An Efficient and Adaptive Reconstructive Homogeneous Block-Based Local Tensor Robust PCA for Feature Extraction of Hyperspectral Images

Longxia Qian, Xianyue Wang, Mei Hong, Hongrui Wang, and Yongchui Zhang

***Abstract*—Model-driven tensor robust principal component analysis (TRPCA) has been widely applied to feature extraction of hyperspectral images (HSIs) and successfully protected two-dimensional spectral contextual information. Nevertheless, the current TRPCA-based feature extraction methods still destroy the underlying spectral and spatial–spectral joint contextual features. Moreover, these global iterative algorithms commonly ignore the heterogeneity of different real-world regions, increase the calculation burden, and improve practice operating time. To solve these issues, an efficient reconstructive homogeneous block-based local TRPCA is proposed for low-rank feature extraction, composed of a homogeneous block rebuilder and a local TRPCA low-rank feature extractor. The proposed local TRPCA is a novel data-model-driven algorithm depending on the data regulation. It remains the primary spatial and spectral contextual information and extracts the underlying homogeneity and heterogeneity characteristics of spatial, spectral, and spatial–spectral joint variables, which provides more essential features for further research than other model-driven TRPCA models. Furthermore, our local TRPCA feature extractor is an elaborate divide-and-rule model that executes on each homogeneous data block to extract low-rank features adaptively, remarkably decreasing computing cost and time. Experimental results on six hyperspectral datasets demonstrate that the proposed local TRPCA is more adaptive to HSIs and outperforms other state-of-the-art TRPCA-based feature extraction algorithms.**

***Index Terms*—Feature extraction, hybrid spatial–spectral frontal slice, hyperspectral image (HSI), reconstructive homogeneous block, tensor robust principal component analysis (TRPCA).**

Manuscript received 27 December 2023; accepted 13 January 2024. Date of publication 19 January 2024; date of current version 12 February 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 52279005, in part by the Open Project of the High Impact Weather Key Laboratory of CMA, in part by the Humanities and Social Science Fund of Ministry of Education under Grant 23YJAZH111, and in part by the Natural Science Foundation of Hunan Province under Grant 2023JJ10054. *(Corresponding authors: Mei Hong; Hongrui Wang.)*

Longxia Qian is with the School of Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China, and also with the High Impact Weather Key Laboratory of CMA, Changsha 410073, China. (e-mail: qianlongxia@njupt.edu.cn).

Xianyue Wang is with the School of Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China. (e-mail: wxy278499177@163.com).

Mei Hong is with the College of Meteorology and Oceanography, National University of Defense Technology, Changsha 410073, China, and also with the High Impact Weather Key Laboratory of CMA, Changsha 410073, China (e-mail: hongmflower@163.com).

Yongchui Zhang is with the College of Meteorology and Oceanography, National University of Defense Technology, Changsha 410073, China. (e-mail: zyc@nudt.edu.cn).

Hongrui Wang is with the College of Water Science, Beijing Normal University, Beijing 100875, China, and also with the Beijing Key Laboratory of Urban Hydrological Cycle and Sponge City Technology, Beijing 100875, China. (e-mail: henrywang@bnu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2024.3355931

I. Introduction

When advanced technical spaceborne or airborne hyperspectral sensors are used, hundreds of visible to infrared narrow spectral bands provide detailed and identified information about the characteristics of objectives within a practical scene [1], [2], [3]. In other words, the acquired three-dimensional (3-D) hyperspectral image (HSI), numerous observations on hundreds of 1-D spectral channels reflected from a specific 2-D real-world spatial region, can record the temporary state of ground-cover objects within a particular area. With the continuous development of related hyperspectral remote sensing technologies, researchers tend to utilize available HSIs in various domains to identify different unknown substances. Meanwhile, the quantity and quality of HSIs are increasing substantially, providing more and more support for concerned studies.

However, equipment noises, system errors, or random errors may probably generate corrupted or outlying observations in recorded data cubes [4], [5]. These observations that hardly degrade the performance of further classification or other application tasks make it difficult to directly apply HSIs to many current research fields, especially for numerous deep-learning methods sensitive to noise [6]. To solve this issue, a bunch of HSI classification models have been proposed by incorporating advanced classification processing with a specific low-rank extractor manner to remove disturbing components and enhance object recognition performance [7]. The adopted low-rank recovery techniques could cover an extensive range of processing concepts and can be interspersed at any point throughout the model. In [8], principal component analysis (PCA) is employed to wipe off unimportant components, followed by a deep-learning-based 3-D hierarchical convolution network. Likewise, a unified network was proposed for HSI classification, integrating with a well-designed loss function to reduce noises and recognize the category of land-cover objects simultaneously [9].

Generally speaking, there are a variety of low-rank extracting techniques suitable for high-dimensional images, ranging from extensions of PCA [10], [11], wavelet transformation [12], and total variation [13] to neural networks [14]. Compared with the later data-driven networks, the prominent advantages of less occupied memory and faster operating speed make the former model-driven methods more applicable to vast hyperspectral

datasets [7]. However, several problems still need to be solved in these model-driven models. The damage to 2-D spatial structure leads to the loss of essential pixel context information, which is precisely the salient one of these problems. It may gravely affect the results of subsequent model exhibition and data analysis, especially for HSIs with more regional spatial similarities than the traditional RGB remote sensing images [15]. Fortunately, some tensor-based low-rank extracting models remedy the deficiencies promptly.

Tensor robust PCA (TRPCA) seems valid to extract low-rank elements while maintaining primary data structure [16]. Most common TRPCA methods tend to integrate tensor representation with 2-D robust PCA (RPCA), recovering the low-rank component from the original noisy multidimensional array without sparse noises. The typical TRPCA low-rank recovery techniques can be roughly divided into two main categories, which are Tucker [17] or CP [18] decomposition-based TRPCA and tensor singular value decomposition (T-SVD) based TRPCA [13]. Nevertheless, the former TRPCA algorithms are less adaptive to HSIs than T-SVD-based algorithms [19]. Tensor decomposition-based TRPCA needs to unfold the 3-D data cube along each dimension to compute the corresponding low rankness, which still disrupts the inherent 2-D spatial contextual information and impedes HSI classification. It also requires finding a congruent numerical value for the regularization parameter in the minimization objective along three dimensions, which is pretty intractable.

To overcome these shortcomings, many TRPCA models employ the T-SVD algorithm to extract primary components. They employ discrete Fourier transformation or inverse version to minimize the created tensor nuclear norm of objective functions [20], [21]. Such methods usually distinguish low-rank tensor data from sparse noises by arranging the data slices in a block-circulant manner without any unfolding processing and congruent regularization parameter value of three dimensions. In other words, T-SVD-based TRPCA is expected to be a helpful denoising tool for HSIs because of the preservation of more salutary spatial contextual information and the utilization of more simple parameter conditions than Tucker or CP decomposition-based TRPCA. Specifically, a 3-D HSI can be denoted as $\mathbf{H} \in \mathbb{R}^{r\times c\times b}$, where parameters r, c, and b are the sizes of spatial rows, spatial columns, and spectral bands, respectively. Since a hyperspectral dataset $\mathbf{H}$ can be broken down into a low-rank tensor component $\mathbf{L} \in \mathbb{R}^{r\times c\times b}$ and a noisy tensor component $\mathbf{N} \in \mathbb{R}^{r\times c\times b}$, TRPCA algorithms tend to acquire a low-rank recovery $\mathbf{L}$ from $\mathbf{H}$ without grossly corrupted and outlier observations holding in $\mathbf{N}$. Current T-SVD-derived TRPCA always introduces a global iterative strategy to achieve the objective minimalization function as

$$\min_{\mathbf{L},\mathbf{N}}(\|\mathbf{L}\|_{\bullet} + \gamma\|\mathbf{N}\|_{\odot}), \text{s.t. } \mathbf{H} = \mathbf{L} + \mathbf{N} \tag{1}$$

where γ is the only adopted regularization parameter, $\|\mathbf{L}\|_{\bullet}$ denotes a specific tensor norm to acquire recovery guarantees for $\mathbf{L}$ by using a tight convex or nonconvex surrogate, and $\|\mathbf{N}\|_{\odot}$ stands for a norm to ensure the sparsity of $\mathbf{N}$. Most TRPCA algorithms mainly differ in choosing distinct tensor norms on $\mathbf{L}$ or $\mathbf{N}$.

For instance, multilinear TRPCA (MLTRPCA) employs multilinear rank-based total tensor unclear norm $\|\mathbf{L}\|_s$ to replace $\|\mathbf{L}\|_{\bullet}$ and $\ell_{1,1,2}$-norm $\|\mathbf{N}\|_{1,1,2}$ instead of $\|\mathbf{N}\|_{\odot}$, executing completion and denoising application for multilinear images [22]. Besides, Lu et al. [23] proposed a novel TRPCA (NTRPCA) equipped with a new tensor unclear norm $\|\mathbf{N}\|_1$ and averaging tensor norm $\|\mathbf{L}\|_a$ to solve the designed convex problem. It aims to provide theoretical and practical guarantees for extracting sufficient underlying low-rank features. In addition, enhanced TRPCA (ETRPCA) employs $\|\mathbf{N}\|_1$ and further treats singular values differently with a weighted Schatten p norm $\|\mathbf{L}\|_{\omega}^{p}$ [24]. As a result, it breaks through the limitation that treats all tensor singular values equally and well preserves the prominent information of the tensor-structured datasets. In [21], nonconvex TRPCA (NNTRPCA) integrates nonlocal self-similarity with $\|\mathbf{N}\|_1$ and a nonconvex formulation tensor adjustable logarithmic norm $\|\mathbf{L}\|_L$ to acquire the detailed edges and texture characteristics of noisy high-order images. Analogously, an improved tensor nuclear norm with a core matrix, improved TRPCA (ITRPCA), has been developed to acquire greater accuracy and faster low-rank tensor approximation [25]. Although several differences exist among these T-SVD-derived TRPCA models, the constituents of the corresponding objective functions are the same as the standard form given in (1). It should be noted that these model-driven T-SVD TRPCA algorithms are not quite adaptive to the vast high-dimensional HSIs because their original modeling purposes are to remove noises existing within grayscale images, RGB images, or even videos.

Some researchers try to apply the T-SVD-based TRPCA algorithms appropriately to hyperspectral datasets. They mainly choose different kinds of slices [26], change the types of norms [27], add parameters to original objective functions, or employ additional preprocessing or postprocessing steps [28]. Nevertheless, these updated TRPCA low-rank extracting techniques destroy the underlying spectral contextual information of HSIs, which may directly result in unsatisfactory classification performance as the homogeneity and heterogeneity information are insufficiently exploited. Similarly, the same outcomes may probably occur in the spatial and spatial–spectral joint dimensions for abandoning the hyperspectral imaging principle and data regulation [29]. Moreover, all these TRPCA algorithms are globally applied to the whole pixel space with the same regularization parameter value. It commonly ignores the distinct heterogeneity of different real-world regions, increases the calculation burden, and improves practice operating time.

To solve these issues, the article presents a localized data-model-driven TRPCA feature-extracting model composed of a homogeneous block rebuilder and a local TRPCA feature extractor. Specifically, it is a superpixelwise reconstructed homogeneous block-based TRPCA that is efficient, effective, and adaptive to hyperspectral datasets, referred to as SPRB-TRPCA [30]. The primary contributions of the designed model are demonstrated as follows.

1) The proposed local TRPCA is a novel data-model-driven feature-extracting algorithm. It remains the primary spatial and spectral contextual information and then extracts the underlying homogeneity and heterogeneity information of spatial, spectral, and spatial–spectral joint

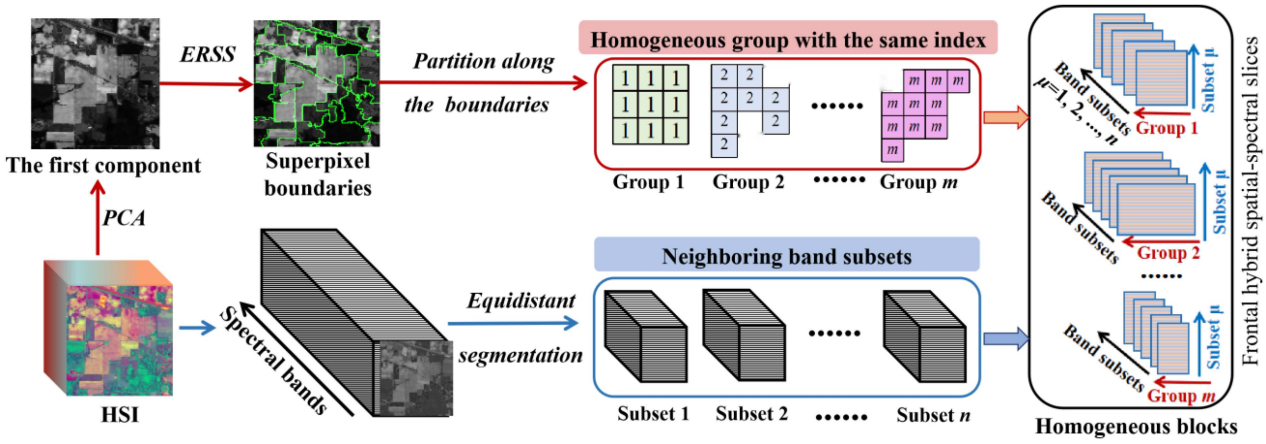

Fig. 1. Homogeneous block rebuilder illustration. First, we utilize ERSS to divide all spatial pixels into multiple heterogeneous superpixels where homoregional pixels are considered to be homogeneous and have the same group index. Meanwhile, all spectral bands are equidistantly partitioned into multiple neighboring subsets. Finally, all frontal hybrid spatial–spectral slices are arranged along heterogeneous spectral subsets, where each slice contains homogeneous pixels as column variables and bands within the same group as row variables.

variables. Considering the hyperspectral data regulation that there exist prominent similarities among regional pixels or neighboring spectral bands, we provide more essential features for further research than other model-driven TRPCA models.

2) Based on the adopted data regulations of spatial and spectral dimensions, SPRB-TRPCA establishes some homogeneous blocks. These blocks make it possible to find low-rank components existing in both spatial and spectral dimensions. Since heterogeneous regions of HSIs contain varying noise levels, informative degrees, and sizes, our local TRPCA can independently find the optimal solutions to objective functions in distinct subregions with different parameter values. From the experimental results, such processing successfully extracts more essential and high-quality features from the vast and high-dimensional datasets.
3) The proposed local TRPCA is conducted on each local dataset instead of the global data cube in an elaborately divide-and-rule manner. Since we divided the whole data cube into several small data blocks, the computing cost and time are significantly decreased. The efficient and effective results reflect that the localization process makes SPRB-TRPCA more adaptive to the vast hyperspectral datasets.

The rest of this article is organized as follows. The proposed methodology with detailed explanations is presented in Section II. Section III discusses thorough results on six hyperspectral datasets, such as parameter sensitivity, remaining features' quality, and classification performance. Section IV provides the discussion. Finally, Section V concludes this article, respectively.

II. Proposed Methodology

A. Preliminaries

In this article, bold straight letters denote the 3-D tensors, bold capital italic letters indicate the 2-D matrices, bold lowercase italic letters denote the vectors, and italic letters denote the scalars. Given a 3-D HSI tensor $\mathbf{H} \in \mathbb{R}^{r\times c\times b}$ with the entries $h_{i,j,t}(i=1,2,\ldots,r; j=1,2,\ldots,c; t=1,2,\ldots,b)$. In particular, $\boldsymbol{H}^{(t)}$ represents its frontal slices. The Frobenius norm on $\mathbf{H}$ is $\|\mathbf{H}\|_F = \sqrt{\sum_{i,j,t}(h_{i,j,t})^2}$. Since SPRB-TRPCA comprises a homogeneous block rebuilder (Fig. 1) and a local TRPCA low-rank feature extractor (Fig. 2), the following subsections will briefly introduce these two steps.

B. Homogeneous Block Rebuilder

Most current T-SVD-derived TRPCA algorithms treat each frontal slice of the input HSI independently, which protects the structure of 2-D spatial pixels but destroys the underlying spectral and spatial–spectral joint contextual information. It may lead to unsatisfactory denoising and classification performance because the homogeneity and heterogeneity information are not sufficiently exploited. For one thing, the exploitation of spatial and spectral homogeneity and heterogeneity information can promote the efficiency of further analysis. For another, it also helps to remove noises within different spatial and spectral regions. Therefore, we intend to extract more spatial, spectral, and spatial–spectral joint contextual characteristics to provide a basis for subsequent local denoising and further objective recognition.

First, we employ entropy rate superpixel segmentation (ERSS) [30] on the first principal component of spectral variables to divide each spatial slice into m varying subsets by visually recognizing the boundaries among different objects. As a result, the whole 2-D ground pixels are divided into different connected heterogeneous pixel clusters, and each group is composed of regional homogeneous spatial samples. To better demonstrate the segmentation results, the pixels within each homogeneous cluster are assigned the same labeled number. Then, we denote vector $\boldsymbol{p}_\nu$ as a spatial pixel group composed of the pixels assigned the same value of v and scalar m_ν as the size (i.e., quantity) of spatial pixels within the obtained cluster $\boldsymbol{p}_\nu$. The heterogeneity segmentation of spatial pixels can be given

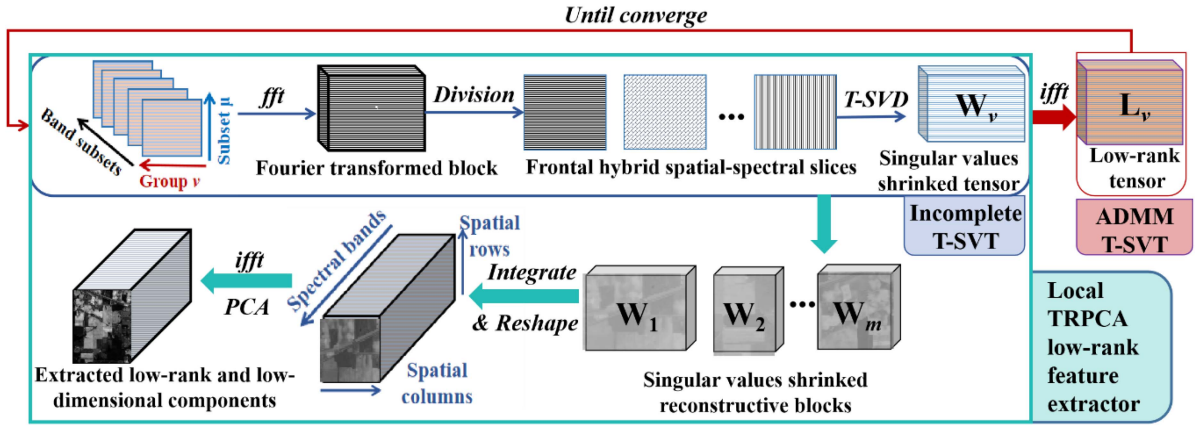

Fig. 2. Local TRPCA low-rank feature extractor illustration. First, the incomplete T-SVT is produced on each homogeneous block by performing T-SVD to shrink singular values on all frontal hybrid spatial–spectral slices after discrete Fourier transformation. Meanwhile, the ADMM is conducted to generate low-rank tensors until it satisfies certain convergence conditions. Subsequently, the inversed results are delivered by inverse discrete Fourier transformation, following the second block reconstruction that reshapes the partitioned bands into one group. Finally, PCA is utilized to extract primary low-rank components.

by

$$\boldsymbol{p}_\nu = \underset{h(i)=v}{\cup} i \in \mathbb{R}^{m_v}, \text{s.t.} \underset{\nu=1}{\overset{m}{\cup}} \boldsymbol{p}_\nu \in \mathbb{R}^{rc}, \sum_{v=1}^{m} m_v = rc \quad (2)$$

where $h(i)$ denotes the resulting group number v of the ith pixel obtained by ERSS. To retain more spatial–spectral contextual information for the subsequent homogeneous block rebuilder processing, the pixel grouping results are the same for all spatial slices $\boldsymbol{H}^{(t)}(t = 1, 2, \ldots, b)$ of the original HSI data cube. After that, the homoclustered pixels hold approximately the same degree of sparse noise and retain similar spatial information. Then, an equidistant partition technique is performed on the third band dimension to roughly obtain homogeneous bands, which is another crucial step for naturally creating hybrid spatial–spectral slices.

Suppose $[\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_b] = \text{unfold}(\mathbf{H})$ is the transformation of $\mathbf{H} \in \mathbb{R}^{r\times c\times b}$, which is acquired by unfolding the 3-D cube into a series of 2-D spectral band vectors $\boldsymbol{\beta}_i(i = 1, 2, \ldots, b)$ along the spectral dimension. The equidistant partition on $\mathbf{H}$ results in n adjacent band clusters $\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_n$, where the μth band matrix $\boldsymbol{B}_\mu(\mu = 1, 2, \ldots, n)$ can be formulated as

$$\boldsymbol{B}_\mu = [\boldsymbol{\beta}_{(\mu-1)\delta+1}, \boldsymbol{\beta}_{(\mu-1)\delta+2}, \ldots, \boldsymbol{\beta}_{\mu\delta}], \delta = b/n \quad (3)$$

where δ denotes the column size of each reconstructive slice. To maintain the reconstructive hybrid spatial–spectral slice size, n is set as a factor of the band quantity b. This segmentation roughly results in heterogeneous clusters of bands since similar continuous narrow channels may be grouped when choosing a suitable group number.

Since we successfully acquired homogeneous spatial pixels and spectral bands, there seem to be $m\times n$ reconstructive slices. Specifically, by setting the vth homogeneous pixel cluster $\boldsymbol{p}_\nu$ as the column variable and the μth adjacent band group $\boldsymbol{B}_\mu$ as the row variable, the corresponding reconstructive slice matrix $\boldsymbol{F}_\mu(\boldsymbol{p}_\nu)$ can be created as

$$\boldsymbol{F}_\mu(\boldsymbol{p}_\nu) = [\boldsymbol{\beta}_{(\mu-1)\delta+1}(\boldsymbol{p}_\nu), \boldsymbol{\beta}_{(\mu-1)\delta+2}(\boldsymbol{p}_\nu), \ldots, \boldsymbol{\beta}_{\mu\delta}(\boldsymbol{p}_\nu)]^{\mathrm{T}} \quad (4)$$

where $\boldsymbol{\beta}_i(\boldsymbol{p}_\nu)(i = (\mu-1)\delta+1, (\mu-1)\delta+2, \ldots, \mu\delta)$ denotes the subset of $\boldsymbol{\beta}_i$, holding nothing but the observation values of all homogeneous pixels contained in $\boldsymbol{p}_\nu$ and spectral vector $\boldsymbol{\beta}_i$. Therefore, the slice matrix $\boldsymbol{F}_\mu(\boldsymbol{p}_\nu)$ is a subset of $\boldsymbol{B}_\mu$, employing the same homogeneous pixel cluster $\boldsymbol{p}_\nu$.

Although each spectral subset has the same size, the number of pixels differs because of heterogeneous spatial subsets by ERSS. Moreover, performing global TRPCA on all slices of distinct sizes is inappropriate for properties of tensor operations. To solve this issue, we further design a homogeneous block reconstruction technique defined as

$$\mathbf{F}_v = \text{fold}(\boldsymbol{F}_1(\boldsymbol{p}_\nu), \boldsymbol{F}_2(\boldsymbol{p}_\nu), \ldots, \boldsymbol{F}_n(\boldsymbol{p}_\nu)). \quad (5)$$

Here, the vth reconstructed block array $\mathbf{F}_\nu \in \mathbb{R}^{m_\nu\times\delta\times n}$ is structured by arranging all 2-D subsets that only remain the universal homogeneous pixels in the cluster $\boldsymbol{p}_\nu$ of all spectral subsets $\boldsymbol{B}_\mu(\mu = 1, 2, \ldots, n)$ along the heterogeneous band group dimension.

By establishing spatial–spectral alliances from the entire tensor, the main contextual and structural features of both pixel and band variables can remain, and the input HSIs heterogeneous regions can be distinguished. Considering the hyperspectral data regulation that prominent similarities exist among regional pixels or neighboring spectral bands, the data-model-driven model provides more essential features for further research than other model-driven TRPCA models. Meanwhile, the proposed homogeneous block rebuilder offers the necessary conditions for the subsequent low-rank feature-extracting technique.

C. Local TRPCA Low-Rank Feature Extractor

Most existing T-SVD-derived TRPCA models utilize tensor singular value thresholding (T-SVT) to find the low-rank recovery component of a 3-D tensor, removing sparse parts to improve the image quality [23]. However, such approaches are costly operations requiring quite a lot of memory, global manners ignoring the heterogeneity of different spatial subregions to increase the difficulty and insufficiency of noise reduction,

and iterative algorithms separately operating on each band to destroy spectral contextual information. On the contrary, we utilize T-SVT to shrink the singular values in a divide-and-rule manner, flexibly and sufficiently reducing noises in various spatial–spectral blocks independently. The proposed local TR-PCA algorithm is performed on each reconstructive block with the objective function defined as

$$\min_{\mathbf{L}_v,\mathbf{N}_v}(\|\mathbf{L}_v\|_a+\gamma_v\|\mathbf{N}_v\|_1),\text{s.t. }\mathbf{F}_\nu=\mathbf{L}_v+\mathbf{N}_v \tag{6}$$

where the averaging tensor norm $\|\mathbf{L}_v\|_a$ and ℓ_1-norm $\|\mathbf{N}_v\|_1$ [23] are employed on the vth reconstructive homogenous block tensor $\mathbf{F}_\nu\in\mathbb{R}^{m_\nu\times\delta\times n}$ to perfectly separate its low-rank tensor element $\mathbf{L}_\nu$ from the noisy tensor element $\mathbf{N}_\nu$. $\gamma_\nu=(\max(m_\nu,\delta)\cdot n)^{-1/2}$ is the regularization parameter to make a balance between two different norms. To achieve the objective function, the alternating direction method of multipliers (ADMM) [31] is employed with an augmented Lagrange multiplier [32] for the exact recovery from corrupted observations. Since the tensor $\mathbf{F}_\nu$ is typically already known, the optimized objective model can be rewritten as

$$\begin{aligned}&\min_{\mathbf{L}_v,\mathbf{N}_v}\Phi(\mathbf{L}_v,\mathbf{N}_v,\boldsymbol{E}_v,\gamma_v,\alpha_v)\\&=\min_{\mathbf{L}_v,\mathbf{N}_v}\left(\|\mathbf{L}_v\|_a+\gamma_v\|\mathbf{N}_v\|_1+\frac{\alpha_v}{2}\|\mathbf{F}_v-\mathbf{L}_v-\mathbf{N}_v\|_F^2\right.\\&\left.+\langle\boldsymbol{E}_v,\mathbf{F}_v-\mathbf{L}_v-\mathbf{N}_v\rangle\right).\end{aligned} \tag{7}$$

Here, tensor $\boldsymbol{E}_v$ stands for the corresponding Lagrange multiplier tensor, α_v denotes the matched nonnegative parameter, and $\langle\cdot\rangle$ stands for the inner product of the input two elements. The objective low-rank recovery tensor $\mathbf{L}_v$ can be generated by iteratively updating the objective function until it reaches the convergence conditions. Specifically, when the kth $\mathbf{N}_v^k$, $\boldsymbol{E}_v^k$, γ_v, and α_v^k are fixed, the $(k+1)$th low-rank recovery tensor $\mathbf{L}_v^{k+1}$ can be generated by

$$\mathbf{L}_v^{k+1}=\arg\min_{\mathbf{L}_v}\left(\|\mathbf{L}_v\|_a+\frac{\alpha_v^k}{2}\left\|\mathbf{L}_v+\mathbf{N}_v^k-\mathbf{F}_\nu+\frac{\mathbf{E}_v^k}{\alpha_v^k}\right\|_F^2\right). \tag{8}$$

According to Lu et al. [33], the function $\phi(\cdot)$ obeys the soft-thresholding rule with a thresholding parameter $\tau_v\geq 0$ to shrink the singular values, reducing noises. Moreover, it can also serve as a proximal operator on the averaging tensor nuclear norm by providing a closed-form solution to the above updating problem as

$$\phi(\mathbf{Q}_v^k)=\arg\min_{\mathbf{L}_v}\left(\tau_v\|\mathbf{L}_v\|_a+\frac{1}{2}\left\|\mathbf{L}_v-\mathbf{Q}_v^k\right\|_F^2\right),\tau_v=(\alpha_v^k)^{-1} \tag{9}$$

where tensor $\mathbf{Q}_v^k=\mathbf{N}_v^k-\mathbf{F}_\nu+(\alpha_v^k)^{-1}\mathbf{E}_v^k$. By employing the T-SVT operator, the $(k+1)$th low-rank recovery tensor $\mathbf{L}_v^{k+1}$ can be easily obtained by $\mathbf{L}_v^{k+1}=\phi(\mathbf{Q}_v^k)$. It should be noted that the proximal operator can finally lead to a convergent result on each block of real-word grossly noisy HSI. Then, the $(k+1)$th sparse noise tensor $\mathbf{N}_v^{k+1}$ can be computed by

$$\mathbf{N}_v^{k+1}=\arg\min_{\mathbf{N}_v}\left(\gamma_v\|\mathbf{N}_v\|_1+\frac{\alpha_v^k}{2}\left\|\mathbf{L}_v^{k+1}+\mathbf{N}_v-\mathbf{F}_\nu+\frac{\mathbf{E}_v^k}{\alpha_v^k}\right\|_F^2\right). \tag{10}$$

Denote $\lceil n/2\rceil$ the smallest integer greater than or equal to $n/2$, SVD$(\cdot)$ is the matrix SVD operator, $(\hat{\boldsymbol{S}}_v^{k(l)}-\tau)_+$ denotes a diagonal matrix by shrinking the elements of $\hat{\boldsymbol{S}}_v^{k(l)}$ that is less than τ to zero, conj$(\cdot)$ is the complex conjugate of each entry of the input matrix, discrete fast Fourier transform (DFFT) of $\mathbf{H}$ is $\hat{\mathbf{H}}=\text{fft}(\mathbf{H})$, and the corresponding reverse operation is $\mathbf{H}=\text{ifft}(\hat{\mathbf{H}})$. $\hat{\boldsymbol{S}}_v^{k(l)}$ represents the diagonal eigenvalue matrix of $\hat{\boldsymbol{Q}}_v^{k(l)}$, which is the lth frontal slice of the DFFT $\hat{\mathbf{Q}}_v^k$ of $\mathbf{Q}_v^k$. The final low-rank and low-dimensional components are generated by iteratively updating the objective function until convergence, as shown in Algorithm 1.

Each reconstructive block can quickly generate convergent results and satisfy tensor incoherence conditions [23]. Therefore, the proposed SPRB-TRPCA can successfully recover the low-rank and sparse components of $\mathbf{H}$. The main cost of each iteration is calculating $\mathbf{L}_v^{k+1}\in\mathbb{R}^{m_\nu\times\delta\times n}$ $(v=1,2,\ldots,m)$ by T-SVT on each reconstructive block, holding the complexity of $O(m_\nu b\log(n)+n_{(1)}n_{(2)}^2n)$, where $n_{(1)}=\max(m_\nu,\delta)$ and $n_{(2)}=\min(m_\nu,\delta)$. Furthermore, given $M=$

Algorithm 1: Low-Rank Recovery by ADMM and T-SVT.

Input: $\mathbf{F}_\nu$, γ_ν, $\boldsymbol{p}_\nu$, $v=1,2,\ldots,m$, $\rho=1.1$, $\varepsilon=10^{-8}$, $\alpha_v^{\max}=10^{10}$, $\hat{\boldsymbol{W}}=\mathbf{0}\in\mathbb{R}^{rc\times b}$

Initialize: $\mathbf{L}_v^0=\mathbf{N}_v^0=\boldsymbol{E}_v^0=\mathbf{0}\in\mathbb{R}^{m_\nu\times\delta\times n}$, $\alpha_v^0=10^{-3}$

Output: Low-rank tensor $\mathbf{L}$

For $v=1,2,\ldots,m$ **do**

While not converged do

1: Update $\mathbf{L}_v^{k+1}$ by $\mathbf{L}_v^{k+1}=\phi(\mathbf{Q}_v^k)$ as

1.1: Perform $\hat{\mathbf{Q}}_v^k=\text{fft}(\mathbf{Q}_v^k)$.

1.2: Conduct SVD on $\hat{\boldsymbol{Q}}_v^{k(l)}$ by $\tau_v=(\alpha_v^k)^{-1}$ as

for $l=1,2,\ldots,\lceil n/2\rceil$ do

$[\hat{\boldsymbol{U}}_v^{k(l)},\hat{\boldsymbol{S}}_v^{k(l)},\hat{\boldsymbol{V}}_v^{k(l)}]=\text{SVD}(\hat{\boldsymbol{Q}}_v^{k(l)})$

$\hat{\boldsymbol{W}}_v^{k(l)}=\hat{\boldsymbol{U}}_v^{k(l)}\cdot(\hat{\boldsymbol{S}}_v^{k(l)}-\tau_v)_+\cdot(\hat{\boldsymbol{V}}_v^{k(l)})^{\mathrm{T}}$

$\hat{\boldsymbol{W}}_v^{k(n-l+1)}=\text{conj}(\hat{\boldsymbol{W}}_v^{k(l)})$ end for

1.3: Perform $\phi(\mathbf{Q}_v^k)=\text{ifft}(\hat{\mathbf{W}}_v^k)$

2: Update $\mathbf{N}_v^{k+1}$ by (17)

3: Update $\mathbf{E}_v^{k+1}$ by $\mathbf{E}_v^{k+1}=\mathbf{E}_v^k+\alpha_v^k(\mathbf{L}_v^{k+1}+\mathbf{N}_v^{k+1}-\mathbf{F}_\nu)$

4: Update α_v^{k+1} by $\alpha_v^{k+1}=\min(\rho\alpha_v^k,\alpha_v^{\max})$

Until all following convergence conditions are satisfied

$\|\mathbf{L}_v^{k+1}-\mathbf{L}_v^k\|_\infty\leq\varepsilon$, $\|\mathbf{N}_v^{k+1}-\mathbf{N}_v^k\|_\infty\leq\varepsilon$

$\|\mathbf{L}_v^{k+1}+\mathbf{N}_v^{k+1}-\mathbf{F}_v\|_\infty\leq\varepsilon$

End while

5: Construct $\hat{\boldsymbol{W}}$ by $\hat{\boldsymbol{W}}(\boldsymbol{p}_\nu)=\text{unfold}(\hat{\mathbf{W}}_v^k)\in\mathbb{R}^{m_\nu\times b}$

End For

6: Reshape $\hat{\boldsymbol{W}}$ to $\hat{\mathbf{W}}=\text{reshape}(\hat{\boldsymbol{W}},[r,c,b])$

7: Compute $\mathbf{L}=\text{ifft}(\hat{\mathbf{P}})$

8: Extract primary low-rank components of $\mathbf{L}$ by PCA

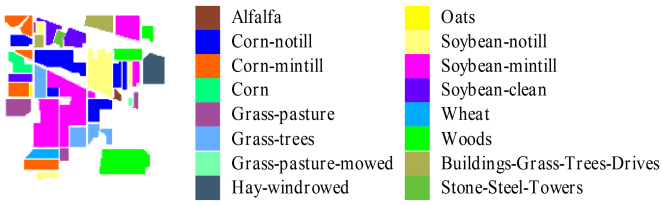

Fig. 3. Ground truth map with category labels of Indian Pines.

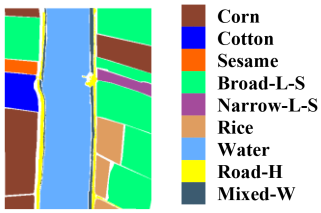

Fig. 4. Ground truth map with land-cover category labels of Longkou.

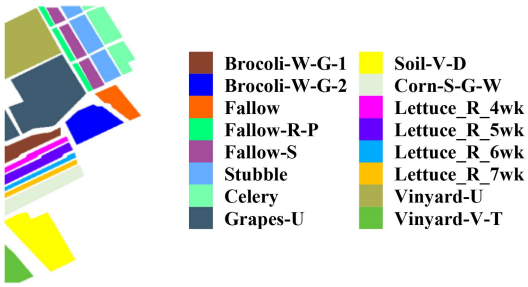

Fig. 5. Ground truth map with land-cover category labels of Salinas.

$\max(m_v)$, $N_{(1)} = \max(M, \delta)$, and $N_{(2)} = \max(n_{(2)})$, the execution complexity of the whole $\mathbf{H}$ per iteratively is $O(Mb\log(n) + N_{(1)}N_{(2)}^2 n)$. Moreover, given $\kappa_1 = \max(r, c)$, $\vartheta_1 = \min(r, c)$, $\kappa_2 = \max(r, c, b)$, and $\vartheta_2 = \min(r, c, b)$, the complexity of our local TRPCA is much lower than the complexity $O(rcb\log(b) + \kappa_1\vartheta_1^2 b + \kappa_2\vartheta_2^2)$ of the ITRPCA and the complexity $O(rcb\log(b) + \kappa_1\vartheta_1^2 b)$ of NTRPCA.

III. Experiments and Analyses

A. Dataset Description and Experimental Setup

The subsequent experiments are mainly implemented on the following publicly available hyperspectral datasets, termed Indian Pines [34], WHU-HI-Longkou (Longkou) [35], [36], Salinas, Botswana, Pavia University (PU), and Kennedy Space Center (KSC) [37]. Ground truth maps of the labeled land-cover objects and the category indices of Indian Pines, Longkou, Salinas, Botswana, PU, and KSC are all displayed in Figs. 3–8 in sequence. It should be noted that the white spaces of each map are all unlabeled spatial pixels, and the labeled pixels of Botswana and KSC only cover a small spatial subset.

Therefore, we add black borders to these maps.

1) An airborne AVIRIS sensor acquired Indian Pines within a specific scene on June 12, 1992. After abandoning 24 water-absorption channels, the HSI holds 200 available spectral bands with a wavelength range between 400 and 2500 nm. Each spectral band contains 145 × 145 pixels, which can be artificially divided into 16 categories.

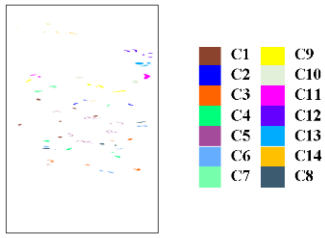

Fig. 6. Ground truth map with category labels of Botswana.

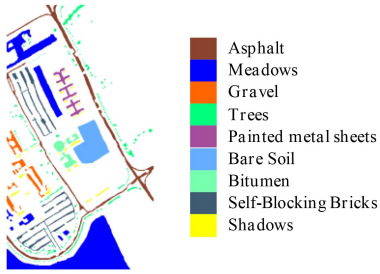

Fig. 7. Ground truth map with land-cover category labels of PU.

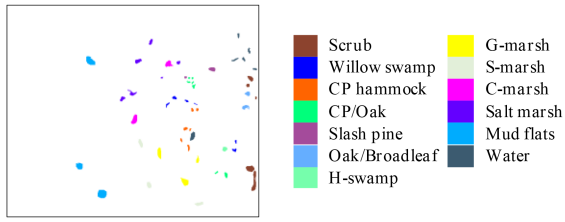

Fig. 8. Ground truth map with land-cover category labels of KSC.

2) Longkou was gathered by a UAV platform equipped with a hyperspectral sensor. The available data cube consists of 270 spectral channels of 0.463 m resolution. In addition, the dataset contains 550 × 400 pixels covering nine crop species.
3) The Salinas dataset was obtained by an airborne AVIRIS sensor from the Salinas Valley test site. After eliminating several redundant spectral variables, the employed data hold 204 bands and 512 × 217 pixels, which can be divided into 16 kinds of land covers.
4) Botswana dataset was acquired by the NASA EO-1 satellite sensor from Botswana. After removing several futile bands, the test data cube contains 145 spectral signatures and 1476 × 256 pixels, including 14 kinds of different land covers.
5) PU was gathered by a ROSIS sensor in Pavia, Italy. It contains 103 spectral bands and 610 × 340 spatial pixels, including nine material categories.

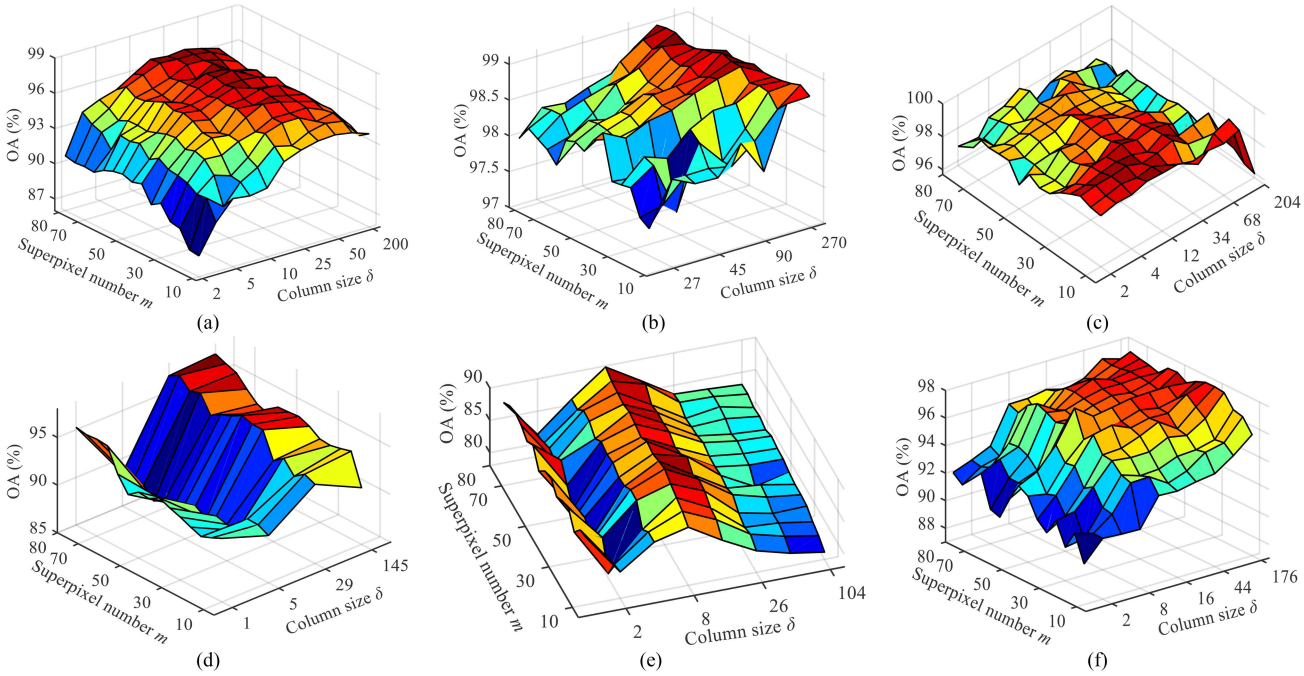

Fig. 9. Parameter tuning results in OA (%) on (a) Indian Pines, (b) Longkou, (c) Salinas, (d) Botswana, (e) PU, and (f) KSC.

6) KSC was recorded by NASA AVIRIS from the Kennedy Space Center, Florida, on March 23, 1996. After eliminating several useless channels, the dataset contains 176 spectral bands and 512 × 614 spatial pixels covering 13 species.

To verify the effectiveness of remaining spatial–spectral joint information and local denoising, a nonlinear support vector machine (SVM) classifier is introduced for recognizing objectives [38]. Considering the global pixel and spectral size, 10% of randomly selected labeled pixels of each class are considered training samples in Indian Pines, Salinas, Botswana, PU, and KSC, and 1% of randomly selected labeled pixels of each category are grouped as the training set for Longkou. The testing samples are gained by complementing the training set. Apart from parameter tunning that performs classification once, other investigations are conducted by fivefold cross validations, eliminating the effect of random operations. For objective comparison, the training sets and the parameter settings of the SVM classifier are identical and fixed in the same HSI for distinct approaches rather than random selection. To evaluate the quality of extracted features by TRPCA-related algorithms, three widely used accuracy metrics, i.e., the accuracy of each class (CA), average overall accuracy (AA), overall accuracy (OA), and kappa coefficient (Kappa), are conducted. The classification maps also choose different colors to describe separate classes and can visually illustrate the quality of extracting features.

All experiments are conducted in MATLAB on a laptop with a 1.60 GHz CPU, Windows 10 system, Intel Core i5 processors, and 8 GB memory.

B. Parameter Tuning

To obtain competitive results, fine-tuning is still necessary for further comparison. Since the proposed algorithm mainly consists of a homogeneous block rebuilder and a local TRPCA denoiser, a grid-search strategy is implemented on the two modules to tune essential hyperparameters, which are the heterogeneous superpixel number m and the number δ of homogeneous bands with each spectral subset (i.e., the column size of each frontal hybrid spatial–spectral slice). Considering the computational cost, m is set from 10 to 80 and δ is chosen from {2, 4, 5, 8, 10, 20, 25, 40, 50, 100, 200} for Indian Pines, {18, 27, 30, 45, 54, 90, 135, 270} for Longkou, {2, 3, 4, 6, 12, 17, 34, 51, 68, 102, 204} for Salinas, {1, 5, 29, 145} for Botswana, and {1, 2, 4, 8, 11, 16, 22, 44, 88, 176} for KSC. Since the number of spectral bands (i.e., 103) is a prime number, we add the last band to the original HSI. Therefore, δ is chosen from {1, 2, 4, 8, 13, 26, 52, 104} for PU.

Fig. 9 demonstrates the corresponding experimental results under different parameter values. The vertex coordinates of each block represent the effect of m and δ, where the darker red the color block indicates, the higher the accuracy of the corresponding four vertices. On Indian Pines, the classification accuracy increases as δ rises until it arrives around 40. Accordingly, setting the values of m and δ as (55, 40) is acceptable for Indian Pines. Different from the case on Indian Pines, one can observe that OA achieves satisfactory results when m arrives at the upper limit on the Longkou dataset. Therefore, setting the values of m and δ as (75, 270) and (30, 12) is recommended to guarantee satisfactory performance for Longkou and Salinas, respectively. Similarly,

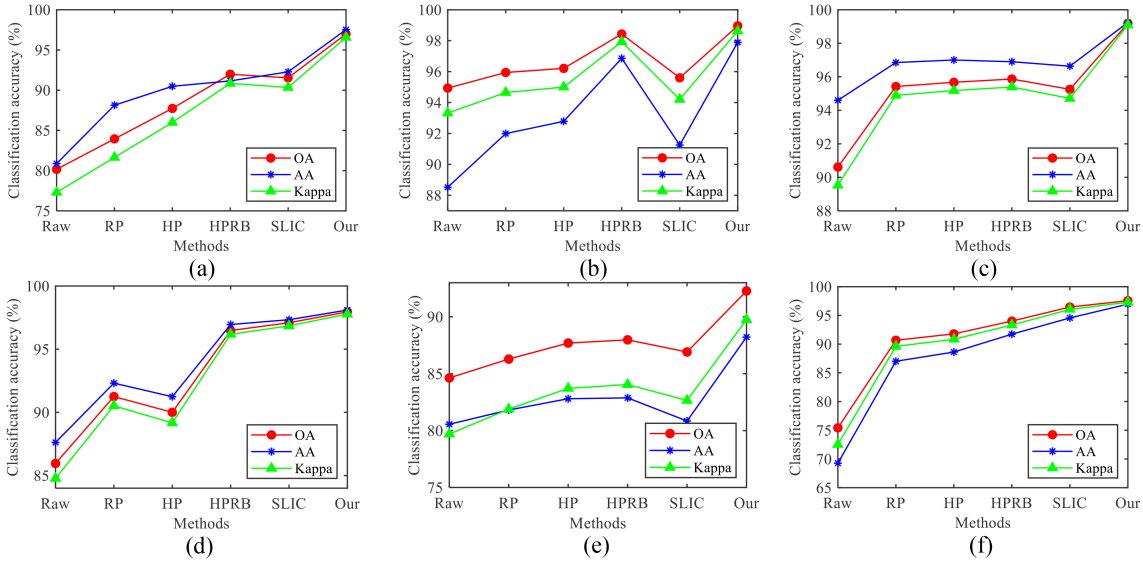

Fig. 10. Ablation experimental classification results in OA, AA, and Kappa (%) obtained by raw, RP, HP, HPRB, SLIC, and our model (Our) on (a) Indian Pines, (b) Longkou, (c) Salinas, (d) Botswana, (e) PU, and (f) KSC.

(75, 29), (80, 8), and (30, 176) are the setting values of (m, δ)on Botswana, PU, and KSC, respectively.

Comparing the results**,** we have two observations**.** First, as shown in Fig. 9, the setting values of δ on Longkou and KSC are significantly larger than those in other HSIs. This is probably because Longkou and KSC have less spectral heterogeneity and more spare noise in spatial–spectral contexts than others. Consequently, a more significant δ is needed to retain the contextual information of homogeneous spectral bands and remove the noises among spatial and spectral subspaces. Moreover, it should be noted that satisfactory performance is usually generated when m is remarkably larger than the number of classes. Such results may be attributed to the pixels on homolabeled regions still holding distinguishable characteristics, demonstrating the effectiveness of the designed homogeneous block rebuilder and local TRPCA denoiser.

C. Ablation Study

In this section, step-by-step ablation experiments are conducted to validate the necessities of every essential component of our approach. Since the proposed model consists of a homogeneous block rebuilder and a local TRPCA low-rank feature extractor, the ablation study is mainly conducted on these two components. To be specific, we employ the raw HSI (Raw), local TRPCA with only regional pixels (RP), local TRPCA with only homogeneous pixels (HP), local TRPCA with homogeneous pixels, and randomly grouped bands (HPRB) as comparative models. Moreover, simple linear iterative clustering (SLIC) [39], another superpixel segmentation technique, is also adopted as a replacement for ERSS. Fig. 10 illustrates the classification results of these comparative models and our model in OA, AA, and Kappa on the six introduced hyperspectral datasets.

From Fig. 10, one can observe that the classification accuracy in all metrics generally increases with each step of the model on most datasets, and our proposed model surpasses all comparative models on all datasets. Generally, the designed recognition of homogeneous pixels and bands, local TRPCA processing, and ERSS are all beneficial to promote HSI classification. Specifically, in Fig. 10(a), the accuracy of HP is higher than RB, illustrating the effectiveness of homogeneous spatial pixels and TRPCA localization. Such superiority also occurs when applying homogeneous band groups (Our) instead of only random band groups (HPRB), which can prove the effectiveness of employing homogeneous spectral bands. Furthermore, it should be noted that SLIC is not adaptative to provide beneficial homogeneity information for the proposed model because the adaptation of SLIC may further decrease the classification accuracy in most datasets. It also demonstrates the necessity of the ERSS technique to identify heterogeneous pixel boundaries.

D. Robust Feature Extraction Capability to Different HSIs

Several relevant algorithms are taken as comparative methods to demonstrate the superiority of our model for HSIs. RPCA [11] is one of the most classic and well-known techniques for signal feature extraction and is a 2-D foundation of TRPCA. Furthermore, MLTRPCA [22], ETRPCA [24], PTRPCA [40], NNTRPCA [21], ITRPCA [25], and NTRPCA [23] are manipulated as control groups during the parallel experiments. For objective comparison in all parallel tasks, 35 principal components of all HSIs are constantly retained by PCA.

The regularization parameter λ of ETRPCA was set to $(\min(r, c)\, b)^{-1/2}$, while the parameter of PTRPCA, NTRPCA, and NNTRPCA was set to $(\max(r, c)\, b)^{-1/2}$. The regularization parameters of ITRPCA were assigned to $(\max(\min(r, c), b))^{-1/2}$ and $2(\max(r, c)\, b)^{-1/2}$, and the parameter of MLTRPCA was fixed at $r^{1/2}$. Subsequently, we mainly employ computing time, classification accuracy, and feature visualization to validate the

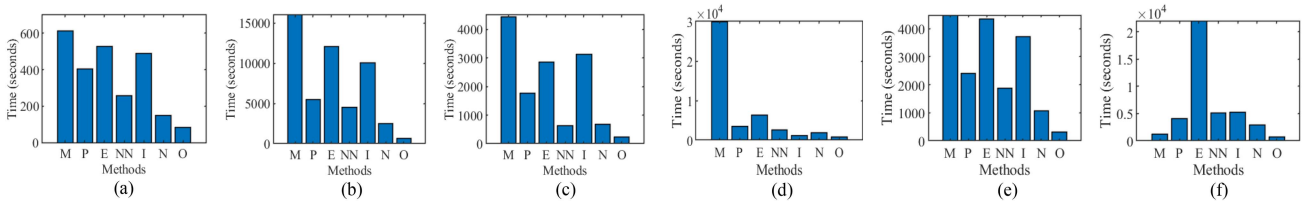

Fig. 11. Computing time (seconds) of MLTRPCA (M), PTRPCA (P), ETRPCA (E), NNTRPCA (NN), ITRPCA (I), NTRPCA (N), and our model (O) on (a) Indian Pines, (b) Longkou, (c) Salinas, (d) Botswana, (e) PU, and (f) KSC.

TABLE I
CLASSIFICATION ACCURACIES (%) AND STANDARD DEVIATIONS ON INDIAN PINES (FOLLOWING THE PLUS/MINUS SIGNS)

Class	RPCA	MLTRPCA	PTRPCA	ETRPCA	NCRPCA	ITRPCA	NTRPCA	SPRB-TRPCA
Alfalfa	81.52 ± 0.69	47.22 ± 1.35	62.56 ± 1.46	98.02 ± 0.27	**100.0 ± 0.00**	98.57 ± 0.32	97.60 ± 0.40	**100.0 ± 0.00**
Corn-N-T	65.98 ± 0.07	76.96 ± 0.25	64.64 ± 0.65	89.79 ± 0.09	90.73 ± 0.15	88.45 ± 0.09	88.59 ± 0.04	**95.80 ± 0.16**
Corn-M-T	69.56 ± 0.33	66.06 ± 0.34	72.55 ± 3.31	90.48 ± 0.23	87.97 ± 0.24	91.92 ± 0.24	91.96 ± 0.21	**94.94 ± 0.08**
Corn	64.38 ± 0.29	66.87 ± 0.31	62.53 ± 4.79	86.79 ± 0.28	83.11 ± 0.39	84.32 ± 0.31	83.78 ± 0.33	**88.41 ± 0.44**
Grass-P	85.78 ± 0.19	75.58 ± 0.48	87.60 ± 2.56	**99.15 ± 0.10**	99.10 ± 0.08	99.08 ± 0.08	99.08 ± 0.09	98.97 ± 0.09
Grass-T	90.27 ± 0.13	84.85 ± 0.18	93.17 ± 1.88	99.03 ± 0.04	98.52 ± 0.07	99.36 ± 0.05	99.33 ± 0.05	**99.73 ± 0.05**
Grass-P-M	**100.0 ± 0.00**	35.42 ± 0.79	94.27 ± 4.24	99.20 ± 0.18	99.23 ± 0.17	99.20 ± 0.18	**100.0 ± 0.00**	99.23 ± 0.17
Hay-W	92.52 ± 0.12	98.75 ± 0.07	94.86 ± 1.59	98.86 ± 0.11	99.31 ± 0.08	98.22 ± 0.13	98.14 ± 0.15	**100.0 ± 0.00**
Oats	76.00 ± 4.34	31.15 ± 0.62	61.20 ± 18.2	**100.0 ± 0.00**	95.42 ± 0.73	98.67 ± 0.30	**100.0 ± 0.00**	98.95 ± 0.24
Soybean-N-T	67.20 ± 0.16	74.77 ± 0.26	73.45 ± 2.75	92.69 ± 0.24	91.44 ± 0.38	91.77 ± 0.25	92.16 ± 0.23	**96.52 ± 0.18**
Soybean-M-T	66.18 ± 0.13	82.45 ± 0.18	79.27 ± 1.19	92.39 ± 0.14	91.59 ± 0.11	92.76 ± 0.11	92.80 ± 0.13	**96.25 ± 0.14**
Soybean-C-T	79.83 ± 0.14	82.08 ± 0.38	67.55 ± 5.05	93.16 ± 0.14	91.28 ± 0.33	92.96 ± 0.18	93.12 ± 0.18	**96.81 ± 0.12**
Wheat	96.68 ± 0.10	89.43 ± 0.33	95.14 ± 2.12	**100.0 ± 0.00**	**100.0 ± 0.00**	**100.0 ± 0.00**	**100.0 ± 0.00**	98.97 ± 0.23
Woods	88.89 ± 0.07	94.62 ± 0.13	92.90 ± 0.82	98.14 ± 0.08	96.72 ± 0.04	97.55 ± 0.08	97.76 ± 0.09	**99.40 ± 0.10**
Buildings	74.06 ± 0.51	90.59 ± 0.29	67.57 ± 7.75	94.43 ± 0.31	89.64 ± 0.54	92.81 ± 0.25	92.85 ± 0.23	**96.53 ± 0.34**
Stone-S-T	99.17 ± 0.08	81.03 ± 1.29	97.53 ± 1.19	98.80 ± 0.07	97.99 ± 0.16	98.86 ± 0.12	99.13 ± 0.08	**100.0 ± 0.00**
OA (%)	75.33 ± 0.05	81.24 ± 0.09	78.65 ± 0.72	93.90 ± 0.05	92.92 ± 0.06	93.61 ± 0.06	93.69 ± 0.05	**97.00 ± 0.02**
AA (%)	81.13 ± 0.25	73.61 ± 0.23	79.17 ± 1.01	95.68 ± 0.06	94.50 ± 0.10	95.28 ± 0.08	95.39 ± 0.06	**97.53 ± 0.05**
Kappa×100	71.66 ± 0.06	78.59 ± 0.11	75.61 ± 0.81	93.03 ± 0.06	91.92 ± 0.07	92.70 ± 0.07	92.80 ± 0.06	**96.57 ± 0.03**

efficiency, adaptivity, and effectiveness of the proposed SPRB-TRPCA.

1) *Efficiency and Adaptivity Analysis:* Since the TRPCA algorithm mainly relies on ADMM alternatively obtaining the optimal solution to the objective problem, the computing time is remarkably higher than other model-driven traditional feature extractors but lower than data-driven deep-learning machines. Nevertheless, our data-model-driven SPRB-TRPCA relies on both hyperspectral data regulation and algorithm mechanism to promote the processing course. Based on the reconstructive homogeneous blocks and local TRPCA, our model ought to be more efficient and adaptive to hyperspectral datasets than others. To prove this advantage, we compare the practical running time of different TRPCA algorithms on six employed HSIs, which is demonstrated in Fig. 11.

In Fig. 11, our designed model SPRB-TRPCA costs the lost operating time among these models on all datasets. Benefiting to homogeneous blocks and elaborately divide-and-rule manner, we only need to find the optimal solution in local spatial–spectral regions. Although such localized alternative processing needs to be conducted several times because of different blocks, the whole computing time is still much lower than the others, especially on Indian Pines and PU datasets. The stable and remarkable efficiency also represents that our SPRB-TRPCA is adaptive to the high-dimensional HSIs, illustrating the superiority of the localization to TRPCA models.

2) *Effectiveness, Robustness, and Adaptivity Analysis in Classification: Our* SPRB-TRPCA mainly consists of a homogeneous block rebuilder and a local TRPCA feature extractor. As a result, the final output of our model is low-dimensional primary features. The extracted features are indispensable for effectively classifying the unknown land-cover objectives and mitigating the Hughes phenomenon, which also reduces operating costs and increases classification performance. Therefore, the classification application of different feature sets on different HSIs is another essential and practical way to reflect the superiority of our model. The detailed setting of the classification process is already introduced in the experimental setup. Tables I–VI present the classification results in CA, OA, AA, and Kappa with a standard error of five cross validations of different TRPCA-related algorithms on six employed testing HSIs, while the matching classification maps are given in Figs. 12–17. In each Table, the best results for each quantitative metric are labeled in bold. In general, the performance of our model is significantly better than RPCA, MLTRPCA, PTRPCA, ETRPCA, NNTRPCA, ITRPCA, and NTRPCA combined.

TABLE II
CLASSIFICATION ACCURACIES (%) AND STANDARD DEVIATIONS ON LONGKOU (FOLLOWING THE PLUS/MINUS SIGNS)

Class	RPCA	MLTRPCA	PTRPCA	ETRPCA	NNTRPCA	ITRPCA	NTRPCA	SPRB-TRPCA
Corn	92.68 ± 0.09	73.71 ± 0.17	96.24 ± 0.31	98.11 ± 0.06	97.77 ± 0.04	98.08 ± 0.04	98.10 ± 0.04	**98.71 ± 0.04**
Cotton	71.61 ± 0.26	74.74 ± 0.04	81.31 ± 1.41	87.81 ± 0.16	87.15 ± 0.13	90.10 ± 0.18	90.04 ± 0.19	**97.64 ± 0.16**
Sesame	67.01 ± 0.28	32.21 ± 0.96	97.96 ± 1.66	99.15 ± 0.08	98.56 ± 0.16	99.31 ± 0.08	99.29 ± 0.08	**99.63 ± 0.03**
Broad-L-S	88.03 ± 0.11	78.74 ± 0.20	94.52 ± 0.74	97.15 ± 0.05	96.47 ± 0.12	96.60 ± 0.09	96.61 ± 0.09	**98.95 ± 0.04**
Narrow-L-S	71.07 ± 0.76	56.07 ± 0.74	83.80 ± 2.01	88.41 ± 0.40	90.47 ± 0.32	90.90 ± 0.40	90.85 ± 0.40	**97.97 ± 0.04**
Rice	92.68 ± 0.18	70.73 ± 0.29	99.38 ± 0.27	99.82 ± 0.01	99.68 ± 0.02	**99.90 ± 0.00**	**99.90 ± 0.00**	99.80 ± 0.02
Water	99.92 ± 0.01	97.08 ± 0.04	99.96 ± 0.00	99.98 ± 0.00	**99.99 ± 0.00**	99.98 ± 0.00	99.98 ± 0.00	99.98 ± 0.00
Road-H	82.31 ± 0.17	64.27 ± 0.19	93.13 ± 2.79	95.12 ± 0.18	94.95 ± 0.20	96.09 ± 0.16	**96.11 ± 0.16**	96.10 ± 0.28
Mixed-W	69.21 ± 0.32	58.83 ± 0.76	91.56 ± 3.90	90.77 ± 0.25	90.38 ± 0.13	90.56 ± 0.32	90.50 ± 0.32	**92.25 ± 0.21**
AA (%)	91.29 ± 0.03	82.05 ± 0.08	96.08 ± 0.27	97.63 ± 0.02	97.36 ± 0.04	97.64 ± 0.02	97.64 ± 0.02	**98.96 ± 0.00**
OA (%)	81.61 ± 0.12	67.38 ± 0.19	93.09 ± 0.74	95.15 ± 0.06	95.04 ± 0.06	95.72 ± 0.08	95.71 ± 0.07	**97.89 ± 0.05**
Kappa×100	88.48 ± 0.04	76.12 ± 0.11	94.83 ± 0.36	96.88 ± 0.03	96.52 ± 0.05	96.89 ± 0.03	96.90 ± 0.03	**98.63 ± 0.01**

TABLE III
CLASSIFICATION ACCURACIES (%) AND STANDARD DEVIATIONS ON SALINAS (FOLLOWING THE PLUS/MINUS)

Class	RPCA	MLTRPCA	PTRPCA	ETRPCA	NNTRPCA	ITRPCA	NTRPCA	SPRB-TRPCA
Brocoli-W-G-1	89.49 ± 0.23	66.08 ± 0.39	99.53 ± 0.44	99.82 ± 0.02	99.94 ± 0.01	99.81 ± 0.03	99.82 ± 0.02	**100.0 ± 0.00**
Brocoli-W-G-2	91.83 ± 0.12	88.37 ± 0.31	99.50 ± 0.62	98.77 ± 0.14	99.57 ± 0.04	99.18 ± 0.08	99.23 ± 0.07	**99.99 ± 0.00**
Fallow	76.65 ± 0.37	60.94 ± 0.24	97.94 ± 0.69	98.52 ± 0.17	98.26 ± 0.21	99.20 ± 0.05	99.28 ± 0.05	**99.79 ± 0.01**
Fallow-R-P	97.61 ± 0.06	22.41 ± 0.55	98.54 ± 0.86	**99.34 ± 0.03**	99.07 ± 0.08	98.85 ± 0.08	98.92 ± 0.07	98.89 ± 0.09
Fallow-S	91.07 ± 0.20	37.91 ± 0.16	96.72 ± 0.82	98.51 ± 0.10	98.29 ± 0.10	98.76 ± 0.04	98.72 ± 0.05	**99.77 ± 0.04**
Stubble	**99.98 ± 0.00**	45.66 ± 0.14	99.97 ± 0.00	98.50 ± 0.33	99.31 ± 0.06	98.44 ± 0.34	98.38 ± 0.35	99.95 ± 0.00
Celery	98.20 ± 0.06	58.42 ± 0.17	**99.61 ± 0.67**	97.89 ± 0.29	99.41 ± 0.09	98.87 ± 0.09	98.51 ± 0.14	98.41 ± 0.17
Grapes-U	71.69 ± 0.09	75.83 ± 0.36	79.32 ± 0.88	84.72 ± 0.16	84.52 ± 0.13	86.64 ± 0.23	86.96 ± 0.16	**98.54 ± 0.07**
Soil-V-D	94.25 ± 0.13	88.38 ± 0.15	98.76 ± 0.27	99.35 ± 0.02	99.62 ± 0.03	99.50 ± 0.03	99.49 ± 0.03	**99.63 ± 0.02**
Corn-S-G-W	84.17 ± 0.22	60.74 ± 0.22	96.75 ± 0.62	97.71 ± 0.13	98.06 ± 0.08	**98.96 ± 0.04**	98.87 ± 0.04	98.93 ± 0.04
Lettuce_R_4wk	78.93 ± 0.77	40.24 ± 0.48	98.05 ± 0.94	99.21 ± 0.04	98.86 ± 0.05	99.75 ± 0.02	99.73 ± 0.02	**99.94 ± 0.01**
Lettuce_R_5wk	86.19 ± 0.20	46.15 ± 0.18	96.54 ± 0.81	99.04 ± 0.05	**99.38 ± 0.04**	99.26 ± 0.04	99.35 ± 0.05	98.41 ± 0.14
Lettuce_R_6wk	90.04 ± 0.22	21.92 ± 0.34	95.23 ± 1.78	96.61 ± 0.10	96.51 ± 0.18	97.67 ± 0.15	97.92 ± 0.19	**98.90 ± 0.00**
Lettuce_R_7wk	94.26 ± 0.24	27.14 ± 0.32	95.21 ± 3.17	98.34 ± 0.11	97.97 ± 0.12	**98.52 ± 0.10**	98.45 ± 0.10	97.59 ± 0.06
Vinyard-U	63.65 ± 0.28	80.25 ± 0.14	77.15 ± 0.98	80.01 ± 0.14	80.92 ± 0.08	82.03 ± 0.25	82.20 ± 0.25	**99.16 ± 0.03**
Vinyard-V-T	99.09 ± 0.04	76.24 ± 0.23	98.05 ± 1.90	99.90 ± 0.02	99.74 ± 0.05	99.61 ± 0.07	99.61 ± 0.07	**99.99 ± 0.00**
OA (%)	84.08 ± 0.08	67.49 ± 0.05	91.58 ± 0.19	93.30 ± 0.06	93.63 ± 0.03	94.19 ± 0.05	94.26 ± 0.05	**99.19 ± 0.01**
AA (%)	87.94 ± 0.09	56.04 ± 0.07	95.43 ± 0.23	96.64 ± 0.03	96.84 ± 0.03	97.19 ± 0.03	97.21 ± 0.03	**99.24 ± 0.01**
Kappa×100	82.22 ± 0.09	63.61 ± 0.06	90.61 ± 0.22	92.53 ± 0.07	92.90 ± 0.04	93.53 ± 0.06	93.61 ± 0.05	**99.10 ± 0.01**

TABLE IV
CLASSIFICATION ACCURACIES (%) AND STANDARD DEVIATIONS ON BOTSWANA (FOLLOWING THE PLUS/MINUS)

Class	RPCA	MLTRPCA	PTRPCA	ETRPCA	NNTRPCA	ITRPCA	NTRPCA	SPRB-TRPCA
C1	98.62 ± 0.83	30.77 ± 7.53	98.86 ± 0.60	**100.0 ± 0.00**	**100.0 ± 0.00**	**100.0 ± 0.00**	99.84 ± 0.37	99.84 ± 0.37
C2	79.41 ± 6.68	15.52 ± 14.7	98.41 ± 1.90	**99.78 ± 0.50**	99.13 ± 1.40	98.18 ± 1.55	98.90 ± 0.77	99.35 ± 0.97
C3	84.70 ± 3.64	11.86 ± 2.52	90.50 ± 5.14	98.30 ± 1.54	99.56 ± 0.54	99.05 ± 1.67	99.82 ± 0.39	**99.65 ± 0.57**
C4	85.88 ± 4.02	8.050 ± 7.67	90.71 ± 5.29	95.51 ± 3.32	97.13 ± 2.46	**98.30 ± 1.89**	97.72 ± 2.27	97.39 ± 1.37
C5	78.29 ± 7.27	19.55 ± 2.72	82.76 ± 3.63	88.26 ± 5.25	93.14 ± 5.44	92.83 ± 4.95	93.41 ± 6.51	**95.27 ± 2.44**
C6	67.11 ± 2.56	16.44 ± 1.63	68.34 ± 6.15	80.24 ± 2.39	92.41 ± 3.14	94.57 ± 3.07	94.43 ± 3.53	**96.75 ± 1.90**
C7	98.57 ± 0.49	15.77 ± 2.97	99.04 ± 0.70	99.91 ± 0.19	**100.0 ± 0.00**	**100.0 ± 0.00**	**100.0 ± 0.00**	99.91 ± 0.19
C8	94.48 ± 2.34	11.60 ± 6.60	95.82 ± 3.49	98.18 ± 2.00	**99.89 ± 0.25**	99.88 ± 0.26	99.78 ± 0.30	99.34 ± 0.46
C9	70.39 ± 3.23	13.33 ± 1.21	79.86 ± 3.18	87.35 ± 1.93	95.23 ± 4.58	98.05 ± 1.33	98.31 ± 0.74	**99.65 ± 0.50**
C10	66.28 ± 2.73	23.52 ± 2.85	77.79 ± 3.31	87.54 ± 4.25	94.02 ± 2.39	**95.96 ± 1.96**	94.58 ± 2.07	93.76 ± 3.32
C11	84.73 ± 2.43	32.90 ± 4.91	86.18 ± 3.08	95.39 ± 1.59	**99.70 ± 0.67**	99.55 ± 0.31	99.63 ± 0.64	99.13 ± 1.54
C12	78.35 ± 6.48	39.31 ± 5.51	84.23 ± 1.02	91.79 ± 0.85	95.34 ± 2.00	93.55 ± 3.53	94.69 ± 1.83	**99.75 ± 0.56**
C13	72.86 ± 4.21	35.63 ± 4.88	77.17 ± 2.14	87.47 ± 3.15	95.26 ± 2.03	93.89 ± 2.82	95.69 ± 2.19	**97.68 ± 2.35**
C14	97.33 ± 2.68	21.96 ± 27.4	99.76 ± 0.55	98.14 ± 2.11	98.19 ± 2.59	98.89 ± 2.48	99.10 ± 2.01	**100.0 ± 0.00**
OA (%)	81.12 ± 1.69	21.30 ± 1.38	86.10 ± 1.50	92.52 ± 1.41	96.79 ± 0.84	97.20 ± 0.46	97.38 ± 0.70	**98.20 ± 0.50**
AA (%)	82.64 ± 1.66	21.26 ± 3.80	87.82 ± 1.07	93.42 ± 1.28	97.07 ± 0.71	97.34 ± 0.50	97.57 ± 0.58	**98.39 ± 0.45**
Kappa×100	79.54 ± 1.84	14.17 ± 1.91	84.92 ± 1.63	91.89 ± 1.53	96.52 ± 0.91	96.96 ± 0.50	97.16 ± 0.76	**98.05 ± 0.54**

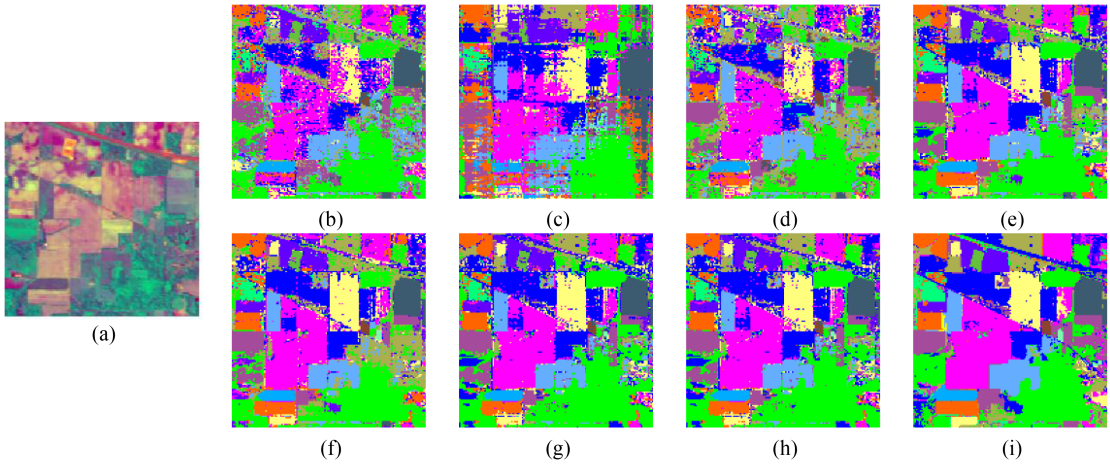

Fig. 12. Classification maps of different TRPCA-related algorithms on Indian Pines. (a) False RGB image. (b) RPCA. (c) MLTRPCA. (d) PTRPCA. (e) ETRPCA. (f) NNTRPCA. (g) ITRPCA. (h) NTRPCA. (i) SPRB-TRPCA.

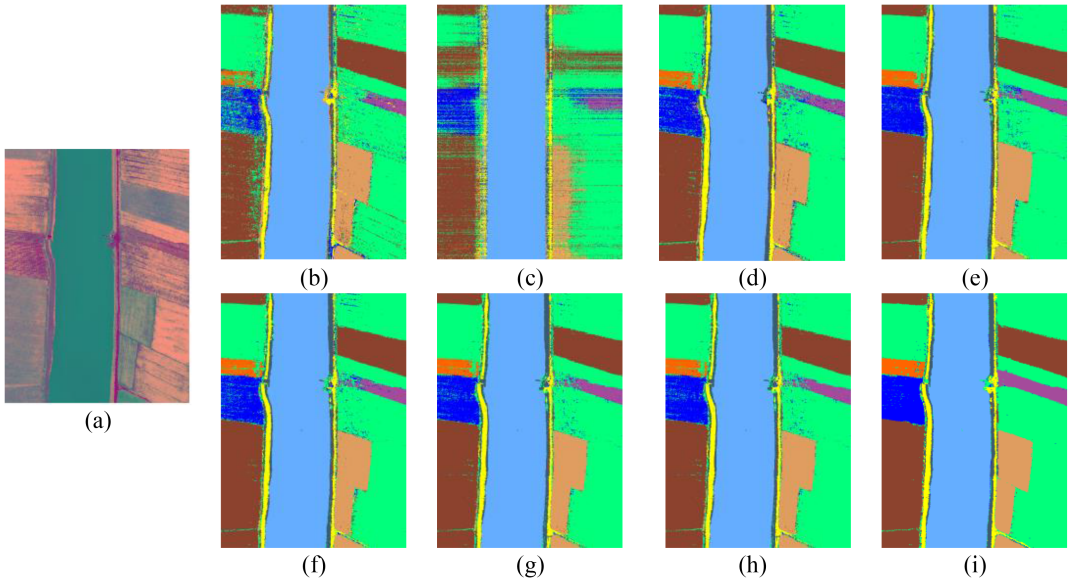

Fig. 13. Classification maps of different TRPCA-related algorithms on Longkou. (a) False RGB image. (b) RPCA. (c) MLTRPCA. (d) PTRPCA. (e) ETRPCA. (f) NNTRPCA. (g) ITRPCA. (h) NTRPCA. (i) SPRB-TRPCA.

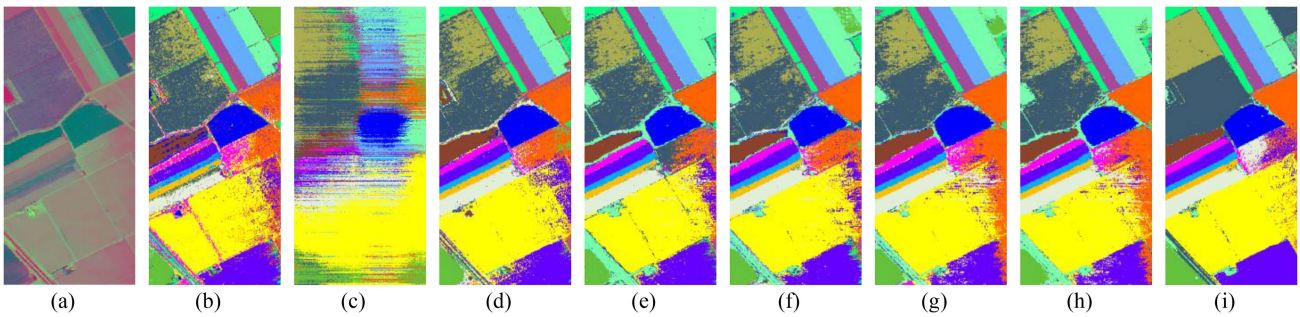

Fig. 14. Classification maps of different TRPCA-related algorithms on Salinas. (a) False RGB image. (b) RPCA. (c) MLTRPCA. (d) PTRPCA. (e) ETRPCA. (f) NNTRPCA. (g) ITRPCA. (h) NTRPCA. (i) SPRB-TRPCA.

TABLE V
CLASSIFICATION ACCURACIES (%) AND STANDARD DEVIATIONS ON PU (FOLLOWING THE PLUS/MINUS)

Class	RPCA	MLTRPCA	PTRPCA	ETRPCA	NNTRPCA	ITRPCA	NTRPCA	SPRB-TRPCA
Asphalt	89.03 ± 1.44	42.08 ± 19.8	89.19 ± 1.60	91.86 ± 1.02	90.57 ± 0.49	91.60 ± 0.95	91.61 ± 0.87	**92.40 ± 0.97**
Meadows	86.91 ± 0.81	68.45 ± 2.67	94.15 ± 0.32	95.88 ± 0.19	96.02 ± 0.45	95.91 ± 0.28	95.91 ± 0.27	**98.20 ± 0.25**
Gravel	45.37 ± 2.39	43.34 ± 0.84	64.67 ± 3.12	70.90 ± 4.51	69.34 ± 3.19	69.37 ± 4.64	69.48 ± 4.57	**75.44 ± 3.76**
Trees	80.01 ± 6.54	15.86 ± 1.30	95.67 ± 0.80	97.78 ± 0.78	97.87 ± 0.54	97.09 ± 1.74	97.08 ± 1.66	**98.37 ± 0.92**
Painted metal	94.47 ± 2.79	68.70 ± 3.50	99.98 ± 0.00	99.98 ± 0.00	**100.0 ± 0.00**	99.98 ± 0.00	99.98 ± 0.00	**100.0 ± 0.00**
Bare Soil	70.86 ± 6.45	43.60 ± 4.49	77.95 ± 4.14	81.16 ± 2.63	80.14 ± 2.24	81.51 ± 2.43	81.44 ± 2.50	**93.75 ± 1.65**
Bitumen	48.67 ± 3.48	32.01 ± 5.86	49.38 ± 2.15	**58.81 ± 1.90**	53.69 ± 2.95	57.39 ± 3.54	57.42 ± 3.67	58.60 ± 5.01
Self-Blocking	63.18 ± 2.65	26.43 ± 1.02	69.43 ± 2.66	71.20 ± 1.55	72.55 ± 3.08	70.90 ± 1.95	70.92 ± 1.92	**77.56 ± 1.53**
Shadows	**99.93 ± 0.00**	13.64 ± 7.86	99.05 ± 0.54	98.61 ± 1.26	98.88 ± 0.75	99.28 ± 0.66	99.23 ± 0.74	99.67 ± 0.32
OA (%)	79.69 ± 1.32	50.17 ± 0.60	86.11 ± 0.86	88.66 ± 0.62	88.22 ± 0.43	88.45 ± 0.78	88.45 ± 0.81	**92.27 ± 0.55**
AA (%)	75.38 ± 1.22	39.35 ± 1.73	82.16 ± 0.94	85.13 ± 0.63	84.34 ± 0.36	84.78 ± 1.02	84.79 ± 1.05	**88.22 ± 0.69**
Kappa×100	72.89 ± 1.39	33.74 ± 1.66	81.62 ± 1.09	85.01 ± 0.78	84.45 ± 0.56	84.74 ± 1.00	84.74 ± 1.04	**89.75 ± 0.71**

TABLE VI
CLASSIFICATION ACCURACIES (%) AND STANDARD DEVIATIONS ON KSC (FOLLOWING THE PLUS/MINUS SIGNS)

Class	RPCA	MLTRPCA	PTRPCA	ETRPCA	NCRPCA	ITRPCA	NTRPCA	SPRB-TRPCA
Scrub	92.15 ± 3.51	38.11 ± 1.63	88.72 ± 2.00	95.97 ± 2.30	95.90 ± 1.43	93.65 ± 0.88	95.02 ± 3.60	**96.61 ± 0.97**
Willow swamp	81.41 ± 1.82	38.33 ± 39.1	88.29 ± 2.61	95.74 ± 1.67	97.14 ± 1.37	95.73 ± 3.13	94.82 ± 5.36	**99.19 ± 0.97**
CP hammock	70.88 ± 2.11	16.67 ± 23.6	76.51 ± 3.33	86.51 ± 2.85	91.72 ± 3.01	76.04 ± 3.82	85.35 ± 5.59	**97.09 ± 1.53**
CP/Oak	46.59 ± 4.85	9.170 ± 12.6	62.95 ± 5.32	80.50 ± 2.36	86.83 ± 2.34	71.96 ± 2.46	81.75 ± 5.35	**93.14 ± 2.93**
Slash pine	50.04 ± 12.4	0.000 ± 0.00	76.05 ± 7.41	87.56 ± 2.81	92.46 ± 8.32	88.52 ± 6.29	**93.20 ± 5.23**	91.94 ± 3.50
Oak/Broadleaf	73.78 ± 9.73	0.000 ± 0.00	67.22 ±1.69	93.50 ± 2.39	**94.30 ± 2.56**	84.36 ± 2.63	93.63 ± 2.78	92.96 ± 2.30
H-swamp	75.25 ± 5.13	0.000 ± 0.00	80.24 ± 6.56	92.70 ± 2.28	93.78 ± 0.93	97.74 ± 2.18	91.50 ± 2.59	**99.37 ± 0.94**
Water	79.62 ± 2.32	11.78 ± 7.25	88.11 ± 2.58	95.81 ± 0.96	**97.28 ± 1.64**	92.94 ± 2.09	95.68 ± 1.12	96.64 ± 0.67
G-marsh	89.78 ± 3.07	27.33 ± 0.91	90.96 ± 1.42	98.59 ± 0.56	**99.62 ± 0.32**	98.15 ± 0.25	96.94 ± 0.65	98.40 ± 0.69
S-marsh	95.00 ± 3.04	27.01 ± 5.23	98.54 ± 1.53	99.94 ± 0.12	98.90 ± 2.31	**100.0 ± 0.00**	99.35 ± 0.78	98.82 ± 1.48
C-marsh	97.69 ± 0.39	29.15 ± 2.81	98.53 ± 0.80	99.06 ± 1.29	97.69 ± 3.02	**99.26 ± 0.39**	99.16 ± 1.19	98.04 ± 1.84
Salt marsh	91.45 ± 5.88	60.44 ± 4.10	96.38 ± 0.92	98.85 ± 0.24	**99.02 ± 0.24**	98.89 ± 0.83	98.99 ± 0.58	98.31 ± 0.43
Mud flats	99.74 ± 0.26	59.29 ± 0.60	**100.0 ± 0.00**	**100.0 ± 0.00**	**100.0 ± 0.00**	99.98 ± 0.00	**100.0 ± 0.00**	**100.0 ± 0.00**
OA (%)	86.35 ± 0.86	42.22 ± 0.39	89.96 ± 0.58	96.13 ± 0.55	96.95 ± 0.42	94.23 ± 0.47	95.75 ± 0.93	**97.57 ± 0.31**
AA (%)	80.26 ± 1.94	24.41 ± 2.64	85.58 ± 1.27	94.21 ± 0.58	95.74 ± 0.84	92.09 ± 0.77	94.26 ± 0.67	**96.96 ± 0.49**
Kappa×100	84.79 ± 0.96	33.01 ± 0.48	88.80 ± 0.64	95.68 ± 0.62	96.61 ± 0.47	93.57 ± 0.53	95.27 ± 1.04	**97.29 ± 0.35**

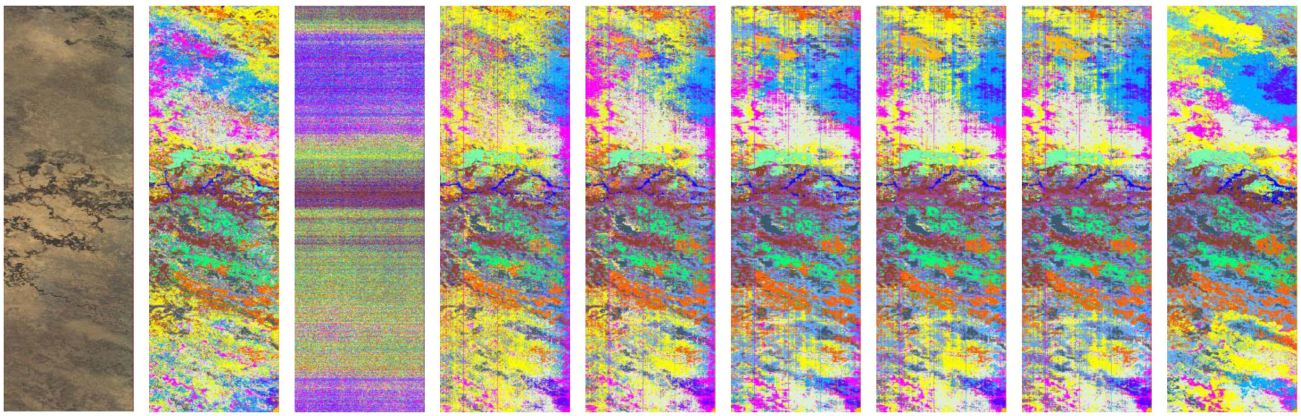

Fig. 15. Classification maps of different TRPCA-related algorithms in Botswana. (a) False RGB image. (b) RPCA. (c) MLTRPCA. (d) PTRPCA. (e) ETRPCA. (f) NNTRPCA. (g) ITRPCA. (h) NTRPCA. (i) SPRB-TRPCA.

In Tables I–VI, one can observe that SPRB-TRPCA is always more effective than others in OA, AA, and Kappa on all HSI datasets. Moreover, it should be noted that not all tensor-built TRPCA algorithms outperform the nontensor RPCA algorithm. Theoretically, the TRPCA is supposed to surpass RPCA in maintaining a 3-D data structure and a 2-D spatial structure. However, the classification performance of features by MLTRPCA is always lower than that of RPCA on most datasets except Indian Pines. It may be the consequence of that MLTRPCA is not adaptive to the vast and high-dimensional HSIs. Fortunately, other TRPCA algorithms constantly outperform RPCA on different HSIs. It means that the inherent low-rank tensor information

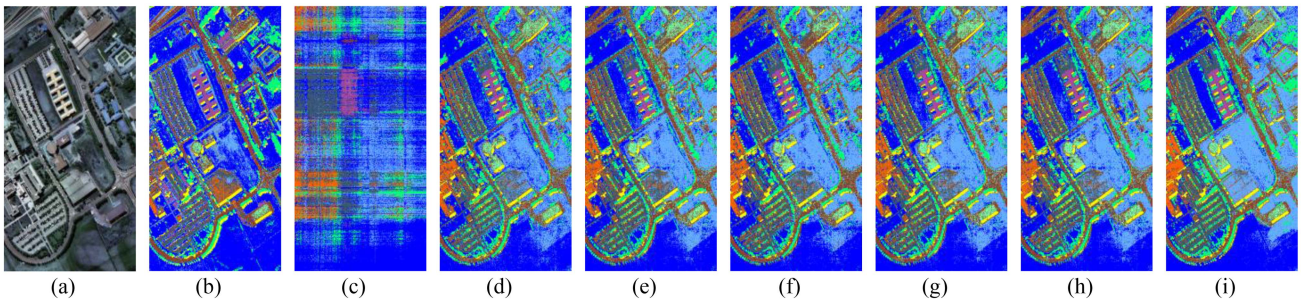

Fig. 16. Classification maps of different TRPCA-related algorithms on PU. (a) False RGB image. (b) RPCA. (c) MLTRPCA. (d) PTRPCA. (e) ETRPCA. (f) NNTRPCA. (g) ITRPCA. (h) NTRPCA. (i) SPRB-TRPCA.

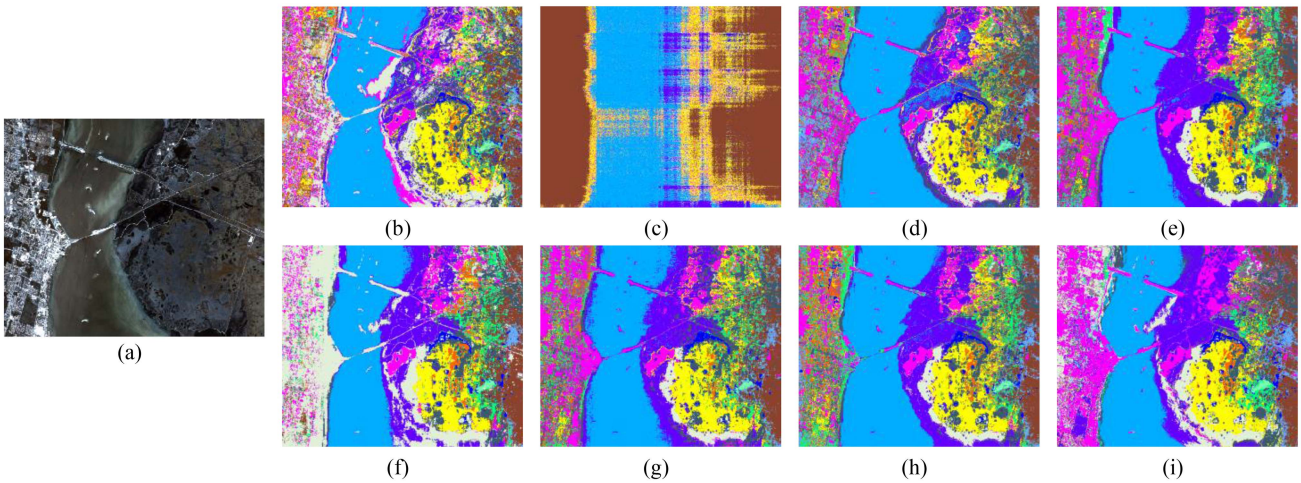

Fig. 17. Classification maps of different TRPCA-related algorithms on KSC. (a) False RGB image. (b) RPCA. (c) MLTRPCA. (d) PTRPCA. (e) ETRPCA. (f) NNTRPCA. (g) ITRPCA. (h) NTRPCA. (i) SPRB-TRPCA.

across pixels and channels needs to be considered during the feature extraction for pixel classification. Furthermore, the most crucial observation is that the proposed TRPCA-built algorithm outperforms other methods, which may benefit from sufficient informative features extracted and harmful noises removed. For instance, in Table I, it acquires the highest overall accuracy, around 3% better than the second-best results of NTRPCA in OA, AA, and Kappa. In addition, our proposal also holds the most elevated category accuracy on most land-cover objects in CA. In some classes of Indian Pines, such as Corn-N-T, Corn-M-T, Soybean-N-T, and Soybean-M-T, the CA values obtained by our model are at least around 4% better than the comparative models. Likewise, the CA values acquired by our model are at least 7% better than the relative algorithms in Cotton and Narrow-L-S of Longkou and at least 16% better than the comparative algorithms in Vinyard-U of Salinas. Such advantages also demonstrate in Botswana, PU, and KSC. The proposed model is always superior to others. Moreover, it is worth noting that our model usually obtains the highest CA in more categories, while others are much less. In Table V, we find that SPRB-TRPCA acquires the best performances in seven classes while there are nine classes. Such a phenomenon indicates that our designed model is adequate and robust to different HSIs.

In Figs. 12–17, each homogrouped subfigure is acquired by the same training and testing pixel set. As seen in these subfigures, our proposal got the optimal visual classification performance among the tested algorithms. Our model incorrectly classified fewer pixels than the other algorithms suffering from more errors and outliers. From Figs. 12 to 17, it should be noted that the classification maps of all pixels acquired by our model are the ones that are closest to the false RGB image, also demonstrating the excellent feature-extracting performance to identify a bunch of unknown pixels and the high quality of SPRB-TRPC. The constant outstanding results powerfully validate that our model is effective, robust, and adaptive to different HSIs.

IV. Discussion

One of the contributions of our model is to maintain the homogeneity and heterogeneity of spatial and spectral dimensions to promote the quality of the remaining features. Here, to further evaluate the quality of different feature sets, we also employ a t-distributed stochastic neighbor embedding (t-SNE) algorithm [41]. The t-SNE technique can intuitively reflect whether the obtained features are beneficial to identify labeled spatial pixels precisely by displaying the pixel distribution maps. Each map corresponds to the feature quality of resulting features by one extracting model, where one color represents one labeled category. The visualization maps of RPCA, MLTRPCA, PTRPCA, ETRPCA, NNTRPCA, ITRPCA, and NTRPCA on Indian Pines and Salinas are illustrated in Fig. 18 and Fig. 19, respectively.

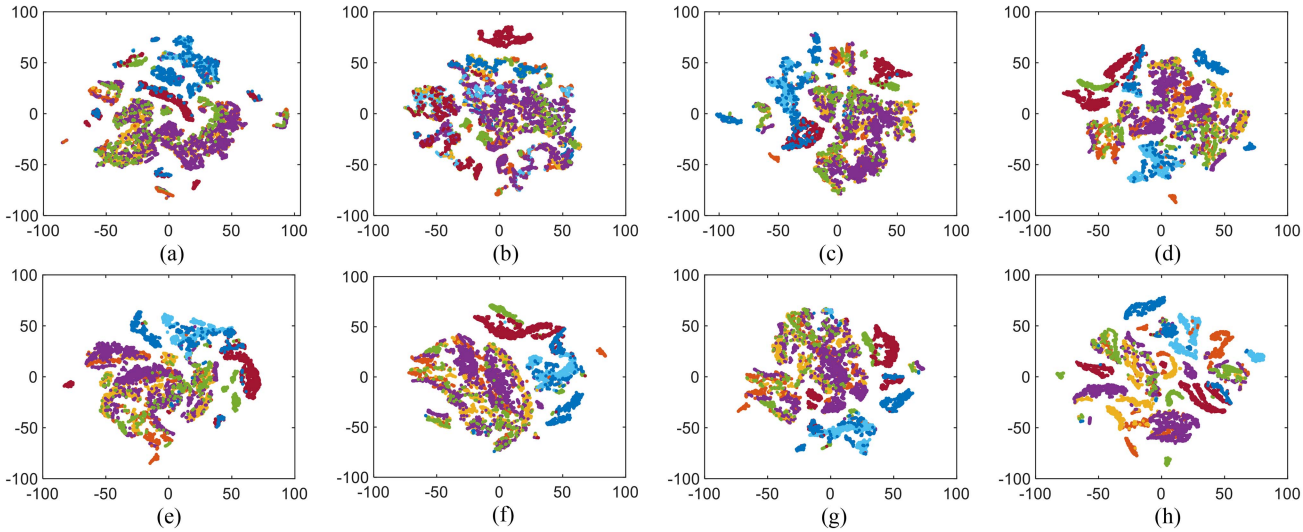

Fig. 18. Visualization by t-SNE algorithm of extracting features in (a) RPCA, (b) MLTRPCA, (c) PTRPCA, (d) ETRPCA, (e) NNTRPCA, (f) ITRPCA, (g) NTRPCA, and (h) SPRB-TRPCA on Indian Pines.

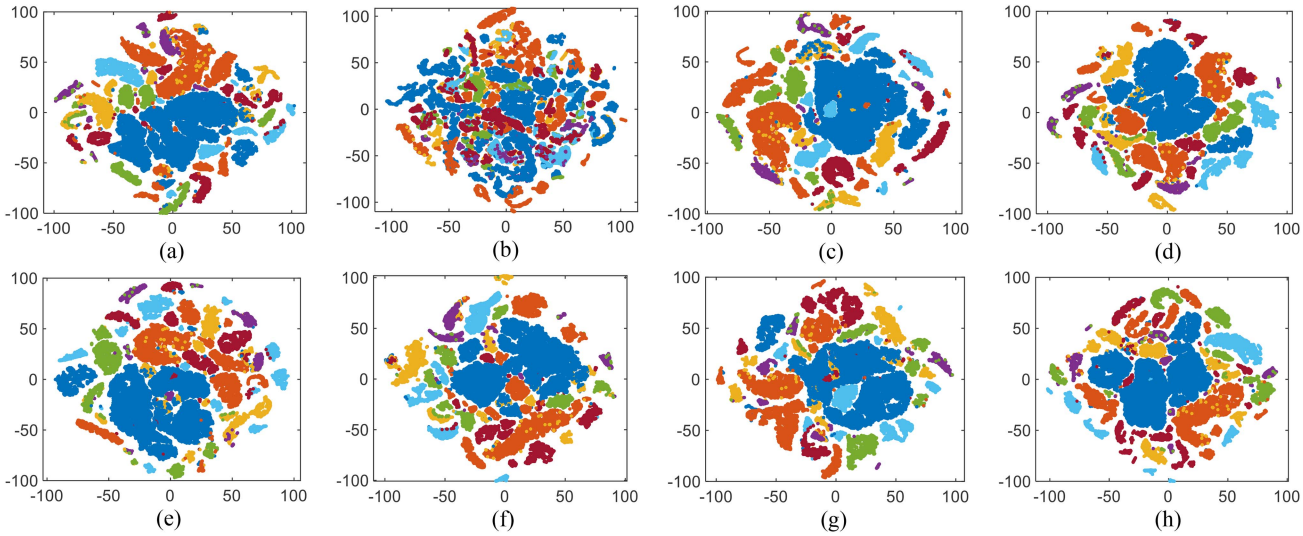

Fig. 19. Visualization by t-SNE algorithm of extracting features in (a) RPCA, (b) MLTRPCA, (c) PTRPCA, (d) ETRPCA, (e) NNTRPCA, (f) ITRPCA, (g) NTRPCA, and (h) SPRB-TRPCA on Salinas.

We only display the images of two hyperspectral datasets because of limited space. The experimental results are like the classification maps; we generally generate the best results. It can be seen from Fig. 19 that the labeled pixel distribution of our model is more prominent, clear, and compact, significantly outperforming other techniques, verifying the effectiveness and adaptivity of the designed SPRB-TRPCA to extracting features from the vast and high-dimensional HSI datasets.

V. Conclusion

This article presented a novel data-model-driven local TRPCA model for HSIs in an elaborately divide-and-rule manner, yielding stable and brilliant results in low-rank feature extracting and generating hybrid spatial–spectral frontal slices-based local 3-D block. As a result, the primary homogeneous contextual information is sufficiently and efficiently remained. In six publicly available hyperspectral datasets, the proposed model constantly shows its robustness, efficiency, effectiveness, and adaptivity to different HSIs than other advanced model-driven TRPCA algorithms.

In the future, we will focus on applying the effective and efficient data-model-driven feature-extracting technique in other remote sensing data, including space–time remote sensing datasets, or optimizing it to be a vital component of an integrating network for other HSI application fields, such as object recognition and change detection. Furthermore, we would try our best to determine how to employ such homogeneous reconstruction ideas on other huge data-derived and complicated models to improve analysis accuracy and execution speed.

REFERENCES

[1] P. H. Akwensi, Z. Kang, and R. Wang, "Hyperspectral image-aided LiDAR point cloud labeling via spatio-spectral feature representation learning," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 120, Jun. 2023, Art. no. 103302.

[2] W. N. Khotimah, M. Bennamoun, F. Boussaid, L. Xu, D. Edwards, and F. Sohel, "MCE-ST: Classifying crop stress using hyperspectral data with a multiscale conformer encoder and spectral-based tokens," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 118, Apr. 2023, Art. no. 103286.

[3] X. Wang, L. Qian, J. Shi, M. Hong, and J. Cao, "An efficient dual feature fusion model of feature extraction for hyperspectral images," *Int. J. Remote Sens.*, vol. 44, no. 14, pp. 4217–4238, Jul. 2023.

[4] P. Duan, J. Lai, J. Kang, X. Kang, P. Ghamisi, and S. Li, "Texture-aware total variation-based removal of sun glint in hyperspectral images," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 359–372, Aug. 2020.

[5] A. M. Teodoro, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Block-Gaussian-mixture priors for hyperspectral denoising and inpainting," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2478–2486, Mar. 2021.

[6] J. Penget 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.

[7] Q. Zhang, Y. Zheng, Q. Yuan, M. Song, H. Yu, and Y. Xiao, "Hyperspectral image denoising: From model-driven, data-driven, to model-data-driven," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2023.3278866.

[8] M. Esmaeili, D. Abbasi-Moghadam, A. Sharifi, A. Tariq, and Q. Li, "ResMorCNN model: Hyperspectral images classification using residual-injection morphological features and 3DCNN layers," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, no. 17, pp. 219–243, Oct. 2024.

[9] X. Li, M. Ding, Y. Gu, and A. Pižurica, "An end-to-end framework for joint denoising and classification of hyperspectral images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 7, pp. 3269–3283, Jul. 2023.

[10] P. Luet al., "Spectral–spatial and superpixelwise unsupervised linear discriminant analysis for feature extraction and classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Nov. 2023, Art. no. 5530515.

[11] Y. Chen, X. Cao, Q. Zhao, D. Meng, and Z. Xu, "Denoising hyperspectral image with non-i.i.d. noise structure," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 1054–1066, Mar. 2018.

[12] Y. Long, X. Wang, M. Xu, S. Zhang, S. Jiang, and S. Jia, "Dual self-attention Swin transformer for hyperspectral image super-resolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, May 2023, Art. no. 5512012.

[13] Y. Chen, W. Cao, L. Pang, and X. Cao, "Hyperspectral image denoising with weighted nonlocal low-rank model and adaptive total variation regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Oct. 2022, Art. no. 5544115.

[14] A. Maffei, J. M. Haut, M. E. Paoletti, J. Plaza, L. Bruzzone, and A. Plaza, "A single model CNN for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2516–2529, Apr. 2020.

[15] M. M. Salut and D. V. Anderson, "Randomized tensor robust PCA for noisy hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, Mar. 2023, Art. no. 5502405.

[16] Q. Xiao, L. Zhao, S. Chen, and X. Li, "Robust tensor low-rank sparse representation with saliency prior for hyperspectral anomaly detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Nov. 2023, Art. no. 5529920.

[17] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 208–220, Jan. 2013.

[18] D. Driggs, S. Becker, and J. Boyd-Graber, "Tensor robust principal component analysis: Better recovery with atomic norm regularization," *Adv. Neural Inform. Process. Syst.*, Jan. 2019, doi: 10.48550/arXiv.1901.10991.

[19] X. Bai, F. Xu, L. Zhou, Y. Xing, L. Bai, and J. Zhou, "Nonlocal similarity based nonnegative tucker decomposition for hyperspectral image denoising," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 701–712, Mar. 2018.

[20] C. Lu, X. Peng, and Y. Wei, "Low-rank tensor completion with a new tensor nuclear norm induced by invertible linear transforms," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, 2019, pp. 5989–5997.

[21] X. Geng, Q. Guo, and C. Zhang, "Color image denoising via tensor robust PCA with nonconvex and nonlocal regularization," *ACM Multimedia Asia Assoc. Comput. Mach.*, vol. 53, pp. 1–5, Dec. 2021.

[22] Z. Zhang, G. Ely, S. Aeron, N. Hao, and M. Kilmer, "Novel methods for multilinear data completion and de-noising based on tensor-SVD," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, 2014, pp. 3842–3849.

[23] 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, Apr. 2020.

[24] Q. Gao, P. Zhang, W. Xia, D. Xie, X. Gao, and D. Tao, "Enhanced tensor RPCA and its application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 2133–2140, Jun. 2021.

[25] Y. Liu, L. Chen, and C. Zhu, "Improved robust tensor principal component analysis via low-rank core matrix," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 6, pp. 1378–1389, Dec. 2018.

[26] W. Sun, G. Yang, J. Peng, and Q. Du, "Lateral-slice sparse tensor robust principal component analysis for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 1, pp. 107–111, Jan. 2020.

[27] Y.-J. Deng, H. -C. Li, S.-Q. Tan, J. Hou, Q. Du, and A. Plaza, "t-linear tensor subspace learning for robust feature extraction of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Jan. 2023, Art. no. 5501015.

[28] Y. Wang, T. Li, L. Chen, Y. Yu, Y. Zhao, and J. Zhou, "Tensor-based robust principal component analysis with locality preserving graph and frontal slice sparsity for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Jul. 2021, Art. no. 5508319.

[29] X. Wang, L. Qian, M. Hong, and Y. Liu, "Dual homogeneous patches-based band selection methodology for hyperspectral classification," *Remote Sens.*, vol. 15, no. 15, Aug. 2023, Art. no. 3841.

[30] M. -Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy rate superpixel segmentation," in *Proc. CVPR 2011*, Colorado Springs, CO, USA, 2011, pp. 2097–2104.

[31] C. Lu, J. Feng, S. Yan, and Z. Lin, "A unified alternating direction method of multipliers by majorization minimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 527–541, Mar. 2018.

[32] K. Joo, T. -H. Oh, and I. S. Kweon, "L1-based photometric stereo via augmented Lagrange multiplier method," in *Proc. 10th Int. Conf. Ubiquitous Robots Ambient Intell.*, Jeju, South Korea, 2013, pp. 615–616.

[33] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 5249–5257.

[34] M. F. Baumgardner, L. L. Biehl, and D. A. Landgrebe, "220 band AVIRIS hyperspectral image data set: June 12, 1992 Indian Pine test site 3," Purdue Univ. Res. Repos., 2015, doi: 10.4231/R7RX991C.

[35] Y. Zhonget al., "Mini-UAV-borne hyperspectral remote sensing: From observation and processing to applications," *IEEE Geosci. Remote Sens. Mag.*, vol. 6, no. 4, pp. 46–62, Dec. 2018.

[36] Y. Zhong, X. Hu, C. Luo, X. Wang, J. Zhao, and L. Zhang, "WHU-Hi: UAV-borne hyperspectral with high spatial resolution (H2) benchmark datasets and classifier for precise crop identification based on a deep convolutional neural network with CRF," *Remote Sens. Environ.*, vol. 250, Dec. 2020, Art. no. 112012.

[37] Hyperspectral Remote Sensing Scenes, Accessed on: Sep. 29, 2023. [Online]. Available: http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

[38] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, 2011, Art. no. 27.

[39] J. Nowosad and T. F. Stepinski, "Extended SLIC superpixels algorithm for applications to non-imagery geospatial rasters," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 112, Aug. 2022, Art. no. 102935.

[40] "Tensor robust principal component analysis via weighted tensor Schatten p norm and Lp norm," Accessed on: Sep. 29, 2023. [Online]. Available: https://github.com/qguo2010/p-TRPCA

[41] M. Liu, H. Pan, H. Ge, and Li. Wang, "MS3Net: Multiscale stratified-split symmetric network with quadra-view attention for hyperspectral image classification," *Signal Process.*, vol. 212, Nov. 2023, Art. no. 109153.

Longxia Qian received the B.A.Sc. degree in mathematics from Anhui Normal University, Wuhu, China, in 2006, the M.A.Sc. degree in hydrology and water resources from Beijing Normal University, Beijing, China, in 2009, and the Ph.D. degree in atmospheric science from the National University of Defense Technology, Nanjing, China, in 2018.

In 2019, she joined the School of Science, Nanjing University of Posts and Telecommunications, as an Associate Professor. Recently, she has been mainly engaged in feature extraction, remote sensing imagery processing, and risk warning.

Xianyue Wang received the B.A.Sc. degree in statistics from Huaqiao University, Xiamen, China, in 2021. She is currently working toward the M.A.Sc. degree in applied statistics with the School of Science, Nanjing University of Posts and Telecommunications, Nanjing, China.

Her research interests include feature extraction, data mining, and hyperspectral image processing.

Mei Hong received the B.A.Sc. degree in marine science and the Ph.D. degree in atmospheric science from the College of Meteorology and Oceanography, PLA University of Technology, Nanjing, China, in 2004 and 2009, respectively.

She is an Associate Professor with the Department of Marine Science, College of Meteorology and Oceanography, National University of Defense Science and Technology, Changsha, China. Her research interests include artificial intelligence forecasting, ocean acoustics, marine environmental risk assessment and decision-making, and climate change.

Hongrui Wang received the B.A.Sc. degree in applied mathematics from the Department of Mathematics, Beijing Normal University, Beijing, China, in 1985, the M.A.Sc. degree in environmental science from the Institute of Environmental Sciences, Beijing Normal University, Beijing, China, in 1992, and the Ph.D. degree in hydraulic engineering from the School of Environment, Beijing Normal University, Beijing, China, in 2004.

He is currently a Professor and Doctoral Supervisor with the Institute of Water Science, Beijing Normal University. His research interests include data mining, hydrology and water resources, and environmental planning assessment.

Yongchui Zhang received the B.A.Sc., M.A.Sc. and the Ph.D. degrees in atmospheric science from the College of Meteorology and Oceanography, PLA University of Technology, Nanjing, China, in 2004, 2007, and 2010, respectively.

He is an Associate Professor with the Department of Marine Science, College of Meteorology and Oceanography, National University of Defense Science and Technology, Changsha, China. He is mainly engaged in researching ocean mesoscale processes and their safeguard.