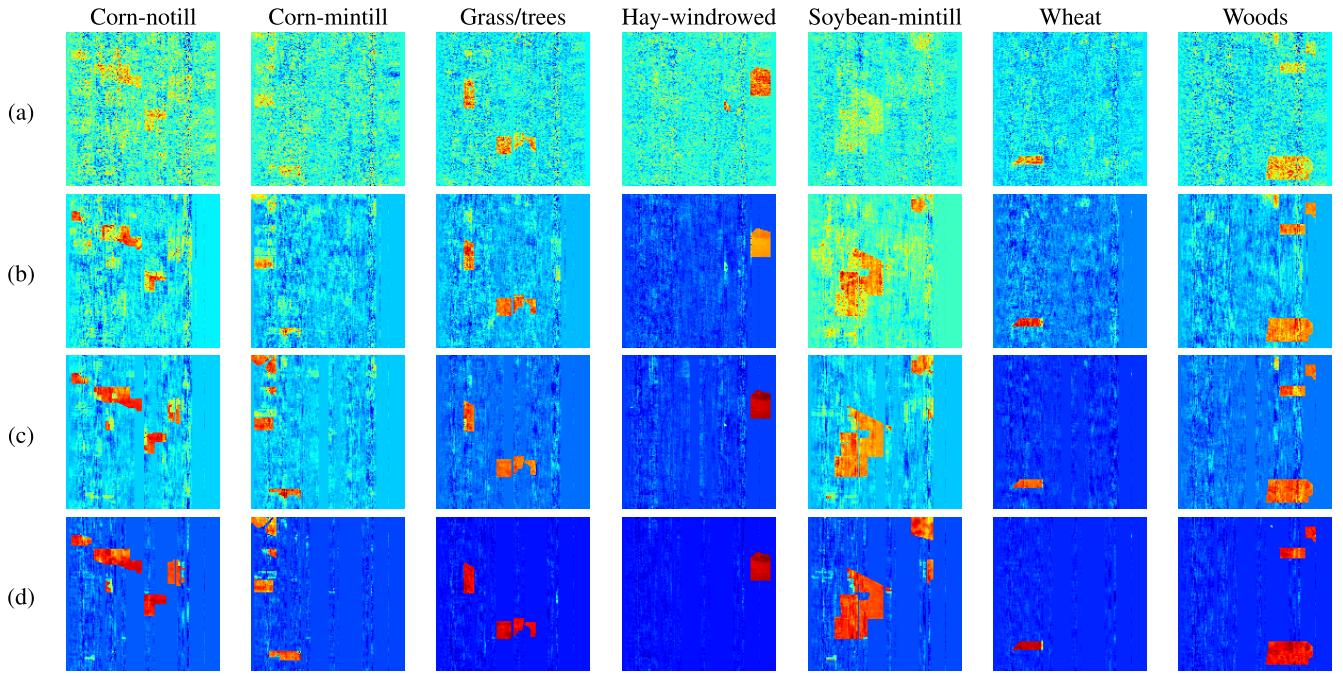


Fig. 19. Feature maps obtained by TLRSR-IDL with a different number of IDL iterations for Indian Pines. (a) Without IDL iteration. (b) With one IDL iteration. (c) With three IDL iterations. (d) With five IDL iterations.

On the one hand, there are differences in the classification principle. The 3-D CNN-related deep learning models rely on end-to-end hierarchical feature learning using operations,

including pooling, convolution, and other tricks, and depend on Softmax for classification, whereas TLRSR-IDL performs classification according to the minimum reconstruction error

calculated by restoring HSI based on each subdictionary and the corresponding coefficient.

On the other hand, the proposed method is more interpretable. Although deep learning provides a very powerful feature learning architecture, the mechanism behind is hard to understand and the “black box” need to open [5]. In contrast, the TLRSR-IDL model is originally proposed with comprehensive theoretical interpretation in terms of formulation derivation, problem definition, optimization, and complexity analysis.

V. CONCLUSION

This article proposed a novel TLRSR-IDL method for HSI classification. In our method, HSI is regarded as a third-order tensor to retain its original 3-D intrinsic structure. The low-rank and sparse constraints make the tensor-based representation effectively maintain global and local data structures. Moreover, an ORRN method is designed for exploiting spectral–spatial information. In particular, an IDL scheme is designed to augment samples and continuously improve the completeness and representation ability of the dictionary. For classification performance, TLRSR obtains an OA of 98.18% for Indian Pines with 0.97%–16.83% improvements, an OA of 98.49% for Pavia University with 1.25%–6.89% improvements, and an OA of 99.41% for Salinas with 0.85%–6.67% improvements. For generalization performance, TLRSR-IDL performs better in most of the cases under different percentages of training samples. The comparison results with deep learning-related methods show that the proposed method has better practicability under small samples. For the IDL performance, the proposed scheme can iteratively enhance the representation ability of the dictionary and greatly improve the classification accuracy by introducing augmented samples. An ablation study by removing IDL and distance metrics verifies the effectiveness of the proposed method.

Although our experimental results are encouraging, further work on additional scenes and comparison methods should be conducted in the future. We are also planning to further study the method from two perspectives: 1) attempt to incorporate other more effective sample augmentation methods, e.g., active learning, and 2) impose an effective structural constraint on TLRSR-IDL to further explore the low-dimensional manifold structure of HSI.

APPENDIX

A. Tensor Operation Definitions and Lemmas

Definition 1 (Tensor Frobenius Norm): For a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, the corresponding tensor Frobenius norm is defined as

$$\|\mathcal{A}\|_F = \left(\sum_{i=1}^{I_1} \sum_{j=1}^{I_2} \sum_{k=1}^{I_3} |\mathcal{A}_{(i,j,k)}|^2 \right)^{1/2}. \quad (15)$$

Definition 2 (Tensor ℓ_1 -Norm): For a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, the ℓ_1 -norm is defined as the sum of absolute values

$$\|\mathcal{A}\|_1 = \sum_{i=1}^{I_1} \sum_{j=1}^{I_2} \sum_{k=1}^{I_3} |\mathcal{A}_{(i,j,k)}|. \quad (16)$$

Definition 3 (Tensor $\ell_{2,1}$ -Norm): For a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, the $\ell_{2,1}$ -norm is defined as

$$\|\mathcal{A}\|_{2,1} = \sum_{j=1}^{I_2} \left(\sum_{i=1}^{I_1} \sum_{k=1}^{I_3} |\mathcal{A}_{(i,j,k)}|^2 \right)^{1/2}. \quad (17)$$

Definition 4 [54] (DFT and IDFT): For a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, let $\overline{\mathcal{A}} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ represent the result of discrete Fourier transformation (DFT) of \mathcal{A} , which is denoted as $\overline{\mathcal{A}} = \text{DFT}(\mathcal{A})$ [i.e., each tube $\overline{\mathcal{A}}(i, j, :)$ of $\overline{\mathcal{A}}$ is obtained by DFT of the corresponding tube $\mathcal{A}(i, j, :)$]. In addition, the inverse DFT (IDFT) operation is denoted as $\mathcal{A} = \text{IDFT}(\overline{\mathcal{A}})$.

Definition 5 (Block Diagonal): Let $\widehat{\mathcal{A}} = \text{bdiag}(\overline{\mathcal{A}})$ represent the result of block diagonal matrix operation on the DFT tensor $\overline{\mathcal{A}} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ along the third dimension

$$\text{bdiag}(\overline{\mathcal{A}}) = \begin{bmatrix} \overline{\mathcal{A}}^{(1)} & & \\ & \overline{\mathcal{A}}^{(2)} & \\ & & \ddots \\ & & & \overline{\mathcal{A}}^{(I_3)} \end{bmatrix} \in \mathbb{C}^{I_1 I_3 \times I_2 I_3}. \quad (18)$$

Definition 6 (Block Circulant): Let $\text{bcirc}(\mathcal{A})$ denote the result of the block circulant matrix operation on tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ along the third dimension.

$$\text{bcirc}(\mathcal{A}) = \begin{bmatrix} \mathcal{A}^{(1)} & \mathcal{A}^{(I_3)} & \dots & \mathcal{A}^{(2)} \\ \mathcal{A}^{(2)} & \mathcal{A}^{(1)} & \dots & \mathcal{A}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{A}^{(I_3)} & \mathcal{A}^{(I_3-1)} & \dots & \mathcal{A}^{(1)} \end{bmatrix} \in \mathbb{R}^{I_1 I_3 \times I_2 I_3}. \quad (19)$$

Definition 7 (Unfold and Fold): For a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, the unfold and fold operations are defined as

$$\begin{aligned} \text{unfold}(\mathcal{A}) &= \begin{bmatrix} \mathcal{A}^{(1)} \\ \mathcal{A}^{(2)} \\ \vdots \\ \mathcal{A}^{(I_3)} \end{bmatrix} \in \mathbb{R}^{I_1 I_3 \times I_2} \\ \text{fold}(\text{unfold}(\mathcal{A})) &= \mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}. \end{aligned} \quad (20)$$

Definition 8 [66] (Tensor Product): For tensors $\mathcal{A} \in \mathbb{R}^{I_1 \times k \times I_3}$ and $\mathcal{B} \in \mathbb{R}^{k \times I_2 \times I_3}$, the tensor product of \mathcal{A} and \mathcal{B} is defined as

$$\mathcal{A} * \mathcal{B} = \text{fold}[\text{bcirc}(\mathcal{A}) \times \text{unfold}(\mathcal{B})] \in \mathbb{R}^{I_1 \times I_2 \times I_3}. \quad (21)$$

Lemma 1 [66]: Assume that $\mathcal{A} \in \mathbb{R}^{I_1 \times k \times I_3}$ and $\mathcal{B} \in \mathbb{R}^{k \times I_2 \times I_3}$ are arbitrary tensors, and $\mathcal{C} = \mathcal{A} * \mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. Then, the following two properties hold.

- 1) $\|\mathcal{A}\|_F^2 = (1/I_3) \|\widehat{\mathcal{A}}\|_F^2$.
- 2) $\mathcal{C} = \mathcal{A} * \mathcal{B}$ and $\widehat{\mathcal{C}} = \widehat{\mathcal{A}} \widehat{\mathcal{B}}$ are equivalent to each other.

Lemma 2 [66] (T-SVD): Assume that $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is a third-order tensor; then, \mathcal{A} can be decomposed into

$$\mathcal{A} = \mathcal{U} * \mathcal{E} * \mathcal{V}^T \quad (22)$$

where $\mathcal{E} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is the f -diagonal tensor and $\mathcal{U} \in \mathbb{R}^{I_1 \times I_1 \times I_3}$ and $\mathcal{V} \in \mathbb{R}^{I_2 \times I_2 \times I_3}$ are orthogonal tensors.

Definition 10 (Tensor Tubal Rank): For a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, the tensor tubal rank is defined as the number of

nonzero singular tubes of \mathcal{E} , denoted as $\text{rank}_t(\mathcal{A})$, where \mathcal{E} is from the T-SVD of tensor \mathcal{A} .

Definition 11 [67] (Tensor Nuclear Norm): It represents tensor \mathcal{A} as $\mathcal{A} = \mathcal{U} * \mathcal{E} * \mathcal{V}^T$ by T-SVD, and then, the tensor nuclear norm of \mathcal{A} is defined as

$$\|\mathcal{A}\|_* = \sum_{i=1}^{\text{rank}_t(\mathcal{A})} \mathcal{E}_{(i,i,1)}. \quad (23)$$

B. ADMM for Solving TLRSR-IDL

1) Step 1: Fix $\mathcal{X}^{(k)}$, $\mathcal{E}^{(k)}$, and \mathcal{X}_2 , and update \mathcal{X}_1

$$\min_{\mathcal{X}_1} \|\mathcal{X}_1\|_* + \frac{\mu}{2} \left\| \mathcal{X}_1 - \mathcal{X}^{(k)} + \frac{\mathcal{Q}_2}{\mu} \right\|_F^2. \quad (24)$$

Equation (24) can be equivalently replaced by

$$\min_{\mathcal{X}_1} \left\| \widehat{\mathcal{X}}_1 \right\|_* + \frac{\mu}{2} \left\| \widehat{\mathcal{X}}_1 - \widehat{\mathcal{X}}^{(k)} + \frac{\widehat{\mathcal{Q}}_2}{\mu} \right\|_F^2. \quad (25)$$

$\widehat{\mathcal{X}}_1$, $\widehat{\mathcal{X}}^{(k)}$, and $\widehat{\mathcal{Q}}_2$ are block diagonal matrices, which can be processed according to each diagonal block

$$\min_{\substack{\mathcal{X}_1^{(i)} \\ i=1,2,\dots,I_3}} \left\| \overline{\mathcal{X}}_1^{(i)} \right\|_* + \frac{\mu}{2} \left\| \overline{\mathcal{X}}_1^{(i)} - \overline{\mathcal{X}}^{(k)(i)} + \frac{\overline{\mathcal{Q}}_2^{(i)}}{\mu} \right\|_F^2 = U S_{\frac{1}{\mu}}(\Sigma) V^T \quad (26)$$

where $U \Sigma V^T$ is the SVD of $\overline{\mathcal{X}}^{(k)(i)} - (\overline{\mathcal{Q}}_2^{(i)}/\mu)$ and $S_a(b) = \text{sgn}(b) \max(|b| - a, 0)$ is the shrinkage operator. After obtaining the solutions of all diagonal blocks, the final result can be obtained by the IDFT operation

$$\mathcal{X}_1 = \text{IDFT}(\overline{\mathcal{X}}_1). \quad (27)$$

2) Step 2: Fix \mathcal{X}_1 , $\mathcal{E}^{(k)}$, and \mathcal{X}_2 , and update $\mathcal{X}^{(k)}$

$$\begin{aligned} \min_{\mathcal{X}} & \langle \mathcal{Q}_1, \mathcal{H} - \mathcal{D}^{(k)} * \mathcal{X}^{(k)} - \mathcal{E}^{(k)} \rangle \\ & + \langle \mathcal{Q}_2, \mathcal{X}_1 - \mathcal{X}^{(k)} \rangle + \langle \mathcal{Q}_3, \mathcal{X}_2 - \mathcal{X}^{(k)} \rangle \\ & + \frac{\mu}{2} \left(\left\| \mathcal{H} - \mathcal{D}^{(k)} * \mathcal{X}^{(k)} - \mathcal{E}^{(k)} \right\|_F^2 \right. \\ & \quad \left. + \left\| \mathcal{X}_1 - \mathcal{X}^{(k)} \right\|_F^2 + \left\| \mathcal{X}_2 - \mathcal{X}^{(k)} \right\|_F^2 \right). \end{aligned} \quad (28)$$

To simplify the expression results, four intermediate variables are introduced

$$\begin{aligned} \mathcal{F}_1 &= \left((\mathcal{D}^{(k)})^T * \mathcal{D}^{(k)} + 2\mathcal{I} \right)^{-1} \\ \mathcal{F}_2 &= (\mathcal{D}^{(k)})^T * \left(\frac{\mathcal{Q}_1}{\mu} - \mathcal{E}^{(k)} + \mathcal{X}^{(k)} \right) \\ \mathcal{F}_3 &= \mathcal{X}_1 + \frac{\mathcal{Q}_3}{\mu} \quad \mathcal{F}_4 = \mathcal{X}_2 + \frac{\mathcal{Q}_3}{\mu} \end{aligned} \quad (29)$$

where $(\mathcal{D}^{(k)})^T$ is the conjugate tensor of $\mathcal{D}^{(k)}$, \mathcal{I} represents the identity tensor, and $\mathcal{X}^{(k)}$ can be updated as

$$\mathcal{X}^{(k)} = \mathcal{F}_1 * (\mathcal{F}_2 + \mathcal{F}_3 + \mathcal{F}_4). \quad (30)$$

3) Step 3: Fix \mathcal{X}_1 , $\mathcal{E}^{(k)}$, and $\mathcal{X}^{(k)}$, and update \mathcal{X}_2

$$\begin{aligned} \mathcal{X}_2 &= \min_{\mathcal{X}_2} \|\mathcal{X}_2\|_1 + \frac{\mu}{2} \left\| \mathcal{X}_2 - \mathcal{X}^{(k)} + \frac{\mathcal{Q}_3}{\mu} \right\|_F^2 \\ &= \Theta_{\frac{\mu}{\mu}} \left(\mathcal{X}^{(k)} - \frac{\mathcal{Q}_3}{\mu} \right) \end{aligned} \quad (31)$$

where $\Theta_a^{(b)} = \max(b-a, 0) + \min(b+a, 0)$ is the elementwise shrinkage operator.

4) Step 4: Fix \mathcal{X}_1 , $\mathcal{X}^{(k)}$, and \mathcal{X}_2 , and update $\mathcal{E}^{(k)}$

$$\min_{\mathcal{E}^{(k)}} \beta \left\| \mathcal{E}^{(k)} \right\|_{2,1} + \frac{\mu}{2} \left\| \mathcal{H} - \mathcal{D}^{(k)} * \mathcal{X}^{(k)} - \mathcal{E}^{(k)} + \frac{\mathcal{Q}_1}{\mu} \right\|_F^2. \quad (32)$$

Let $\mathcal{G} = \mathcal{H} - \mathcal{D}^{(k)} * \mathcal{X}^{(k)} - \mathcal{E}^{(k)} + (\mathcal{Q}_1/\mu)$ and $\mathcal{G}^{(j)}$ represent the j th lateral slices of \mathcal{G} along the second dimension. Then, $(\mathcal{E}^{(k)})^{(j)}$ can be updated as

$$(\mathcal{E}^{(k)})^{(j)} = \begin{cases} \frac{\|\mathcal{G}^{(j)}\|_F - \frac{\beta}{\mu}}{\|\mathcal{G}^{(j)}\|_F} * \mathcal{G}^{(j)}, & \frac{\beta}{\mu} \leq \|\mathcal{G}^{(j)}\|_F \\ 0, & \text{otherwise}. \end{cases} \quad (33)$$

ACKNOWLEDGMENT

The authors would like to thank Prof. D. Landgrebe for making the Airborne Visible/Infrared Imaging Spectrometer Indian Pines hyperspectral dataset available to the community and Prof. P. Gamba for providing the Reflective Optics Spectrograph Imaging System data over Pavia, Italy.

REFERENCES

- [1] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Jan. 2002.
- [2] Q. Tong, Y. Xue, and L. Zhang, "Progress in hyperspectral remote sensing science and technology in China over the past three decades," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 1, pp. 70–91, Jan. 2014.
- [3] L. He, J. Li, C. Liu, and S. Li, "Recent advances on spectral-spatial hyperspectral image classification: An overview and new guidelines," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1579–1597, Mar. 2018.
- [4] P. Ghamisi et al., "New frontiers in spectral-spatial hyperspectral image classification: The latest advances based on mathematical morphology, Markov random fields, segmentation, sparse representation, and deep learning," *IEEE Geosci. Remote Sens. Mag.*, vol. 6, no. 3, pp. 10–43, Sep. 2018.
- [5] D. F. Hong et al., "Interpretable hyperspectral artificial intelligence: When nonconvex modeling meets hyperspectral remote sensing," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 2, pp. 52–87, Jun. 2021.
- [6] L. Zhang and L. Zhang, "Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 270–294, Jun. 2022.
- [7] W. Li and Q. Du, "A survey on representation-based classification and detection in hyperspectral remote sensing imagery," *Pattern Recognit. Lett.*, vol. 83, pp. 115–123, Nov. 2016.
- [8] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, Oct. 2011.
- [9] L. Wei and D. Qian, "Collaborative representation for hyperspectral anomaly detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1463–1474, Mar. 2015.
- [10] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.
- [11] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification via kernel sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 217–231, Jan. 2013.

- [12] M. Cui and S. Prasad, "Class-dependent sparse representation classifier for robust hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2683–2695, May 2015.
- [13] Z. Xue, J. Li, L. Cheng, and P. Du, "Spectral–spatial classification of hyperspectral data via morphological component analysis-based image separation," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 70–84, Jan. 2015.
- [14] Z. Xue, P. Du, J. Li, and H. Su, "Simultaneous sparse graph embedding for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 11, pp. 6114–6133, Jun. 2015.
- [15] P. Du, Z. Xue, J. Li, and A. Plaza, "Learning discriminative sparse representations for hyperspectral image classification," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 6, pp. 1089–1104, Sep. 2015.
- [16] Z. Xue, P. Du, J. Li, and H. Su, "Sparse graph regularization for hyperspectral remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 4, pp. 2351–2366, Apr. 2017.
- [17] Z. Xue, S. Yang, and L. Zhang, "Weighted sparse graph regularization for spectral–spatial classification of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 9, pp. 1630–1634, Sep. 2021.
- [18] J. Peng, W. Sun, and Q. Du, "Self-paced joint sparse representation for the classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1183–1194, Feb. 2019.
- [19] M. Wang, Z. Jia, J. Luo, M. Chen, S. Wang, and Z. Ye, "A hyperspectral image classification method based on weight wavelet kernel joint sparse representation ensemble and β -whale optimization algorithm," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2535–2550, 2021.
- [20] J. Peng, L. Li, and Y. Y. Tang, "Maximum likelihood estimation-based joint sparse representation for the classification of hyperspectral remote sensing images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 6, pp. 1790–1802, Jun. 2019.
- [21] Y. Duan, H. Huang, and Y. Tang, "Local constraint-based sparse manifold hypergraph learning for dimensionality reduction of hyperspectral image," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 613–628, Jan. 2021.
- [22] H. Su, B. Zhao, Q. Du, P. Du, and Z. Xue, "Multifeature dictionary learning for collaborative representation classification of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2467–2484, Apr. 2018.
- [23] M. Liang, L. Jiao, and C. Xu, "Deep feature-based multitask joint sparse representation for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 9, pp. 1578–1582, Sep. 2020.
- [24] S. Yang, J. Hou, Y. Jia, S. Mei, and Q. Du, "Hyperspectral image classification via sparse representation with incremental dictionaries," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 9, pp. 1598–1602, Sep. 2020.
- [25] Q. Liu, Z. Wu, L. Sun, Y. Xu, L. Du, and Z. Wei, "Kernel low-rank representation based on local similarity for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 6, pp. 1920–1932, Jun. 2019.
- [26] H. Su, W. Yao, Z. Wu, P. Zheng, and Q. Du, "Kernel low-rank representation with elastic net for China coastal wetland land cover classification using GF-5 hyperspectral imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 171, pp. 238–252, Jan. 2021.
- [27] S. Yang, J. Hou, Y. Jia, S. Mei, and Q. Du, "Superpixel-guided discriminative low-rank representation of hyperspectral images for classification," *IEEE Trans. Image Process.*, vol. 30, pp. 8823–8835, 2021.
- [28] S. Sarkar and R. R. Sahay, "A non-local superpatch-based algorithm exploiting low rank prior for restoration of hyperspectral images," *IEEE Trans. Image Process.*, vol. 30, pp. 6335–6348, 2021.
- [29] H. Zhai, H. Zhang, L. Zhang, and P. Li, "Laplacian-regularized low-rank subspace clustering for hyperspectral image band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1723–1740, Mar. 2019.
- [30] J. An, X. Zhang, H. Zhou, and L. Jiao, "Tensor-based low-rank graph with multimanifold regularization for dimensionality reduction of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4731–4746, Aug. 2018.
- [31] Y. Wang et al., "Self-supervised low-rank representation (SSLRR) for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 5658–5672, Oct. 2018.
- [32] Z. Feng, S. Yang, M. Wang, and L. Jiao, "Learning dual geometric low-rank structure for semisupervised hyperspectral image classification," *IEEE Trans. Cybern.*, vol. 51, no. 1, pp. 346–358, Jan. 2021.
- [33] J. T. Peng et al., "Low-rank and sparse representation for hyperspectral image processing: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 1, pp. 10–43, Mar. 2022.
- [34] W. Li, J. Liu, and Q. Du, "Sparse and low-rank graph for discriminant analysis of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 4094–4105, Jul. 2016.
- [35] S. Mei, J. Hou, J. Chen, L.-P. Chau, and Q. Du, "Simultaneous spatial and spectral low-rank representation of hyperspectral images for classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2872–2886, May 2018.
- [36] Q. Wang, X. He, and X. Li, "Locality and structure regularized low rank representation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 911–923, Feb. 2019.
- [37] Y. Ding, Y. Chong, and S. Pan, "Sparse and low-rank representation with key connectivity for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5609–5622, Oct. 2020.
- [38] 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 Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6085–6098, Nov. 2017.
- [39] W. Sun, G. Yang, B. Du, L. Zhang, and L. Zhang, "A sparse and low-rank near-isometric linear embedding method for feature extraction in hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 4032–4046, Jul. 2017.
- [40] B. Rasti, P. Ghamisi, J. Plaza, and A. Plaza, "Fusion of hyperspectral and LiDAR data using sparse and low-rank component analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6354–6365, Nov. 2017.
- [41] W. Sun, J. Peng, G. Yang, and Q. Du, "Fast and latent low-rank subspace clustering for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 3906–3915, Jun. 2020.
- [42] L. Gao, D. Hong, J. Yao, B. Zhang, P. Gamba, and J. Chanussot, "Spectral superresolution of multispectral imagery with joint sparse and low-rank learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2269–2280, Mar. 2021.
- [43] J. Xue, Y.-Q. Zhao, Y. Bu, W. Liao, J. C.-W. Chan, and W. Philips, "Spatial-spectral structured sparse low-rank representation for hyperspectral image super-resolution," *IEEE Trans. Image Process.*, vol. 30, pp. 3084–3097, 2021.
- [44] Y. Xu, Z. Wu, J. Li, A. Plaza, and Z. Wei, "Anomaly detection in hyperspectral images based on low-rank and sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 1990–2000, Apr. 2016.
- [45] L. Zhuang, L. Gao, B. Zhang, X. Fu, and J. M. Bioucas-Dias, "Hyperspectral image denoising and anomaly detection based on low-rank and sparse representations," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5500117.
- [46] S. Du, Y. Shi, G. Shan, W. Wang, and Y. Ma, "Tensor low-rank sparse representation for tensor subspace learning," *Neurocomputing*, vol. 440, pp. 351–364, Jun. 2021.
- [47] J. Xue, Y. Zhao, S. Huang, W. Liao, J. C.-W. Chan, and S. G. Kong, "Multilayer sparsity-based tensor decomposition for low-rank tensor completion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 6916–6930, Nov. 2022.
- [48] J. Xue, Y. Zhao, Y. Bu, J. C.-W. Chan, and S. G. Kong, "When Laplacian scale mixture meets three-layer transform: A parametric tensor sparsity for tensor completion," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13887–13901, Dec. 2022.
- [49] R. Dian, S. Li, and L. Fang, "Learning a low tensor-train rank representation for hyperspectral image super-resolution," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 9, pp. 2672–2683, Sep. 2019.
- [50] Y. Bu et al., "Hyperspectral and multispectral image fusion via graph Laplacian-guided coupled tensor decomposition," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 648–662, Jan. 2021.
- [51] J. Xue, Y. Zhao, W. Liao, and J. C.-W. Chan, "Nonlocal low-rank regularized tensor decomposition for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 5174–5189, Jul. 2019.
- [52] H. Zeng, X. Xie, H. Cui, H. Yin, and J. Ning, "Hyperspectral image restoration via global L_{1-2} spatial–spectral total variation regularized local low-rank tensor recovery," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3309–3325, Apr. 2021.
- [53] P. Zheng, H. Su, and Q. Du, "Sparse and low-rank constrained tensor factorization for hyperspectral image unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1754–1767, 2021.
- [54] P. Zhou, C. Lu, J. Feng, Z. Lin, and S. Yan, "Tensor low-rank representation for data recovery and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1718–1732, May 2021.

- [55] Z. Xue and X. Nie, "Low-rank and sparse representation with adaptive neighborhood regularization for hyperspectral image classification," *J. Geodesy Geoinf. Sci.*, vol. 5, no. 1, pp. 73–90, Mar. 2022.
- [56] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jul. 2011.
- [57] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.
- [58] G. Camps-Valls, L. Gomez-Chova, J. Munoz-Mari, J. Vila-Frances, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, Jan. 2006.
- [59] J. Jiang, C. Chen, Y. Yu, X. Jiang, and J. Ma, "Spatial-aware collaborative representation for hyperspectral remote sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 3, pp. 404–408, Mar. 2017.
- [60] W. Li, Y. Zhang, N. Liu, Q. Du, and R. Tao, "Structure-aware collaborative representation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 7246–7261, Sep. 2019.
- [61] J. Liu, Z. Wu, L. Xiao, J. Sun, and H. Yan, "Generalized tensor regression for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1244–1258, Feb. 2020.
- [62] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.
- [63] M. E. Paoletti, J. M. Haut, R. Fernandez-Beltran, J. Plaza, A. J. Plaza, and F. Pla, "Deep pyramidal residual networks for spectral–spatial hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 740–754, Feb. 2019.
- [64] Y. Feng, J. Zheng, M. Qin, C. Bai, and J. Zhang, "3D octave and 2D vanilla mixed convolutional neural network for hyperspectral image classification with limited samples," *Remote Sens.*, vol. 13, no. 21, p. 4407, Nov. 2021.
- [65] L. Sun, G. Zhao, Y. Zheng, and Z. Wu, "Spectral–spatial feature tokenization transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5522214.
- [66] M. E. Kilmer and C. D. Martin, "Factorization strategies for third-order tensors," *Linear Algebra Appl.*, vol. 435, no. 3, pp. 641–658, 2011.
- [67] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis with a new tensor nuclear norm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 925–938, Jan. 2020.



Zhaohui Xue (Member, IEEE) received the B.S. degree in geomatics engineering from Shandong Agricultural University, Tai'an, China, in 2009, the M.E. degree in remote sensing from the China University of Mining and Technology, Beijing, China, in 2012, and the Ph.D. degree in cartography and geographic information system from Nanjing University, Nanjing, China, in 2015.

He is currently a Youth Professor (Ph.D. supervisor) with the School of Earth Sciences and Engineering, Hohai University, Nanjing. He has authored more than 50 scientific articles, including more than 30 SCI articles. His research interests include hyperspectral image classification, time-series image analysis, pattern recognition, and machine learning.

Dr. Xue was a recipient of the National Scholarship for Doctoral Graduate Students granted by the Ministry of Education of the People's Republic of China in 2014. He was awarded the Best Reviewer for the IEEE Geoscience and Remote Sensing Society. He is an Editorial Board Member of *National Remote Sensing Bulletin* for the term 2020–2024. He has been a Reviewer for more than ten famous remote sensing journals, including *IEEE Transactions on GEOSCIENCE AND REMOTE SENSING* and *ISPRS Journal of Photogrammetry and Remote Sensing*.



Xiangyu Nie received the B.S. degree in geomatics engineering from the Suzhou University of Science and Technology, Suzhou, China, in 2019, and the M.E. degree in remote sensing from Hohai University, Nanjing, China, in 2022.

His research interests include low-rank and sparse representation for hyperspectral image classification.



Mengxue Zhang received the B.S. and M.E. degrees in geomatics engineering from Hohai University, Nanjing, China, in 2018 and 2021, respectively. He is currently pursuing the Ph.D. degree in electrical engineering with the University of València, València, Spain.

His research interests include hyperspectral image classification and hybrid machine learning for drought and heatwave detection.