A Tensor Subspace Representation-Based Method for Hyperspectral Image Denoising

Jie Lin, *Graduate Student Member, IEEE*, Ting-Zhu Huang, Xi-Le Zhao, Tai-Xiang Jiang, *Member, IEEE*, and Lina Zhuang, *Member, IEEE*

***Abstract*— In hyperspectral image (HSI) denoising, subspace-based denoising methods can reduce the computational complexity of the denoising algorithm. However, the existing matrix subspaces, which are generated by the unfolding matrix of the HSI tensor, cannot completely represent a tensor since the unfolding operation will destroy the tensor structure. To overcome this, we design a novel basis tensor that is directly learned from the original tensor and present a tensor subspace representation (TenSR), which is a more authentic representation for delivering the intrinsic structure of the tensor than a matrix subspace representation. Equipped with the TenSR, we then propose a TenSR-based HSI denoising (TenSRDe) model, which simultaneously considers the low-tubal rankness of the HSI tensor and the nonlocal self-similarity of the coefficient tensor. Moreover, we develop an efficient proximal alternating minimization (PAM) algorithm to solve the proposed nonconvex model and theoretically prove that the algorithm globally converges to a critical point. Experiments implemented on simulated and real data sets substantiate the denoising effect and efficiency of the proposed method.**

***Index Terms*— Hyperspectral image (HSI) denoising, proximal alternating minimization (PAM), tensor singular value decomposition (t-SVD), tensor subspace representation (TenSR).**

I. Introduction

HYPERSPECTRAL images (HSIs), which are essentially third-order tensors, contain three intrinsic components, namely, spectrum, spatial width, and spatial height. Due to their multidimensional information-preserving capability and high spectral resolution, HSIs can deliver more faithful knowledge in the real sense and have been widely used in various fields, such as food safety, mineral exploration, agricultural production, and urban planning [1]–[4]. However, because of the acquisition errors caused by photon effects and sensor disturbance, HSIs are unavoidably polluted by the Gaussian noise, which severely hinders subsequence applications, such as classification, unmixing, and target detection [5]–[8]. Therefore, denoising has become an indispensable preprocessing step for the further analysis and application of HSIs.

As more attention is paid to HSI denoising, a multitude of methods have been proposed. According to the way of processing noisy HSIs, the existing HSI denoising methods can be formally described in two ways: direct approach and subspace representation approach.

1) *Direct Approaches:* First, the traditional 2-D denoising methods were extended to HSI denoising by treating each spectral band as a separate grayscale image with nonlocal self-similarity (NSS) across the space [9]. We refer to the representative methods, such as block-matching 3-D (BM3D) filtering [10], nonlocal Bayes (NL-Bayes) [11], and weighted nuclear norm minimization (WNNM) [12]. Nevertheless, an HSI also possesses global correlation along the spectrum (GCS) [13], and the extended 2-D methods cannot explore it. To take full advantage of the redundancy across the spectral dimension, i.e., GCS, many low-rank-based methods have been proposed. Zhang *et al.* [14] converted a 3-D HSI into a matrix and proposed a low-rank matrix recovery model. Considering the spatial piecewise smooth structure of HSIs, He *et al.* [15] proposed total variation (TV)-regularized low-rank matrix factorization. To obtain a better low-rank approximation, Xie *et al.* [16] applied a nonconvex low-rank regularizer, i.e., the weighted Schatten p-norm (WSN), to achieve a better approximation to the original low-rank assumption; Chen *et al.* [17] utilized a nonconvex norm to approximate the rank function and proposed a nonconvex low-rank approximation method, gaining improved denoising effect. However, an HSI is essentially a third-order tensor, which means that matricization will destroy its intrinsic structure. The tensor technique, which can finely preserve the tensor structure, has achieved great success in the field of HSI processing [18]–[24]. Based on the Tucker decomposition [25], Renard *et al.* [26] used the Tucker rank [27] to describe the correlation of different modes and presented a low-rank tensor approximation method for HSI denoising. To address the uniqueness of the Tucker

Manuscript received February 18, 2020; revised June 18, 2020, July 31, 2020, and September 23, 2020; accepted October 14, 2020. Date of publication November 10, 2020; date of current version August 30, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61772003, Grant 61876203, and Grant 12001446, in part by the Key Projects of Applied Basic Research in Sichuan Province under Grant 2020YJ0216, and in part by the Fundamental Research Funds for the Central Universities under Grant JBK2001035 and Grant JBK2001011. *(Corresponding authors: Ting-Zhu Huang; Xi-Le Zhao.)*

Jie Lin, Ting-Zhu Huang, and Xi-Le Zhao are with the School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: jielin96@126.com; tingzhuhuang@126.com; xlzhao122003@163.com).

Tai-Xiang Jiang is with FinTech Innovation Center, Financial Intelligence and Financial Engineering Research Key Laboratory of Sichuan province, School of Economic Information Engineering, Southwestern University of Finance and Economics, Chengdu 611130, China (e-mail: taixiangjiang@gmail.com).

Lina Zhuang is with the Department of Mathematics, Hong Kong Baptist University, Hong Kong (e-mail: linazhuang@hkbu.edu.hk).

Digital Object Identifier 10.1109/TGRS.2020.3032168

1558-0644 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

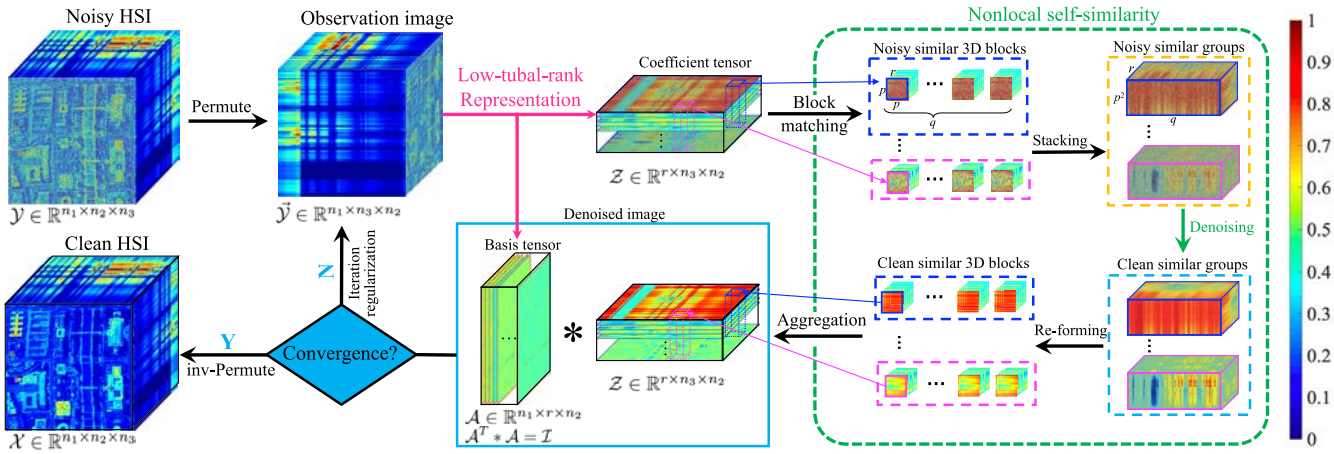


Fig. 1. Flowchart of the proposed TenSRDe method for HSI denoising.

decomposition and multirank estimation, Liu *et al.* [28] utilized parallel factor (PARAFAC) analysis [29] to reconstruct HSIs and reduced the number of estimated ranks to one; however, computing CP rank is NP-hard. To handle previous problems, Fan *et al.* [30] recovered HSIs by a computable tensor singular value decomposition (t-SVD) [31] and achieved improvements in effect and efficiency. Recently, deep learning-based methods for HSI denoising, such as convolutional neural network-based methods [2], [3] and spatial–spectral gradient network-based method [32], have been applied to learn deep image priors from a large number of image samples and have shown promising denoising performance benefits from its high capacity. However, it is difficult to guarantee high-quality denoising results under realistic complex noise scenarios when the underlying assumption does not hold in real images. All the abovementioned direct methods that consider the whole noisy HSI as the observation image have achieved superior performance, but they suffer from heavy computational burdens when the spectrum number and space size are increased.

2) *Subspace Representation Approaches:* The images acquired across the HSI spectrum are highly correlated. This suggests that a clean HSI lies in a low-dimensional subspace. Therefore, the spectral low-rank constraint can be imposed by using subspace representation. Assuming that the spectra in an HSI lie in a low-rank subspace, Sun *et al.* [33] proposed a novel subspace spatial–spectral low-rank learning model, which can be effectively solved by a cyclic descent algorithm, gaining significant improvement in the denoising effect. Zhuang *et al.* [34] learned a matrix subspace from the Casorati matrix by the Hysime algorithm and presented a fast hyperspectral denoising (FastHyDe) method, achieving advanced denoising performance with lower computational complexity. He *et al.* [35] proposed an integrated paradigm that learns a matrix subspace with increasing dimensions and uses iterative refinement to boost the denoising performance. Based on matrix subspace representation, Cao *et al.* [36] presented a subspace-based nonlocal low-rank and sparse factorization (SNLRSF) method, which takes both the spectral low rankness and the spatial NSS into consideration. The aforementioned matrix subspace representation methods project the rearranged 2-D matrix onto the low-dimensional spectral subspace, gaining outstanding performance/efficiency tradeoffs. However, since the matricization can destroy the spatial information of pixels within one band, such approaches lead to the loss of multidimensional structure information [30], and the learned subspaces only explore the low rankness of the spectral dimension. Hence, the potential capacity of the subspace representation still has room to be further enhanced.

A. Motivation

To improve the representation capacity of the subspace, we first design a basis tensor and develop a tensor subspace representation (TenSR). The existing matrix subspace is generated by the unfolding matrix of the tensor along the spectral dimension, which only considers the GCS of an HSI. However, an HSI is essentially a third-order tensor, and the tensor along each mode resides on a low-rank subspace [37], [38], which means that the HSI has a global correlation in spectrum and space (GCSS). Since tensor decomposition can adequately exploit the GCSS of an HSI [39], [40], we advocate that the subspace of the HSI should be learned by a tensor tool. Recently, Kilmer *et al.* [31] proposed a new tensor–tensor multiplication named the tensor product (t-product), which avoids the loss of information inherent in flattening the tensors [41], and a novel tensor decomposition form named t-SVD (see Theorem 1). As the natural generalization of the matrix SVD, t-SVD has the optimal tubal rank (see Definition 5), which is similar to the matrix rank derived from SVD [42]. Moreover, since t-SVD can be computed efficiently by applying the fast Fourier transform (FFT) along each tube of the tensor and tubal rank is calculated from the original tensor, the tubal rank is more computable than the CP rank and more intrinsic than the Tucker rank. Thus, we utilize t-SVD to learn the initial basis tensor and use the

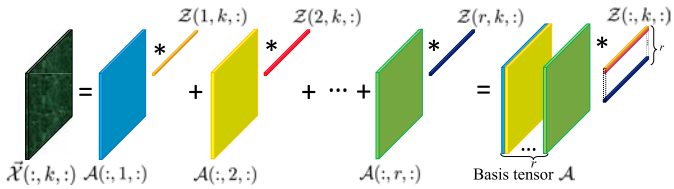


Fig. 2. kth lateral slice of $\vec{\mathcal{X}}$, i.e., the kth band of HSI, is represented by a t-linear [41] combination of the r bases and coefficients.

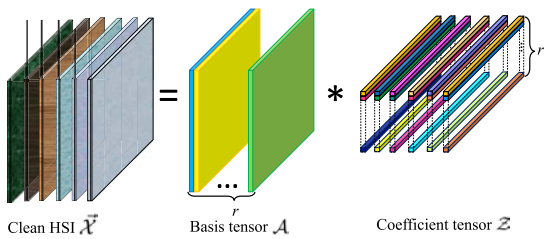


Fig. 3. Clean HSI is represented by the basis tensor and coefficient tensor.

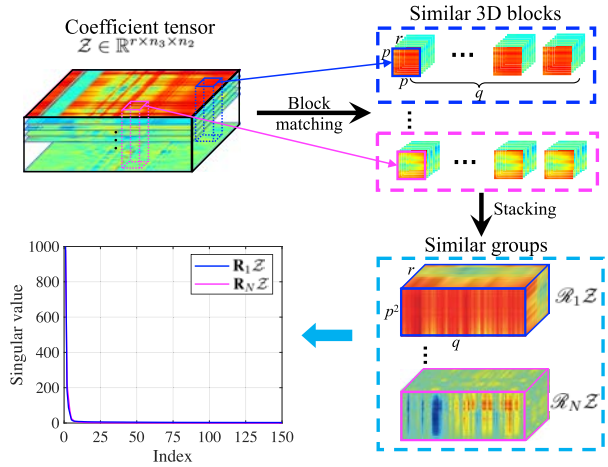


Fig. 4. NSS property of the coefficient tensor. ($\mathbf{R}_i\mathcal{Z}$ is the unfolding matrix of $\mathscr{R}_i\mathcal{Z}$ along mode-2.)

tubal rank to describe the subspace dimension. Using t-SVD and the t-product, the basis tensor $\mathcal{A}$ is learned from the permuted tensor $\vec{\mathcal{X}} = \text{permute}(\mathcal{X}, [1, 3, 2]) \in \mathbb{R}^{n_1\times n_3\times n_2}$, and each band of the HSI can be represented by the finite bases and coefficients (see Fig. 2). Then, the whole HSI can be represented by the basis tensor and coefficient tensor, as shown in Fig. 3.

Based on TenSR, the coefficient tensor, obtained by projecting the HSI onto the basis tensor, inherits the spectral–spatial NSS property of the HSI. From Fig. 4, we can observe the NSS property of the coefficient tensor, which is characterized by the low rankness of the unfolding matrix formed from similar 3-D blocks in coefficient tensor.

To take both the GCSS and NSS prior knowledge into account, we propose a TenSR-based HSI denoising (TenSRDe) method. The proposed method explores the GCSS property of an HSI by TenSR and exploits the spectral–spatial NSS property of the coefficient tensor by using the weighted nuclear norm to characterize the low rankness of the unfolding matrix of each similar group. In summary, the whole denoising flowchart is shown in Fig. 1.

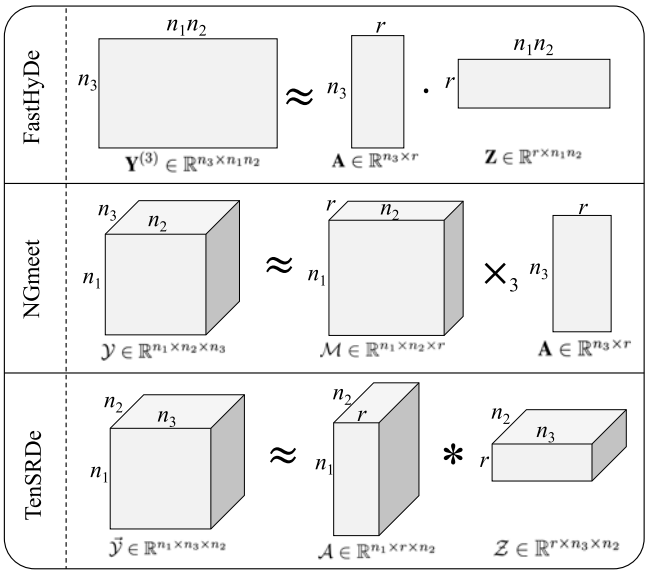


Fig. 5. Illustration of the representation forms of FastHyDe, NGmeet, and TenSRDe.

Remark 1: The proposed TenSRDe is distinct from FastHyDe [34] and NGmeet [35]. For deeper insight into the three methods, we show their representation forms in Fig. 5. Compared with the matrix subspace of FastHyDe, which is based on the assumption that the clean HSI lies in a low-rank matrix subspace and can be represented by a linear combination of finite endmembers, the tensor subspace of TenSRDe is based on the assumption that the clean HSI lies in a low-rank tensor subspace and the clean HSI can be t-linearly [41] represented by the finite bases of the basis tensor (see Fig. 3). Compared with NGmeet, TenSRDe contains two differences in representation form. First, TenSR in TenSRDe is a more complete tensor representation form. As shown in Fig. 5, both representation factors of TenSR are tensors. Especially, when we keep the first frontal slice of $\mathcal{Z} \in \mathbb{R}^{r\times n_3\times n_2}$ unchanged and set other frontal slices to zero, the representation form $\mathcal{A}*\mathcal{Z}$ is equivalent to $\mathcal{M}\times_3\mathbf{A}$, i.e., the representation form of NGmeet is a special case of TenSR. Second, TenSR has a more inherent tensor operation. Compared with the tensor-matrix multiplication "3-mode product" in NGmeet, the "t-product" in TenSR is a tensor–tensor multiplication and can better preserve the tensor intact, which allows TenSR to achieve more information interchange between two factors.

B. Contributions

This article makes the following three contributions.

1) Inspired by the t-SVD, we design a novel basis tensor and develop a tensor subspace representation. Based on the new tensor decomposition t-SVD, the basis tensor can be learned directly and efficiently from the HSI tensor rather than its unfolding matrix. Benefitting from better basis tensor and natural tensor–tensor multiplication "t-product," the developed TenSR can faithfully deliver the intrinsic structure of the HSI tensor.
2) Equipped with TenSR, we propose a TenSR-based HSI denoising model for the Gaussian noise removal.

Compared with direct approaches that only consider GCS or NSS, the proposed model takes both the GCSS and NSS into consideration. Moreover, the TenSR in the proposed model degrades the large-scale tensor into a small-scale coefficient tensor, which significantly reduces the computational complexity in the denoising procedure.

3) We give the closed-form solution of formula (13) with the tensor orthogonal constraint and develop an efficient proximal alternating minimization (PAM) algorithm to solve the proposed nonconvex model. We theoretically prove that our algorithm globally converges to a critical point. Extensive experiments indicate that the proposed method improves the denoising effect and efficiency in comparison with state-of-the-art methods.

C. Organization

The remainder of this article is organized as follows. Section II gives the notations and definitions. Section III proposes the TenSRDe model and corresponding PAM algorithm with convergence proof. Section IV reports the experimental results and discussion. Section V concludes this article.

II. Notations and Definitions

A. Notations

We use lowercase letters for scalars (e.g., a), boldface lowercase letters for vectors (e.g., $\mathbf{a}$), boldface capital letters for matrices (e.g., $\mathbf{A}$), and calligraphic letters for tensors (e.g., $\mathcal{A}$). The real and complex number fields are denoted as $\mathbb{R}$ and $\mathbb{C}$, respectively. We denote the element value of kth-order tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ as $a_{i_1, i_2, \ldots, i_k}$. Especially, for third-order tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we use MATLAB notation $\mathcal{A}(:, :, i)$, $\mathcal{A}(:, i, :)$, and $\mathcal{A}(i, :, :)$ for its ith frontal, lateral, and horizontal slice (see [27] for definitions), respectively, $\mathcal{A}(:, i, j)$, $\mathcal{A}(i, :, j)$, and $\mathcal{A}(i, j, :)$ for its (i, j)th mode-1, mode-2, and mode-3 fibers [27], respectively. For conciseness, we use compact form $\mathbf{A}_{(i)}$ to denote frontal slice $\mathcal{A}(:, :, i)$ and utilize a tube to represent the mode-3 fiber. The unfolding matrix along mode-2 and mode-3 of $\mathcal{A}$ is $\mathbf{A}^{(2)} = \text{unfold}_2(\mathcal{A}) \in \mathbb{R}^{n_2 \times n_3 n_1}$ and $\mathbf{A}^{(3)} = \text{unfold}_3(\mathcal{A}) \in \mathbb{R}^{n_3 \times n_1 n_2}$, respectively. We use $\hat{\mathcal{A}}$ to denote the tensor generated by performing the discrete Fourier transformation (DFT) along each tube of $\mathcal{A}$, i.e., $\hat{\mathcal{A}} = \texttt{fft}(\mathcal{A}, [_], 3)$. We define a permuted tensor of $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ as $\vec{\mathcal{X}} := \text{Permute}(\mathcal{X}, [1, 3, 2]) \in \mathbb{R}^{n_1 \times n_3 \times n_2}$. The operator "Permute" transforms the frontal slice of $\mathcal{X}$ into the lateral slice of $\vec{\mathcal{X}}$, and its inverse operation "inv-Permute" is defined as $\mathcal{X} := \text{inv-Pemute}(\vec{\mathcal{X}}, [1, 3, 2])$. The nuclear norm and weighted nuclear norm of matrix $\mathbf{A}$ are defined as $\|\mathbf{A}\|_* = \sum_i \sigma_i(\mathbf{A})$ and $\|\mathbf{A}\|_{\mathbf{w},*} = \sum_i w_i \sigma_i(\mathbf{A})$, respectively, where $\mathbf{w} = (w_1, w_2, \ldots, w_n)$, $w_i \geq 0$, and $\sigma_i(\mathbf{A})$ is the ith largest singular value of $\mathbf{A}$. The Frobenius norm of $\mathcal{A}$ is defined as $\|\mathcal{A}\|_F = \sqrt{\sum_{i,j,k} |a_{i,j,k}|^2}$.

B. Generalized Definitions

We provide the following definitions [31] related to our work.

Definition 1 (t-Product): Given two third-order tensors $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\mathcal{B} \in \mathbb{R}^{n_2 \times n_4 \times n_3}$. Then, the t-product $\mathcal{A} * \mathcal{B}$ is tensor $\mathcal{C} \in \mathbb{R}^{n_1 \times n_4 \times n_3}$

$$\mathcal{C}(i, j, :) = \sum_{k=1}^{n_2} \mathcal{A}(i, k, :) * \mathcal{B}(k, j, :).$$

Definition 2 (Transpose Tensor): For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, transpose tensor $\mathcal{A}^T \in \mathbb{R}^{n_2 \times n_1 \times n_3}$ is obtained by transposing each of the frontal slices and then reversing the order of transposed frontal slices 2 through n.

Definition 3 (Identity Tensor): The identity tensor $\mathcal{I} \in \mathbb{R}^{m \times m \times n_3}$ is the tensor whose first frontal slice is the $m \times m$ identity matrix and whose other frontal slices are all zeros.

Definition 4 (Orthogonal Tensor): A tensor $\mathcal{A} \in \mathbb{R}^{m \times m \times n_3}$ is orthogonal if $\mathcal{A}^T * \mathcal{A} = \mathcal{A} * \mathcal{A}^T = \mathcal{I}$.

Theorem 1 (t-SVD [31]): For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the t-SVD of $\mathcal{A}$ is given by

$$\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$$

where $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is an f-diagonal tensor, that is, each frontal slice of $\mathcal{S}$ is a diagonal matrix, and $\mathcal{U} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ and $\mathcal{V} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$ are orthogonal tensors.

Definition 5 (Tensor Tubal Rank [43]): For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the tensor tubal rank of $\mathcal{A}$, denoted as $\text{rank}_t(\mathcal{A})$, is defined as the number of nonzero singular tubes of $\mathcal{S}$, that is

$$\text{rank}_t(\mathcal{A}) = \#\{i, \mathcal{S}(i, i, :) \neq 0\}$$

where $\mathcal{S}$ is from the t-SVD of $\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$.

We next provide some definitions in our theoretical proof of the algorithm convergence. Let us recall a few basic definitions of the Kurdyka–Łojasiewicz (KL) property. For a proper and lower semicontinuous function $\Psi : \mathbb{R}^n \to (-\infty, +\infty]$, the effective domain of Ψ is defined as

$$\text{dom}(\Psi) := \left\{\mathbf{x} \in \mathbb{R}^n : \Psi(\mathbf{x}) < +\infty\right\}.$$

For any subset $S \in \mathbb{R}^n$ and any point $\mathbf{x} \in \mathbb{R}^n$, the distance from $\mathbf{x}$ to S is defined as

$$\text{dist}(\mathbf{x}, S) := \inf\{\|\mathbf{y} - \mathbf{x}\| : \mathbf{y} \in S\}.$$

We now introduce the definition of the KL property and semialgebraic function, which is essential to our convergence analysis.

Definition 6 (KL Property [44]): Let $\Psi : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be proper and lower semicontimuous.

1) The function Ψ is said to have the KL property at $\mathbf{x}^* \in \text{dom}(\partial\Psi)$ if there exist $\eta \in (0, +\infty]$, a neighborhood U of $\mathbf{x}^*$, and a continuous concave function $\varphi : [0, \eta) \to \mathbb{R}_+$ such that the following holds.
   a) $\varphi(0) = 0$.
   b) φ is C^1 on $(0, +\infty)$.
   c) For all $s \in (0, \eta)$, $\varphi'(s) > 0$.
   d) For all $\mathbf{x}$ in $U \cap [\Psi(\mathbf{x}^*) < \Psi < \Psi(\mathbf{x}^*) + \eta]$, the KL inequality holds

$$\varphi'\big(\Psi(\mathbf{x}) - \Psi\big(\mathbf{x}^*\big)\big)\text{dist}(0, \partial\Psi(\mathbf{x})) \geq 1.$$

2) If Ψ satisfies the KL property at each point of $\text{dom}(\partial\Psi)$, then Ψ is called a KL function.

Definition 7 (Semialgebraic Sets and Functions [44]): A subset S of $\mathbb{R}^n$ is a real semialgebraic set in which there exists a finite number of real polynomial functions f_{ij}, g_{ij}: $\mathbb{R}^n \to \mathbb{R}$ such that

$$S = \bigcup_{j=1}^{p} \bigcap_{i=1}^{q} \left\{ \mathbf{x} \in \mathbb{R}^n : f_{ij}(\mathbf{x}) = 0, g_{ij}(\mathbf{x}) < 0 \right\}.$$

A function g: $\mathbb{R}^n \to (-\infty, +\infty]$ is called semialgebraic if its graph

$$\left\{ (\mathbf{x}, t) \in \mathbb{R}^{n+1} : g(\mathbf{x}) = t \right\}$$

is a semialgebraic subset of $\mathbb{R}^{n+1}$.

Definition 8 (Lipschitz Continuity [45]): Given an open set $\Omega \subseteq \mathbb{R}^n$, function F is a Lipschitz continuous on the open subset Ω if there exists a constant $l > 0$ such that

$$\|F(\mathbf{x}) - F(\mathbf{y})\| \leq l\|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in \Omega$$

l is the Lipschitz constant of F on Ω.

III. Proposed Model and Algorithm

An HSI is essentially a third-order tensor, where the spatial information and spectral information lie in the first two dimensions and the third dimension, respectively. Under the assumption that the clean HSI is corrupted by additive Gaussian noise, the observation model can be expressed as

$$\mathcal{Y} = \mathcal{X} + \mathcal{N}$$

where $\mathcal{Y}$ is the noisy HSI, $\mathcal{X}$ is the clean HSI, and $\mathcal{N}$ is the Gaussian noise with zero-mean and variance σ^2.

A. Model

From our observation, the permuted clean HSI data $\vec{\mathcal{X}}$ are more low-tubal-rank than the original HSI data $\mathcal{X}$ (see detail in Section IV-C1). We assume that there exists an optimal tubal rank $r \ll n_3$ such that $\vec{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_3 \times n_2}$ can be represented by a tensor subspace with tubal rank r. Then, the tensor subspace representation (TenSR) of $\vec{\mathcal{X}}$ can be formulated as

$$\vec{\mathcal{X}} = \mathcal{A} * \mathcal{Z}$$

where $\mathcal{A} \in \mathbb{R}^{n_1 \times r \times n_2}$ is the basis tensor composed of r bases, r controls the low rankness of $\vec{\mathcal{X}}$, and $\mathcal{Z} \in \mathbb{R}^{r \times n_3 \times n_2}$ is the coefficient tensor.

Using the TenSR to explore the low rankness of the clean HSI $\vec{\mathcal{X}}$ and utilizing the weighted nuclear norm to character the low rankness of the unfolding matrix of similar groups generated from $\mathcal{Z}$ separately, the proposed TenSR-based HSI denoising model is formulated as

$$\begin{aligned} &\min_{\mathcal{A},\mathcal{Z}} \frac{1}{2}\|\vec{\mathcal{Y}} - \mathcal{A} * \mathcal{Z}\|_F^2 + \lambda \sum_i \|\mathbf{R}_i \mathcal{Z}\|_{\mathbf{w},*} \\ &\text{s.t. } \mathcal{A}^T * \mathcal{A} = \mathcal{I} \end{aligned} \tag{1}$$

where $\lambda > 0$ is the regularization parameter, $\mathbf{R}_i \mathcal{Z}$ denotes the unfolding matrix of $\mathscr{R}_i \mathcal{Z}$ along mode-2, and $\mathscr{R}_i \mathcal{Z}$ is a tensor formed by stacking the ith group of similar 3-D blocks in $\mathcal{Z}$. The orthogonality of $\mathcal{A}$ promotes each base of $\mathcal{A}$ to be more distinct from each other, which helps keep noise out of $\mathcal{A}$. The clean HSI is recovered by $\mathcal{X} = \text{inv-Permute}(\mathcal{A}^* * \mathcal{Z}^*)$.

B. Algorithm

We develop an algorithm based on PAM to solve the objective function (1). The subproblems involved in (1) can be solved as follows.

1) $\mathcal{Z}$ Subproblem:

$$\begin{aligned} \mathcal{Z}^{k+1} = \arg\min_{\mathcal{Z}} \frac{1}{2}\|\vec{\mathcal{Y}} - \mathcal{A}^k * \mathcal{Z}\|_F^2 + \lambda \sum_i \|\mathbf{R}_i \mathcal{Z}\|_{\mathbf{w},*} \\ + \frac{\rho}{2}\|\mathcal{Z} - \mathcal{Z}^k\|_F^2 \end{aligned} \tag{2}$$

where ρ is a proximal parameter. It is difficult to solve this problem, and we give Lemma 1.

Lemma 1: [31] For third-order tensors $\mathcal{A}$ and $\mathcal{B}$, if $\mathcal{A}$ is an orthogonal tensor

$$\|\mathcal{A} * \mathcal{B}\|_F = \|\mathcal{B}\|_F.$$

According to the orthogonality of $\mathcal{A}$, by using Lemma 1, (2) can be rewritten as

$$\begin{aligned} \mathcal{Z}^{k+1} &= \arg\min_{\mathcal{Z}} \frac{1}{2}\|\left(\mathcal{A}^k\right)^T * \vec{\mathcal{Y}} - \mathcal{Z}\|_F^2 + \lambda \sum_i \|\mathbf{R}_i \mathcal{Z}\|_{\mathbf{w},*} \\ &\quad + \frac{\rho}{2}\|\mathcal{Z} - \mathcal{Z}^k\|_F^2 \\ &= \arg\min_{\mathcal{Z}} \frac{1}{2}\|\frac{\left(\mathcal{A}^k\right)^T * \vec{\mathcal{Y}} + \rho \mathcal{Z}^k}{1+\rho} - \mathcal{Z}\|_F^2 \\ &\quad + \frac{\lambda}{1+\rho} \sum_i \|\mathbf{R}_i \mathcal{Z}\|_{\mathbf{w},*}. \end{aligned} \tag{3}$$

For simplicity, we denote $\tilde{\mathcal{Z}} = [(\mathcal{A}^k)^T * \vec{\mathcal{Y}} + \rho \mathcal{Z}^k]/(1+\rho)$. We apply the alternating direction method of multipliers (ADMM) to solve problem (3), which can be guaranteed to converge globally [46]. Introducing auxiliary variables $\mathcal{M}_i = \mathcal{Z} (i = 1, 2, \ldots, N)$ to (3), we have

$$\min_{\mathcal{M}_i, \mathcal{Z}} \frac{1}{2}\|\tilde{\mathcal{Z}} - \mathcal{Z}\|_F^2 + \frac{\lambda}{1+\rho} \sum_i \|\mathbf{M}_i^{(2)}\|_{\mathbf{w},*}, \text{ s.t. } \mathbf{M}_i^{(2)} = \mathbf{R}_i \mathcal{Z}. \tag{4}$$

The augmented Lagrangian function of (4) is defined as

$$\begin{aligned} L(\mathcal{M}_i, \mathcal{Z}, \mathcal{T}_i) = \frac{1}{2}\|\tilde{\mathcal{Z}} - \mathcal{Z}\|_F^2 + \frac{\lambda}{1+\rho} \sum_i \|\mathbf{M}_i^{(2)}\|_{\mathbf{w},*} \\ + \frac{\gamma}{2} \sum_i \|\mathcal{M}_i - \mathscr{R}_i \mathcal{Z} + \frac{\mathcal{T}_i}{\gamma}\|_F^2 \end{aligned} \tag{5}$$

where $\mathcal{T}_i$ is the Lagrangian multiplier and $\gamma > 0$ is the penalty parameter. Within the ADMM framework, $\mathcal{M}_i$, $\mathcal{Z}$ and $\mathcal{T}_i$ are alternately updated as

$$\mathcal{M}_i^{p+1} = \arg\min_{\mathcal{M}_i} L\left(\mathcal{M}_i, \mathcal{Z}^p, \mathcal{T}_i^p\right) \tag{6}$$

$$\mathcal{Z}^{p+1} = \arg\min_{\mathcal{Z}} L\left(\mathcal{M}_i^{p+1}, \mathcal{Z}, \mathcal{T}_i^p\right) \tag{7}$$

$$\mathcal{T}_i^{p+1} = \mathcal{T}_i^p + \gamma\left(\mathcal{M}_i^{p+1} - \mathscr{R}_i \mathcal{Z}^{p+1}\right). \tag{8}$$

Problem (6) in the form of tensor unfolding is

$$\begin{aligned} \left(\mathbf{M}_i^{(2)}\right)^{p+1} = \min_{\mathbf{M}_i^{(2)}} \frac{\gamma(1+\rho)}{2\lambda} \left\| \mathbf{M}_i^{(2)} - \left(\mathbf{R}_i \mathcal{Z}^p - \frac{\left(\mathbf{T}_i^{(2)}\right)^p}{\gamma} \right) \right\|_F^2 \\ + \left\| \mathbf{M}_i^{(2)} \right\|_{\mathbf{w},*}. \end{aligned} \tag{9}$$

According to [12], (9) can be solved in the closed-form solution

$$\left(\mathbf{M}_i^{(2)}\right)^{p+1} = \mathbf{U}S_w(\Sigma)\mathbf{V}^T \tag{10}$$

where $\mathbf{U}$ and $\mathbf{V}$ are from the SVD of $\mathbf{R}_i\mathcal{Z}^p - (\mathbf{T}_i^{(2)})^p/\gamma = \mathbf{U}\Sigma\mathbf{V}^T$ and $S_w(\Sigma) = \mathrm{Diag}[\max(\Sigma_{ll} - \lambda w_j/[\gamma(1+\rho)], 0)]$. Then, we have $\mathcal{M}_i^{p+1}$ by folding $(\mathbf{M}_i^{(2)})^{p+1}$.

Problem (7) is

$$\mathcal{Z}^{p+1} = \min_{\mathcal{Z}} \frac{1}{2}\|\tilde{\mathcal{Z}} - \mathcal{Z}\|_F^2 + \frac{\gamma}{2}\sum_i \|\mathscr{R}_i\mathcal{Z} - \left(\mathcal{M}_i^{p+1} + \frac{\mathcal{T}_i^p}{\gamma}\right)\|_F^2. \tag{11}$$

According to [47], the solution of (11) is given by

$$\mathcal{Z}^{p+1} = \left[\tilde{\mathcal{Z}} + \gamma\sum_i \mathscr{R}_i^T\left(\mathcal{M}_i^{p+1} + \frac{\mathcal{T}_i^p}{\gamma}\right)\right] \oslash \left(\mathcal{J} + \gamma\sum_i \mathscr{R}_i^T\mathscr{R}_i\right) \tag{12}$$

where $\mathcal{J}$ is the all-ones tensor, $\mathscr{R}_i^T\mathscr{R}_i$ denotes the number of overlapping blocks that cover the pixel location, $\mathscr{R}_i^T(\cdot)$ denotes the sum of all overlapping reconstruction blocks that cover the pixel location, and $\oslash$ represents the pixel-to-pixel level division. The ADMM algorithm for solving subproblem (3) can be described in Algorithm 1.

Algorithm 1 ADMM Algorithm for Solving (3)

Input: The penalty parameter γ.

Initialize: $\mathcal{Z}^0 = \tilde{\mathcal{Z}}$, $\mathcal{T}_i^0 = 0$.
for $p = 0 : P$ **do**
  Update $\mathcal{M}_i^{p+1}$ by (10), $i = 1, 2, \ldots, N$;
  Update $\mathcal{Z}^{p+1}$ by (12);
  Update $\mathcal{T}_i^{p+1}$ by (8), $i = 1, 2, \ldots, N$.
end for

Output: The solution $\mathcal{Z}^{p+1}$.

2) $\mathcal{A}$ Subproblem:

$$\mathcal{A}^{k+1} = \arg\min_{\mathcal{A}^T*\mathcal{A}=\mathcal{I}} \frac{1}{2}\|\vec{\mathcal{Y}} - \mathcal{A}*\mathcal{Z}^{k+1}\|_F^2 + \frac{\rho}{2}\|\mathcal{A} - \mathcal{A}^k\|_F^2. \tag{13}$$

To solve (13), we introduce the following theorem.

Theorem 2: For any $\mathcal{A} \in \mathbb{R}^{n_1\times n_2\times n_3}$, the following problem:

$$\min_{\mathcal{A}} \frac{1}{2}\|\mathcal{Y} - \mathcal{A}*\mathcal{Z}\|_F^2 + \frac{\rho}{2}\|\mathcal{A} - \mathcal{B}\|_F^2, \quad \text{s.t. } \mathcal{A}^T*\mathcal{A} = \mathcal{I} \tag{14}$$

has the closed-form solution $\mathcal{A}^* = \mathcal{V}*\mathcal{U}^T$, where $\mathcal{U}$ and $\mathcal{V}$ are from the t-SVD of $\mathcal{Z}*\mathcal{Y}^T + \rho\mathcal{B}^T = \mathcal{U}*\mathcal{S}*\mathcal{V}^T$.

To prove our theorem, we give the following lemma.

Lemma 2 (Von Neuman's Trace Inequality [48]): If matrices $\mathbf{M}$ and $\mathbf{N}$ with size $p \times q$ have singular values $\sigma_1(\mathbf{M}) \geq \sigma_2(\mathbf{M}) \geq \ldots \geq \sigma_r(\mathbf{M})$ and $\sigma_1(\mathbf{N}) \geq \sigma_2(\mathbf{N}) \geq \ldots \geq \sigma_r(\mathbf{N})$, respectively, where $r = \min\{p, q\}$, then

$$\mathrm{Trace}\left(\mathbf{M}^T\mathbf{N}\right) \leq \sum_{i=1}^r \sigma_i(\mathbf{M})\sigma_i(\mathbf{N}).$$

Let $\mathbf{M} = \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^T$ and $\mathbf{N} = \mathbf{U}_2\mathbf{S}_2\mathbf{V}_2^T$ be the SVD of $\mathbf{M}$ and $\mathbf{N}$. The equality holds when $\mathbf{U}_1 = \mathbf{U}_2$ and $\mathbf{V}_1 = \mathbf{V}_2$.

Proof: We first introduce the important property in which the block circulant matrix can be block diagonalized [31], that is,

$$\hat{\mathbf{A}} = \mathtt{bdiag}\left(\hat{\mathcal{A}}\right) = \left(\mathbf{F}_{n_3} \otimes \mathbf{I}_{n_1}\right)\mathtt{bcirc}(\mathcal{A})\left(\mathbf{F}_{n_3}^{-1} \otimes \mathbf{I}_{n_2}\right) \tag{15}$$

where $\mathbf{F}_m$ is an $m \times m$ DFT matrix, $\mathbf{I}_n$ is an $n \times n$ identity matrix, $\otimes$ denotes the Kronecker product, and `bcirc` and `bdiag` are two block-based operators [43]. With this property, the t-product is equivalent to the matrix multiplication in the Fourier domain [43], i.e., $\mathcal{C} = \mathcal{A}*\mathcal{B}$ is equivalent to $\hat{\mathbf{C}} = \hat{\mathbf{A}}\hat{\mathbf{B}}$. This is important for efficient calculation and the following proof.

By (15), we convert the t-product in (14) to the matrix multiplication of the frontal slices in the Fourier domain

$$\begin{aligned}
&\min_{\hat{\mathbf{A}}_{(i)}^T\hat{\mathbf{A}}_{(i)}=\mathbf{I}} \|\hat{\mathbf{Y}}_{(i)} - \hat{\mathbf{A}}_{(i)}\hat{\mathbf{Z}}_{(i)}\|_F^2 + \rho\|\hat{\mathbf{A}}_{(i)} - \hat{\mathbf{B}}_{(i)}\|_F^2 \\
&= \min_{\hat{\mathbf{A}}_{(i)}^T\hat{\mathbf{A}}_{(i)}=\mathbf{I}} \mathrm{Trace}\left[\left(\hat{\mathbf{Y}}_{(i)} - \hat{\mathbf{A}}_{(i)}\hat{\mathbf{Z}}_{(i)}\right)^T\left(\hat{\mathbf{Y}}_{(i)} - \hat{\mathbf{A}}_{(i)}\hat{\mathbf{Z}}_{(i)}\right)\right] \\
&\quad + \rho\mathrm{Trace}\left[\left(\hat{\mathbf{A}}_{(i)} - \hat{\mathbf{B}}_{(i)}\right)^T\left(\hat{\mathbf{A}}_{(i)} - \hat{\mathbf{B}}_{(i)}\right)\right] \\
&= \min_{\hat{\mathbf{A}}_{(i)}^T\hat{\mathbf{A}}_{(i)}=\mathbf{I}} -\mathrm{Trace}\left[\hat{\mathbf{A}}_{(i)}\left(\hat{\mathbf{Z}}_{(i)}\hat{\mathbf{Y}}_{(i)}^T + \rho\hat{\mathbf{B}}_{(i)}^T\right)\right].
\end{aligned} \tag{16}$$

Let $\hat{\mathbf{C}}_{(i)} = \hat{\mathbf{Z}}_{(i)}\hat{\mathbf{Y}}_{(i)}^T + \rho\hat{\mathbf{B}}_{(i)}^T$, we can rewrrite (16) as

$$\max_{\hat{\mathbf{A}}_{(i)}} \mathrm{Trace}\left(\hat{\mathbf{A}}_{(i)}\hat{\mathbf{C}}_{(i)}\right), \quad \text{s.t. } \hat{\mathbf{A}}_{(i)}^T\hat{\mathbf{A}}_{(i)} = \mathbf{I}. \tag{17}$$

Since $\hat{\mathbf{A}}_{(i)}^T$ satisfies $\hat{\mathbf{A}}_{(i)}^T\hat{\mathbf{A}}_{(i)} = \mathbf{I}$, all singular values are 1. Let $\hat{\mathbf{C}}_{(i)} = \hat{\mathbf{U}}_{(i)}\hat{\mathbf{S}}_{(i)}\hat{\mathbf{V}}_{(i)}^T$ be the SVD of $\hat{\mathbf{C}}_{(i)}$. From Lemma 2, we can deduce that $\mathrm{Trace}(\hat{\mathbf{A}}_{(i)}\hat{\mathbf{C}}_{(i)})$ attains its upper bound when $\hat{\mathbf{A}}_{(i)}^T = \hat{\mathbf{U}}_{(i)}\hat{\mathbf{I}}_{(i)}\hat{\mathbf{V}}_{(i)}^T$, i.e., $\hat{\mathbf{A}}_{(i)} = \hat{\mathbf{V}}_{(i)}\hat{\mathbf{U}}_{(i)}^T$. Then, we can obtain $\hat{\mathbf{A}} = \hat{\mathbf{V}}\hat{\mathbf{U}}^T$, where $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ are from the SVD of $\hat{\mathbf{C}} = \hat{\mathbf{U}}\hat{\mathbf{S}}\hat{\mathbf{V}}^T$. By (15), $\hat{\mathbf{A}} = \hat{\mathbf{V}}\hat{\mathbf{U}}^T$ is equivalent to $\mathcal{A} = \mathcal{V}*\mathcal{U}^T$, where $\mathcal{U}$ and $\mathcal{V}$ are from the t-SVD of $\mathcal{C} = \mathcal{U}*\mathcal{S}*\mathcal{V}^T$. Therefore, the closed-form solution of (14) is $\mathcal{A}^* = \mathcal{V}*\mathcal{U}^T$, where $\mathcal{U}$ and $\mathcal{V}$ are from the t-SVD of $\mathcal{Z}*\mathcal{Y}^T + \rho\mathcal{B}^T = \mathcal{U}*\mathcal{S}*\mathcal{V}^T$.

By using Theorem 2, $\mathcal{A}$ subproblem (13) has the following closed-form solution:

$$\mathcal{A}^{k+1} = \mathcal{V}*\mathcal{U}^T \tag{18}$$

where $\mathcal{U}$ and $\mathcal{V}$ are from the t-SVD of $\mathcal{Z}^{k+1}*\vec{\mathcal{Y}}^T + \rho(\mathcal{A}^k)^T = \mathcal{U}*\mathcal{S}*\mathcal{V}^T$.

The "initialization+refinement" strategy has been widely utilized in the optimization of nonconvex problems [49], [50]. Since our model is nonconvex, we use this strategy to accelerate the denoising process and promote the denoising performance.

Initialization: We first learn the initial basis tensor $\mathcal{A}^0$ from preprocessed data $\mathcal{Y}_{\text{pre}}$ and obtain $\mathcal{A}^0 = \mathcal{U}_{\text{pre}}(:, 1:r, :)$, where $\mathcal{U}_{\text{pre}}$ is from the t-SVD of $\vec{\mathcal{Y}}_{\text{pre}} = \mathcal{U}_{\text{pre}}*\mathcal{S}_{\text{pre}}*\mathcal{V}_{\text{pre}}^T$ and r is the subspace dimension. In our method, we select FastHyDe [34] to initialize the basis tensor because of its high efficiency. Note that we only learn $\mathcal{A}^0$ from the preprocessed data, and initial coefficient tensor $\mathcal{Z}^0$ is the projection of the original noisy HSI on the subspace. This is different from that we remove noise directly on the preprocessed result.

Refinement: We use iterative regularization [12], [35], [39] to refine the basis tensor, and then, observation image $\mathcal{Y}^{t+1}$ in each outer iteration is updated as

$$\mathcal{Y}^{t+1} = \theta\mathcal{X}^t + (1-\theta)\mathcal{Y}$$

where θ $(0 \le \theta \le 1)$ is the iterative regularization parameter, $\mathcal{X}^t$ is the denoised image, and $\mathcal{Y}$ is the noisy image. In addition, the smaller r^0 can help to keep noise out. With the iteration, the observation image introduces more original information by the iterative regularization. We update r as

$$r^{t+1} = r^t + \delta$$

where δ is a positive integer. Thus, the basis tensor can capture more intrinsic information through iteration.

Summarizing the aforementioned descriptions, the PAM algorithm for the TenSR-based HSI denoising (TenSRDe) is presented in Algorithm 2.

Algorithm 2 PAM Algorithm for TenSRDe

Input: Noisy HSI $\mathcal{Y}$, regularization parameter λ, proximal parameter ρ, iterative regularization parameter θ, and increment δ.

Initialize: $\mathcal{A}^0$, $\mathcal{Z}^0 = (\mathcal{A}^0)^T * \vec{\mathcal{Y}}$, $\mathcal{X}^0 = \mathcal{O}$, and $\epsilon = 10^{-3}$.
for $t = 0 : T-1$ **do**
  Compute $r^{t+1} = r^t + \delta$;
  Compute $\mathcal{Y}^{t+1} = \theta\mathcal{X}^t + (1-\theta)\mathcal{Y}$.
  while not converged **do**
    Update $\mathcal{Z}^{k+1}$ by (3);
    Update $\mathcal{A}^{k+1}$ by (18);
    Check the convergence condition:

$$\|\mathcal{A}^{k+1} * \mathcal{Z}^{k+1} - \mathcal{A}^k * \mathcal{Z}^k\|_F^2 / \|\mathcal{A}^k * \mathcal{Z}^k\|_F^2 \le \epsilon.$$

  end while
  Let $\mathcal{X}^{t+1} = \text{inv-Permute}(\mathcal{A}^{k+1} * \mathcal{Z}^{k+1})$.
end for

Output: Estimated HSI $\mathcal{X}$.

C. Complexity Analysis

For Algorithm 2 with an input HSI $\mathcal{Y} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the main cost lies in the update of $\mathcal{Z}^{k+1}$ and $\mathcal{A}^{k+1}$. Updating $\mathcal{Z}^{k+1}$ requires P iterations, and each iteration needs to compute N SVDs of the $rp^2 \times q$ matrix, where P is the number of iterations in Algorithm 1, N is the number of similar 3-D block groups, p is the 3-D block size, and q is the 3-D block number. Updating $\mathcal{A}^{k+1}$ requires computing a t-SVD of the $n_1 \times n_3 \times n_2$ tensor. Let I denote the number of inner iterations, and the overall computational complexity of Algorithm 2 is $O(I(PNM_1 + n_2M_2))$, where $M_1 = \min\{q^2(p^2r), (p^2r)^2q\}$ and $M_2 = \min\{n_1^2n_3, n_3^2n_1\}$.

D. Convergence Analysis

We establish the global convergence of Algorithm 2. Let $\mathbf{\Omega} = \{\mathcal{A} | \mathcal{A}^T * \mathcal{A} = \mathcal{I}\}$, and then, the objective function (1) can be rewritten as

$$F(\mathcal{A}, \mathcal{Z}) = M(\mathcal{A}, \mathcal{Z}) + N(\mathcal{Z}) + I_\Omega(\mathcal{A}) \tag{19}$$

where $M(\mathcal{A}, \mathcal{Z}) = \frac{1}{2}\|\vec{\mathcal{Y}} - \mathcal{A} * \mathcal{Z}\|_F^2$, $N(\mathcal{Z}) = \lambda \sum_i \|\mathscr{R}_i \mathcal{Z}\|_{\mathbf{w},*}$, and

$$I_\Omega(\mathcal{A}) := \begin{cases} 0, & \text{if } \mathcal{A} \in \mathbf{\Omega} \\ +\infty, & \text{otherwise.} \end{cases}$$

Each variable is updated via

$$\begin{cases} \mathcal{Z}^{k+1} &= \arg\min_{\mathcal{Z}} \{ P_1(\mathcal{Z} \mid \mathcal{Z}^k) \\ &:= M(\mathcal{A}^k, \mathcal{Z}) + N(\mathcal{Z}) + \frac{\rho}{2}\|\mathcal{Z} - \mathcal{Z}^k\|_F^2 \} \\ \mathcal{A}^{k+1} &= \arg\min_{\mathcal{A}} \{ P_2(\mathcal{A} \mid \mathcal{A}^k) \\ &:= M(\mathcal{A}, \mathcal{Z}^{k+1}) + I_\Omega(\mathcal{A}) + \frac{\rho}{2}\|\mathcal{A} - \mathcal{A}^k\|_F^2 \}. \end{cases} \tag{20}$$

We first introduce Lemmas 3 and 4 that are the main ingredients for proving Theorem 3.

Lemma 3 (Descent Lemma): Let $\{\mathcal{A}^k, \mathcal{Z}^k\}$ be the sequence generated by (20). For F defined in (19), there exists $\rho > 0$ such that

$$F(\mathcal{A}^{k+1}, \mathcal{Z}^{k+1}) + \frac{\rho}{2}(\|\mathcal{Z}^{k+1} - \mathcal{Z}^k\|_F^2 + \|\mathcal{A}^{k+1} - \mathcal{A}^k\|_F^2) \le F(\mathcal{A}^k, \mathcal{Z}^k).$$

Proof: Since $\mathcal{A}^k$ is the optimal solution of F, $I_\Omega(\mathcal{A}^k) = 0$. From (19) and (20)

$$P_1(\mathcal{Z}^k \mid \mathcal{Z}^k) = F(\mathcal{A}^k, \mathcal{Z}^k)$$
$$P_1(\mathcal{Z}^{k+1} \mid \mathcal{Z}^k) = F(\mathcal{A}^k, \mathcal{Z}^{k+1}) + \frac{\rho}{2}\|\mathcal{Z}^{k+1} - \mathcal{Z}^k\|_F^2.$$

Since $\mathcal{Z}^{k+1}$ is the optimal solution of $P_1(\mathcal{Z} \mid \mathcal{Z}^k)$

$$P_1(\mathcal{Z}^{k+1} \mid \mathcal{Z}^k) \le P_1(\mathcal{Z}^k \mid \mathcal{Z}^k).$$

Then

$$F(\mathcal{A}^k, \mathcal{Z}^{k+1}) + \frac{\rho}{2}\|\mathcal{Z}^{k+1} - \mathcal{Z}^k\|_F^2 \le F(\mathcal{A}^k, \mathcal{Z}^k). \tag{21}$$

In a similar way, we can deduce

$$F(\mathcal{A}^{k+1}, \mathcal{Z}^{k+1}) + \frac{\rho}{2}\|\mathcal{A}^{k+1} - \mathcal{A}^k\|_F^2 \le F(\mathcal{A}^k, \mathcal{Z}^{k+1}). \tag{22}$$

Combining (21) and (22)

$$F(\mathcal{A}^{k+1}, \mathcal{Z}^{k+1}) + \frac{\rho}{2}(\|\mathcal{Z}^{k+1} - \mathcal{Z}^k\|_F^2 + \|\mathcal{A}^{k+1} - \mathcal{A}^k\|_F^2) \le F(\mathcal{A}^k, \mathcal{Z}^k).$$

The proof is completed.

Lemma 4 (Relative Error Lemma): Let $\{\mathcal{A}^k, \mathcal{Z}^k\}$ be a sequence generated by (20). For each $k \in \mathbb{N}$ and F defined in (19), there exists $Q^{k+1} \in \partial F(\mathcal{A}^{k+1}, \mathcal{Z}^{k+1})$ and $\beta > 0$; then

$$\|Q^{k+1}\|_F \le \beta(\|\mathcal{Z}^{k+1} - \mathcal{Z}^k\|_F + \|\mathcal{A}^{k+1} - \mathcal{A}^k\|_F).$$

Proof:

Step 1: By the first-order optimal condition of (20)

$$0 \in \partial N(\mathcal{Z}^{k+1}) + \nabla_{\mathcal{Z}} M(\mathcal{A}^k, \mathcal{Z}^{k+1}) + \rho(\mathcal{Z}^{k+1} - \mathcal{Z}^k).$$

According to [53, Th. 2.5], the subgradient of $N(\mathcal{Z})$ is

$$\sum_i \mathbf{U}_i \text{Diag}(\mathbf{w}_i) \mathbf{V}_i^T \in \partial N(\mathcal{Z}) \tag{23}$$

where $\mathbf{U}_i$ and $\mathbf{V}_i$ are the SVD matrices of $\mathscr{R}_i\mathcal{Z}$. Let $\mathbf{U}_i^{k+1}$ and $\mathbf{V}_i^{k+1}$ be the SVD matrices of $\mathscr{R}_i\mathcal{Z}^{k+1}$ and then define

$$V_1^{k+1} := \sum_i \mathbf{U}_i^{k+1} \text{Diag}(\mathbf{w}_i^{k+1}) (\mathbf{V}_i^{k+1})^T \in \partial N(\mathcal{Z}^{k+1}).$$

Consequently, there exists V_1^{k+1} such that

$$0 = V_1^{k+1} + \nabla_{\mathcal{Z}} M\left(\mathcal{A}^k, \mathcal{Z}^{k+1}\right) + \rho\left(\mathcal{Z}^{k+1} - \mathcal{Z}^k\right).$$

It is clear that

$$\|V_1^{k+1} + \nabla_{\mathcal{Z}} M\left(\mathcal{A}^k, \mathcal{Z}^{k+1}\right)\|_F = \rho\|\mathcal{Z}^{k+1} - \mathcal{Z}^k\|_F. \quad (24)$$

Similarly, there exists $V_2^{k+1} \in \partial I_\Omega(\mathcal{A}^{k+1})$, and we can obtain

$$\|V_2^{k+1} + \nabla_{\mathcal{A}} M\left(\mathcal{A}^{k+1}, \mathcal{Z}^k\right)\|_F = \rho\|\mathcal{A}^{k+1} - \mathcal{A}^k\|_F. \quad (25)$$

Step 2: Define the following formula:

$$Q^{k+1} := \left\{Q_1^{k+1}, Q_2^{k+1}\right\} \in \partial F\left(\mathcal{A}^{k+1}, \mathcal{Z}^{k+1}\right)$$

where

$$\begin{cases} Q_1^{k+1} = V_1^{k+1} + \nabla_{\mathcal{Z}} M\left(\mathcal{A}^{k+1}, \mathcal{Z}^{k+1}\right) \\ Q_2^{k+1} = V_2^{k+1} + \nabla_{\mathcal{A}} M\left(\mathcal{A}^{k+1}, \mathcal{Z}^{k+1}\right). \end{cases}$$

By the triangle inequality

$$\begin{aligned}\|Q_1^{k+1}\|_F \le{}& \|V_1^{k+1} + \nabla_{\mathcal{Z}} M\left(\mathcal{A}^k, \mathcal{Z}^{k+1}\right)\|_F \\ &+ \|\nabla_{\mathcal{Z}} M\left(\mathcal{A}^{k+1}, \mathcal{Z}^{k+1}\right) - \nabla_{\mathcal{Z}} M\left(\mathcal{A}^k, \mathcal{Z}^{k+1}\right)\|_F.\end{aligned}$$

According to (24) and the Lipschitz continuity (see Definition 8) of ∇M (with Lipschitz constant l)

$$\left\|Q_1^{k+1}\right\|_F \le (\rho + l)\|\mathcal{Z}^{k+1} - \mathcal{Z}^k\|_F. \quad (26)$$

Similarly, we can obtain

$$\left\|Q_2^{k+1}\right\|_F \le (\rho + l)\|\mathcal{A}^{k+1} - \mathcal{A}^k\|_F. \quad (27)$$

Finally, from (26) and (27), for some $\beta > 0$

$$\|Q^{k+1}\|_F \le \beta\left(\|\mathcal{Z}^{k+1} - \mathcal{Z}^k\|_F + \|\mathcal{A}^{k+1} - \mathcal{A}^k\|_F\right).$$

The proof is completed.

Theorem 3 (Global Convergence): Let $\{\mathcal{A}^k, \mathcal{Z}^k\}$ be a sequence generated by Algorithm 2, and it globally converges to a critical point $(\mathcal{A}^*, \mathcal{Z}^*)$ of F defined in (19).

Proof: According to [54, Th. 6.2], to prove Theorem 3, we need to verify the following two conditions: 1) F defined in (19) is a KL function and 2) $\{\mathcal{A}^k, \mathcal{Z}^k\}$ generated by Algorithm 2 is bounded.

For condition 1, since the Frobenius norm and weighted nuclear norm are semialgebraic [44], [53], M and N defined in (19) are semialgebraic functions. Because the orthogonal set in matrix theory is a Stiefel manifold and the Stiefel manifold is a semialgebraic set [44], $\Omega = \{\mathcal{A}|\mathcal{A}^T * \mathcal{A} = \mathcal{I}\}$ is a semialgebraic set by (15). Then, I_Ω defined in (19) is a semialgebraic function because the indicator function of a semialgebraic set is a semialgebraic function [44]. Therefore, F is a semialgebraic function since the finite sum of semialgebraic functions is a semialgebraic function [44]. By [46, Th. 3] and Definition 6, F is a KL function.

For condition (2), first, the indicator function I_Ω should be 0 from its definition in (19) since $\mathcal{A}^k \in \mathbb{R}^{n_1 \times r \times n_2}$ is the optimal solution of F. Thus, by [31, Lemma 3.19]

$$\|\mathcal{A}^k\|_F^2 = \text{Trace}\left(\left(\left(\mathcal{A}^k\right)^T * \mathcal{A}^k\right)_{(:,:,1)}\right) = \text{Trace}(\mathcal{I}(:,:,1)) = r \quad (28)$$

where $\mathcal{A}_{(:,:,1)}$ is the first frontal slice of $\mathcal{A}$. Therefore, $\{\mathcal{A}^k\}$ is bounded. Second, by using (15) and the triangle inequality, we have

$$\begin{aligned}\|\mathcal{Z}^k\|_F^2 - \left\|\left(\mathcal{A}^{k-1}\right)^T\right\|_F^2 \left\|\vec{\mathcal{Y}}\right\|_F^2 &\le \|\mathcal{Z}^k\|_F^2 - \left\|\left(\mathcal{A}^{k-1}\right)^T * \vec{\mathcal{Y}}\right\|_F^2 \\ &\le \left\|\mathcal{Z}^k - \left(\mathcal{A}^{k-1}\right)^T * \vec{\mathcal{Y}}\right\|_F^2\end{aligned}$$

which is equivalent to

$$\|\mathcal{Z}^k\|_F^2 \le \left\|\mathcal{Z}^k - \left(\mathcal{A}^{k-1}\right)^T * \vec{\mathcal{Y}}\right\|_F^2 + \left\|\left(\mathcal{A}^{k-1}\right)^T\right\|_F^2 \left\|\vec{\mathcal{Y}}\right\|_F^2.$$

From Lemma 3, the value of F is monotonically decreasing, the terms $\|\vec{\mathcal{Y}} - \mathcal{A} * \mathcal{Z}\|_F^2$ and $\sum_i \|\mathscr{R}_i \mathcal{Z}\|_{\mathbf{w},*}$ in F are nonnegative, and $\|\vec{\mathcal{Y}} - \mathcal{A} * \mathcal{Z}\|_F^2$ is bounded. By using Lemma 1, $\|\mathcal{Z}^k - (\mathcal{A}^{k-1})^T * \vec{\mathcal{Y}}\|_F^2$ is bounded. In addition, $\|(\mathcal{A}^{k-1})^T\|_F^2$ and $\|\vec{\mathcal{Y}}\|_F^2$ are bounded. Thus, $\{\mathcal{Z}^k\}$ is bounded. In summary, $\{\mathcal{A}^k, \mathcal{Z}^k\}$ generated by Algorithm 2 is bounded.

Lemmas 3 and 4 show that the format of Algorithm 2 is consistent with [54, Algorithm 4]. Combining the above conditions, by [54, Th. 6.2], we obtain that $\{\mathcal{A}^k, \mathcal{Z}^k\}$ converges to a critical point $(\mathcal{A}^*, \mathcal{Z}^*)$ of F.

The proof is completed.

Remark 2: "Global convergence" means that the whole sequence converges to a critical point, which enhances the conventional convergence results "every limit point of the sequence is a critical point." We follow [52] in using "global convergence."

IV. Experimental Results and Discussion

We perform simulated and real experiments to demonstrate the effectiveness of our method for HSI denoising and compare it with eight state-of-the-art methods: the direct approaches BM3D [10], BM4D [54], KBR [39], TDL [55], and LRTFL_0 [56] and the subspace representation approaches FastHyDe [34], SNLRSF [36], and NGmeet [35]. The codes of the comparison methods are downloaded from the authors' homepages, and all parameters in the codes follow the source code settings or are manually tuned to the best results according to the reference recommendations. The parameter setting of our method for all experiments is presented in Table VII. The gray values of the testing HSIs are scaled to the interval [0, 1] band by band, and they are returned to the original level after denoising. All experiments are performed under Windows 10 and MATLAB R2016a running on a desktop with a Core i5 3.30-GHz CPU and 16-GB memory.

A. Simulated Data Experiments

Benchmark Data Set: We select three public data sets to simulate noisy HSIs.

1) *Pavia Centre (PaC[1]):* This data set is collected by the reflective optics system imaging spectrometer (ROSIS-03), and the whole data set contains 1400×512 pixels and 102 spectral bands, including 80 noise-free bands. The subimage of size $200 \times 200 \times 80$ is used in our experiments.

[1] https://rslab.ut.ac.ir/data

2) *Washington DC Mall (WDC[1]):* This data set is collected by the hyperspectral digital imagery collection experiment (HYDICE), and the whole image contains 1208×307 pixels and 210 spectral bands, including 191 noise-free bands. The subimage of size $256 \times 256 \times 191$ is used in our experiments.
3) *Cuprite[1]:* This data set is collected by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS), and the whole data set contains 350×350 pixels and 224 spectral bands, including 188 noise-free bands. The subimage of size $350 \times 350 \times 188$ is used in our experiments.

Evaluation Index: To objectively measure the performance of all comparison methods, four quantitative picture quality indexes (PQIs) are employed for quantitative evaluation, including the peak signal-to-noise ratio (PSNR), structure similarity (SSIM) [57], feature similarity (FSIM) [58], and erreur relative global adimensionnelle de synthèse (ERGAS) [59]. PSNR and SSIM measure the similarity between the clean image and the denoised image based on MSE and structural consistency, respectively. FSIM explains the perceptual consistency with the clean image. The larger these three PQIs are, the better the denoising results are. ERGAS evaluates the spectral quality of all fusion bands within the spectral range and considers the overall situation of spectral changes. In contrast to the previous PQIs, a lower ERGAS corresponds to a better result.

Experimental Setting: To test the denoising performance of the proposed TenSRDe method under different noise levels, we add zero-mean additive Gaussian noise with different variances σ^2 to three benchmark data sets and obtain the simulated noisy HSIs with $\sigma = \{0.02, 0.04, 0.06, 0.08, 0.10\}$. The quantitative and qualitative comparisons of the denoising results of all competing methods on three data sets are reported as follows.

1) Results on PaC Data Set: For a quantitative comparison, Table I presents the mean PSNR (MPSNR), mean SSIM (MSSIM), mean FSIM (MFSIM), ERGAS, and running time of all comparison methods for PaC data set denoising. The highest values of MPSNR, MSSIM, and MFSIM and the lowest value of ERGAS and running time are highlighted in bold. It can be observed that the proposed TenSRDe method significantly outperforms the other competing methods with respect to all evaluation indexes. Especially, TenSRDe achieves an approximately 0.6–1.1-dB improvement in PSNR over the second-best method NGmeet. This can be easily explained by the stronger subspace representation of the tensor subspace, which can preserve the more intrinsic tensor structure and then naturally lead to the better denoising performance of TenSRDe. Fig. 6 shows the PSNR and SSIM across all brands of the PaC data set under five noise levels. It is easy to observe that the proposed TenSRDe method (red curve) obtains the best PSNR and SSIM values across almost all the bands. This demonstrates the effectiveness of TenSRDe for HSI denoising.

For a visual quality comparison, Fig. 9 shows the pseudocolor images of the denoising results under noise level $\sigma = 0.10$. We enlarged the same area of each subfigure with a blue box for a better comparison. As shown in Fig. 9, BM3D removes the noise but loses the detail of the original image because of oversmoothing. All the remaining methods remove the Gaussian noise effectively. Due to the outstanding performance of the comparison methods, it is hard to find the evident difference from the visual displays of the denoising results. Thus, we employ the residual image (the difference between each denoised band and the original band) to show the superiority of the denoising methods. Fig. 10 gives the typical band 32 of the denoising result and its residual image under noise level $\sigma = 0.08$. By comparing the residual images, the proposed TenSRDe achieves the smallest residual, which verifies the improvement of the representation capability of the tensor subspace.

2) Washington DC Mall: Table I gives the MPSNR, MSSIM, MFSIM, ERGAS, and running time of all comparison methods for the WDC data set denoising. The highest values of MPSNR, MSSIM, and MFSIM and the lowest value of ERGAS and running time are highlighted in bold. As observed, the proposed TenSRDe method yields comparable results in terms of all measure indexes. Compared with the matrix subspace methods, TenSRDe achieves an approximately 1–2-dB gain in PSNR. This validates that the tensor subspace has better representation capability than the matrix subspace. Fig. 7 shows the PSNR and SSIM across all bands of the WDC data set under five noise levels. Compared with all competing methods, the proposed TenSRDe method (red curve) achieves higher PSNR and SSIM values across all bands, which verifies the robustness of TenSRDe for HSI denoising.

Fig. 11 shows the pseudocolor images of the denoising results under noise level $\sigma = 0.10$. We enlarged the same area of each subfigure with a green box for a clearer comparison. As shown in Fig. 11(c), BM3D still smooths some details of the original image, and the other methods achieve satisfactory denoising effects. Fig. 12 gives the typical band 163 and the residual image of denoising results under noise level $\sigma = 0.06$. The proposed TenSRDe attains the highest PSNR and SSIM in band 163, and the residual images indicate that TenSRDe gains the best denoising performance.

3) Cuprite: Table I lists the MPSNR, MSSIM, MFSIM, ERGAS, and running time of all competing methods for the Cuprite data set denoising. The highest values of MPSNR, MSSIM, MFSIM, and the lowest value of ERGAS and running time are highlighted in bold. As observed, the proposed TenSRDe method consistently achieves the best performance on all measure indexes, in which the proposed TenSRDe exceeds approximately 1–1.7 dB in PSNR compared with the matrix subspace methods. This also demonstrates that the tensor subspace possesses better representation capability than the matrix subspace, i.e., it can avoid the loss of the intrinsic tensor structure information, which results in a better denoising performance than the matrix subspace methods. Fig. 8 presents the PSNR and SSIM across all bands of the Cuprite data set under five noise levels. Except for the PSNR and SSIM of KBR being higher than those of the TenSRDe in the last few bands when σ is 0.02, the proposed TenSRDe method (red curve) attains higher PSNR and SSIM values across almost every band under the noise level $\sigma = \{0.04, 0.06, 0.08, 0.10\}$. Overall, TenSRDe achieves the best performance over all the competing methods for both PSNR and SSIM.

TABLE I

QUANTITATIVE EVALUATION AND RUNNING TIME (IN MINUTES) OF DIFFERENT METHODS ON THREE DATA SETS WITH SIMULATED NOISE

Dataset	σ	Index	Noisy	BM3D	BM4D	KBR	TDL	LRTF$L0$	FastHyDe	SNLRSF	NGmeet	TenSRDe
PaC	0.02	MPSNR	33.983	36.744	43.862	45.994	46.367	45.010	47.078	47.761	48.251	**49.247**
		MSSIM	0.9326	0.9670	0.9930	0.9954	0.9959	0.9948	0.9966	0.9971	0.9973	**0.9978**
		MFSIM	0.9628	0.9779	0.9956	0.9973	0.9975	0.9965	0.9979	0.9983	0.9984	**0.9987**
		ERGAS	74.069	53.384	23.703	18.909	18.211	21.220	16.753	15.589	14.677	**13.469**
		Time	—	0.5245	1.6773	7.6816	0.1454	15.127	**0.0873**	6.3234	0.5243	0.3777
	0.04	MPSNR	27.961	32.774	39.312	41.100	41.862	41.029	42.566	43.273	43.917	**44.642**
		MSSIM	0.7937	0.9226	0.9808	0.9865	0.9889	0.9873	0.9908	0.9922	0.9932	**0.9940**
		MFSIM	0.8955	0.9492	0.9876	0.9918	0.9933	0.9920	0.9941	0.9953	0.9958	**0.9961**
		ERGAS	148.15	84.246	39.907	32.869	30.363	33.321	27.935	25.729	23.904	**22.649**
		Time	—	0.5476	1.6747	6.8034	0.1225	9.2317	**0.0881**	6.2494	0.5143	0.3472
	0.06	MPSNR	24.434	30.628	36.795	38.262	39.042	38.638	40.036	40.657	41.359	**41.943**
		MSSIM	0.6526	0.8786	0.9664	0.9751	0.9801	0.9789	0.9841	0.9863	0.9884	**0.9890**
		MFSIM	0.8298	0.9215	0.9782	0.9849	0.9877	0.9865	0.9895	0.9916	0.9925	**0.9929**
		ERGAS	222.41	107.80	53.260	45.379	41.783	43.893	37.282	34.568	31.895	**30.829**
		Time	—	0.5785	1.6732	6.6956	0.1187	6.3294	**0.0826**	6.1783	0.5214	0.2995
	0.08	MPSNR	21.936	29.182	34.997	36.456	37.166	37.039	38.330	38.928	39.546	**40.022**
		MSSIM	0.5327	0.8371	0.9500	0.9627	0.9698	0.9692	0.9771	0.9800	0.9829	**0.9835**
		MFSIM	0.7719	0.8959	0.9673	0.9775	0.9811	0.9809	0.9846	0.9876	0.9887	**0.9891**
		ERGAS	296.46	127.28	65.438	55.729	51.656	52.722	45.370	42.058	39.125	**37.958**
		Time	—	0.6103	1.6442	6.6286	0.1234	6.2203	**0.0883**	6.2116	0.5176	0.2918
	0.10	MPSNR	19.998	28.100	33.655	35.500	35.642	35.702	37.046	37.483	38.210	**38.645**
		MSSIM	0.4362	0.7984	0.9334	0.9585	0.9586	0.9580	0.9701	0.9733	0.9775	**0.9783**
		MFSIM	0.7214	0.8725	0.9563	0.9744	0.9745	0.9746	0.9796	0.9831	0.9850	**0.9855**
		ERGAS	370.51	144.13	76.337	62.159	61.375	61.528	52.530	49.573	45.543	**44.932**
		Time	—	0.6334	1.6863	6.5056	0.1443	5.8963	**0.0877**	6.1883	0.5176	0.2777
WDC	0.02	MPSNR	33.977	36.315	43.604	47.556	47.592	46.742	49.687	49.041	49.787	**51.623**
		MSSIM	0.9366	0.9667	0.9934	0.9971	0.9973	0.9964	0.9981	0.9980	0.9982	**0.9988**
		MFSIM	0.9689	0.9798	0.9960	0.9982	0.9984	0.9976	0.9989	0.9987	0.9990	**0.9993**
		ERGAS	75.448	56.575	24.684	16.245	16.733	17.797	12.891	14.826	12.645	**10.368**
		Time	—	2.1015	7.6493	24.075	0.5627	127.33	**0.1565**	10.543	0.9347	1.5100
	0.04	MPSNR	27.956	32.026	38.894	42.782	42.719	42.312	44.797	45.122	44.979	**46.398**
		MSSIM	0.8074	0.9164	0.9809	0.9917	0.9918	0.9907	0.9945	0.9953	0.9952	**0.9962**
		MFSIM	0.9117	0.9515	0.9886	0.9945	0.9950	0.9939	0.9967	0.9970	0.9970	**0.9977**
		ERGAS	150.92	92.455	42.244	27.614	28.701	29.390	22.344	21.794	21.983	**18.671**
		Time	—	2.1884	7.5624	24.033	0.5056	86.384	**0.1583**	10.558	0.9474	1.3845
	0.06	MPSNR	24.435	29.813	36.222	39.526	39.583	39.819	42.002	42.622	42.441	**43.752**
		MSSIM	0.6773	0.8667	0.9651	0.9832	0.9845	0.9843	0.9901	0.9919	0.9916	**0.9933**
		MFSIM	0.8548	0.9240	0.9794	0.9883	0.9904	0.9899	0.9940	0.9948	0.9948	**0.9959**
		ERGAS	226.36	119.13	57.323	39.832	40.547	38.982	30.561	28.436	29.033	**25.252**
		Time	—	2.2965	7.5517	24.023	0.5095	58.577	**0.1603**	10.496	0.9424	1.2896
	0.08	MPSNR	21.939	28.367	34.349	37.138	37.405	38.027	40.084	40.784	40.549	**41.740**
		MSSIM	0.5656	0.8207	0.9469	0.9720	0.9760	0.9777	0.9853	0.9881	0.9873	**0.9897**
		MFSIM	0.8031	0.8988	0.9693	0.9808	0.9856	0.9860	0.9909	0.9924	0.9923	**0.9937**
		ERGAS	301.69	140.55	71.050	52.237	51.712	47.576	37.998	34.770	35.814	**31.764**
		Time	—	2.3885	7.5383	23.954	0.5137	34.382	**0.1614**	10.563	0.9413	1.2353
	0.10	MPSNR	20.001	27.311	32.935	35.949	35.913	36.764	38.702	39.203	39.113	**40.263**
		MSSIM	0.4741	0.7788	0.9279	0.9679	0.9671	0.9703	0.9803	0.9835	0.9829	**0.9859**
		MFSIM	0.7569	0.8755	0.9584	0.9805	0.9802	0.9815	0.9878	0.9894	0.9894	**0.9914**
		ERGAS	377.06	158.62	83.534	59.399	61.032	54.871	44.322	41.511	42.031	**37.456**
		Time	—	2.4635	7.5322	23.983	0.5167	36.033	**0.1617**	10.682	0.9404	1.2337
Cuprite	0.02	MPSNR	33.977	38.361	43.677	46.282	46.490	44.982	46.539	46.227	46.175	**47.745**
		MSSIM	0.8380	0.9355	0.9787	0.9871	0.9872	0.9823	0.9850	0.9839	0.9867	**0.9874**
		MFSIM	0.9442	0.9639	0.9903	0.9941	0.9948	0.9923	0.9931	0.9926	0.9925	**0.9942**
		ERGAS	62.717	37.637	21.698	17.942	17.688	21.260	22.400	25.369	22.986	**20.250**
		Time	—	4.8724	12.867	49.268	1.3943	306.74	**0.3117**	19.634	1.6673	2.2445
	0.04	MPSNR	27.960	34.977	39.688	42.273	42.751	41.731	44.292	44.074	43.719	**45.283**
		MSSIM	0.5840	0.8704	0.9490	0.9717	0.9743	0.9680	0.9785	0.9779	0.9827	**0.9838**
		MFSIM	0.8447	0.9166	0.9722	0.9852	0.9880	0.9843	0.9898	0.9894	0.9912	**0.9919**
		ERGAS	125.39	55.639	34.638	27.465	26.981	28.798	26.170	27.844	23.711	**22.033**
		Time	—	5.1665	12.845	49.296	1.2578	169.23	**0.3094**	19.523	1.7146	2.0883
	0.06	MPSNR	24.437	33.314	37.491	40.139	40.502	39.936	42.649	42.421	42.301	**43.465**
		MSSIM	0.4004	0.8238	0.9189	0.9555	0.9597	0.9525	0.9717	0.9710	0.9750	**0.9752**
		MFSIM	0.7503	0.8818	0.9512	0.9745	0.9801	0.9762	0.9861	0.9853	0.9871	**0.9877**
		ERGAS	188.09	67.472	44.140	35.158	33.721	34.278	29.405	30.610	28.085	**27.254**
		Time	—	5.2976	12.864	49.492	1.1816	176.37	**0.3103**	19.597	1.6972	2.0142
	0.08	MPSNR	21.937	32.254	36.056	38.905	38.653	38.610	41.223	41.109	41.082	**41.948**
		MSSIM	0.2833	0.7894	0.8920	0.9449	0.9429	0.9376	0.9646	0.9633	0.9655	**0.9685**
		MFSIM	0.6705	0.8573	0.9313	0.9666	0.9704	0.9679	0.9820	0.9806	0.9819	**0.9840**
		ERGAS	250.83	76.344	51.340	39.399	40.373	39.488	32.735	33.449	32.599	**31.501**
		Time	—	5.3306	13.135	49.293	1.3004	197.13	**0.3186**	19.512	1.7004	1.9527
	0.10	MPSNR	20.000	31.472	35.019	38.096	37.460	37.618	40.214	40.067	40.052	**40.831**
		MSSIM	0.2080	0.7625	0.8682	0.9375	0.9269	0.9235	0.9575	0.9555	0.9569	**0.9619**
		MFSIM	0.6043	0.8390	0.9132	0.9607	0.9606	0.9598	0.9778	0.9758	0.9768	**0.9803**
		ERGAS	313.50	83.567	57.307	42.601	45.497	44.712	35.564	36.230	35.958	**34.377**
		Time	—	5.3397	13.003	49.554	1.2132	205.77	**0.3183**	19.612	1.7167	1.8112

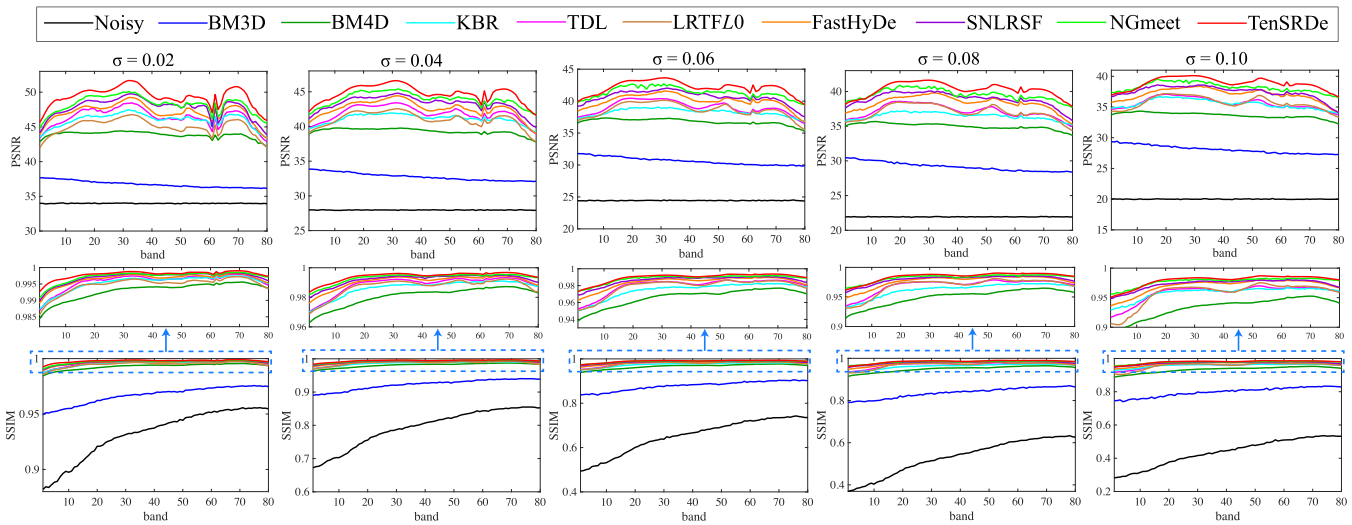


Fig. 6. PSNR and SSIM values of each band of different methods on the PaC data set. The first to fifth columns correspond to the Gaussian noise level σ from 0.02 to 0.10 with an increment of 0.02.

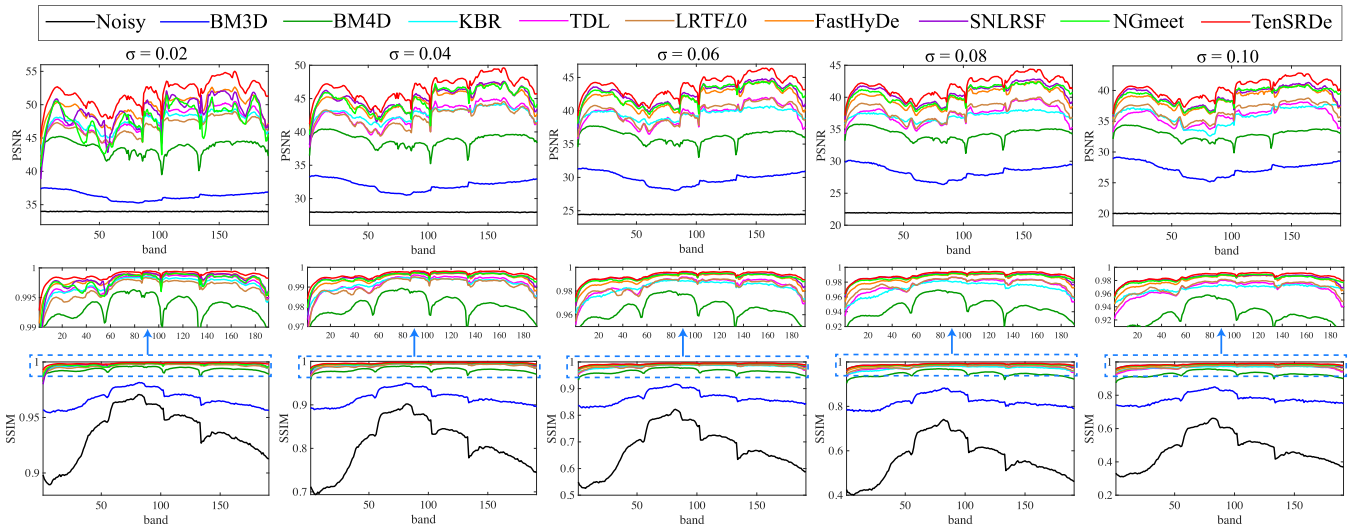


Fig. 7. PSNR and SSIM values of each band of different methods on the WDC data set. The first to fifth columns correspond to the Gaussian noise level σ from 0.02 to 0.10 with an increment of 0.02.

Fig. 13 shows the pseudocolor images of the denoising results under noise level $\sigma = 0.10$. We enlarged the same area of each subfigure with a red box for a more detailed comparison. All methods achieve outstanding noise removal performance except for BM3D. For a comprehensive comparison, Fig. 14 gives the typical band 126 and the residual image of denoising result under noise level $\sigma = 0.08$. It is obvious that TenSRDe attains the higher PSNR and SSIM and reaches the minimum residual in the band 126. This illustrates the superiority of TenSRDe.

Running Time: Table I presents the running time of all methods on the PaC, WDC, and Cuprite data sets. We have the following observations. First, the subspace representation approaches, FastHyDe, NGmeet, and TenSRDe, spend less time than the direct approach (except for TDL) since the former projects the original image into a low-dimensional subspace, leading to a decrease in computational complexity. Second, in all the subspace representation methods, the proposed TenSRDe runs for a shorter time than SNLRSF on average and runs slightly longer than FastHyDe and NGmeet. Considering the denoising performance, TenSRDe achieves the best performance/efficiency tradeoff.

B. Real Data Experiments

We use the Indian Pines[1] and Urban[1] data sets to evaluate the performance of the proposed TenSRDe on real data sets. To guarantee a possibly good performance, all involved parameters in the competing methods are finely adjusted or set according to the original references. Since the noise level σ in our method needs to be known, we estimate the initial noise variances of the Indian Pines and Urban data sets by a multiple regression theory-based approach [60]. Then, all parameters of the proposed method are determined according to Table VII.

1) Indian Pines Data Set: The Indian Pines data set is collected by AVIRIS and contains 145×145 pixels and

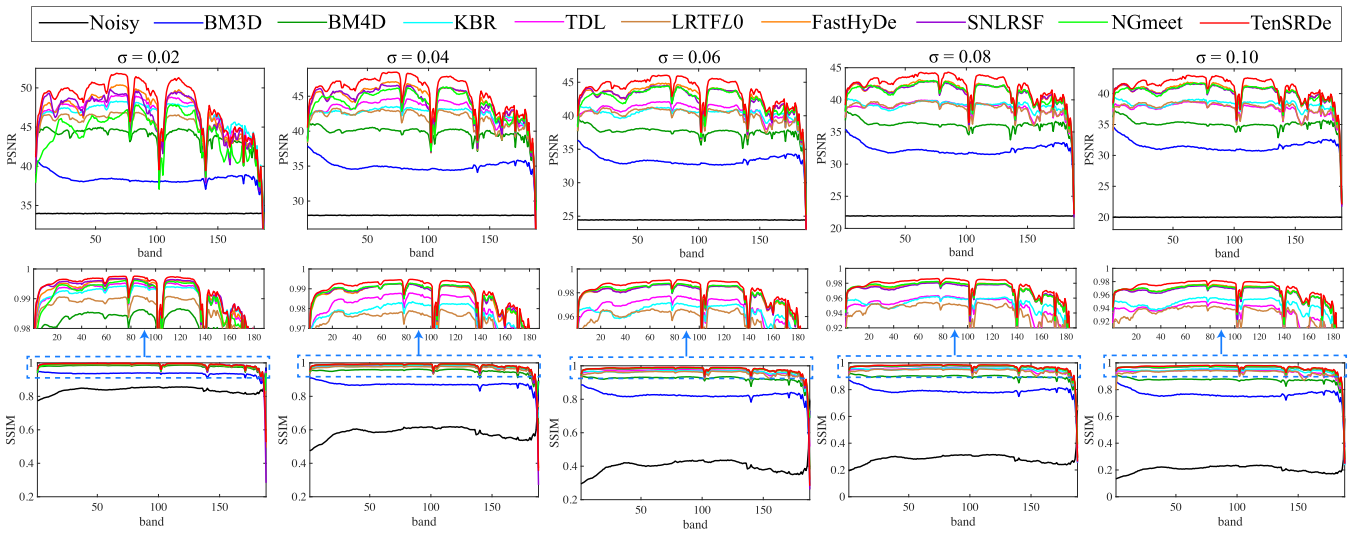


Fig. 8. PSNR and SSIM values of each band of different methods on the Cuprite data set. The first to fifth columns correspond to the Gaussian noise level σ from 0.02 to 0.10 with an increment of 0.02.

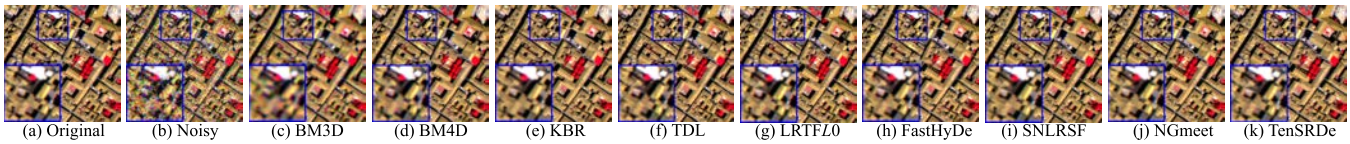


Fig. 9. (a) Original pseudocolor image (R: 70, G: 40, B: 10) of the PaC data set. (b) Noisy image under the Gaussian noise level $\sigma = 0.10$; Denoised images of (c) BM3D, (d) BM4D, (e) KBR, (f) TDL, (g) LRTFL0, (h) FastHyDe, (i) SNLRSF, (j) NGmeet, and (k) TenSRDe. To facilitate the observation of details, the demarcated area in each image is enlarged.

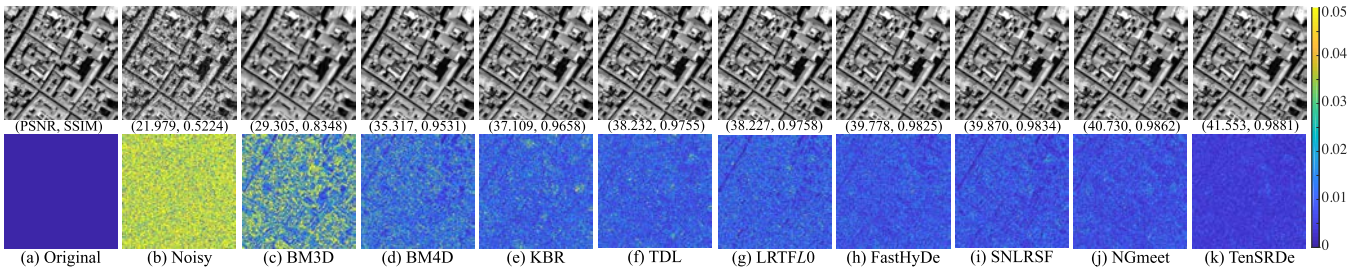


Fig. 10. (a) Original image and residual image of the PaC data set in band 32. (b) Noisy image and residual image under the Gaussian noise level $\sigma = 0.08$. Denoised images and residual images (the difference between each denoised band with the original band) of (c) BM3D, (d) BM4D, (e) KBR, (f) TDL, (g) LRTFL0, (h) FastHyDe, (i) SNLRSF, (j) NGmeet, and (k) TenSRDe.

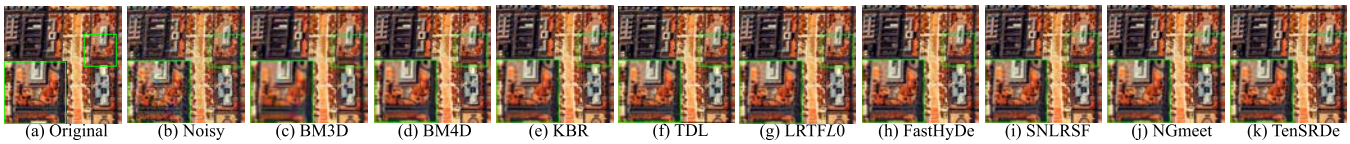


Fig. 11. (a) Original pseudocolor image (R: 70, G: 120, B: 170) of the WDC data set. (b) Noisy image under the Gaussian noise level $\sigma = 0.10$. Denoised results of (c) BM3D, (d) BM4D, (e) KBR, (f) TDL, (g) LRTFL0, (h) FastHyDe, (i) SNLRSF, (j) NGmeet, and (k) TenSRDe. To facilitate the observation of details, the demarcated area in each image is enlarged.

224 spectral bands. We use a subimage of size $145 \times 145 \times 200$ in our experiment. Fig. 15 shows the pseudocolor image of the original and denoised Indian Pines data set. From Fig. 15, we can observe that BM3D, KBR, and TDL can remove some noise and BM4D oversmooths the image. LRTF$L0$, FastHyDe, SNLRSF, NGmeet, and TenSRDe achieve a satisfactory result. To further compare the details, we enlarged the same area of each subfigure with a red box. The denoising results of BM3D, BM4D, KBR, and TDL exhibit edge distortion or blur. LRTF$L0$, FastHyDe, SNLRSF, NGmeet, and TenSRDe preserve the sharp and clear edge details.

2) Urban Data Set: The Urban data set is collected by HYDICE and contains 307×307 pixels and 210 spectral bands. We use the whole image in our experiment. Fig. 16 shows the pseudocolor image of the denoised Urban data set. From Fig. 16, we can observe that the original image is contaminated by Gaussian noise and stripes. The results of BM3D and TDL maintain the blue stripes and BM4D oversmooths the image. LRTF$L0$ removes the noise but causes color distortion, and KBR, FastHyDe, SNLRSF, NGmeet, and TenSRDe remove both the Gaussian noise and blue stripe. In terms of visual effect, KBR and TenSRDe provide better

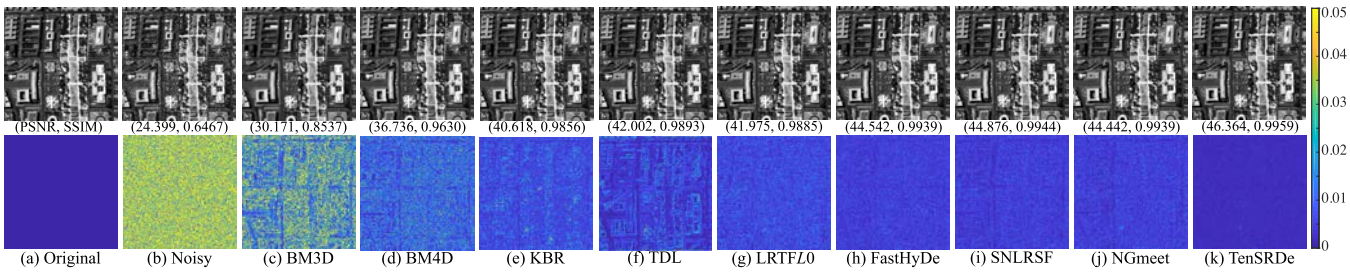


Fig. 12. (a) Original image and residual image of the WDC data set in band 163. (b) Noisy image and residual image under the Gaussian noise level $\sigma = 0.06$. Denoised images and residual images (the difference between each denoised band with the original band) of (c) BM3D, (d) BM4D, (e) KBR, (f) TDL, (g) LRTFL0, (h) FastHyDe, (i) SNLRSF, (j) NGmeet, and (k) TenSRDe.

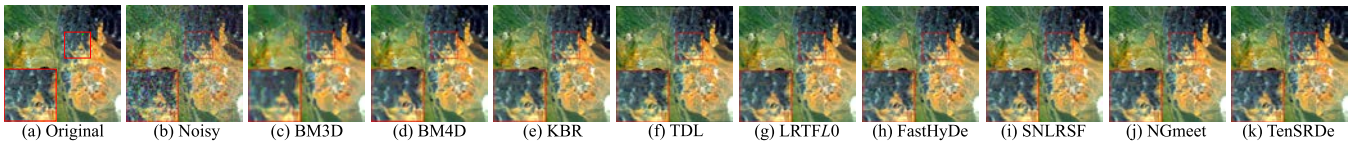


Fig. 13. (a) Original pseudocolor image (R: 80, G: 140, B: 180) of the Cuprite data set. (b) Noisy image with the Gaussian noise level $\sigma = 0.10$. Denoised results of (c) BM3D, (d) BM4D, (e) KBR, (f) TDL, (g) LRTFL0, (h) FastHyDe, (i) SNLRSF, (j) NGmeet, and (k) TenSRDe. To facilitate the observation of details, the demarcated area in each image is enlarged.

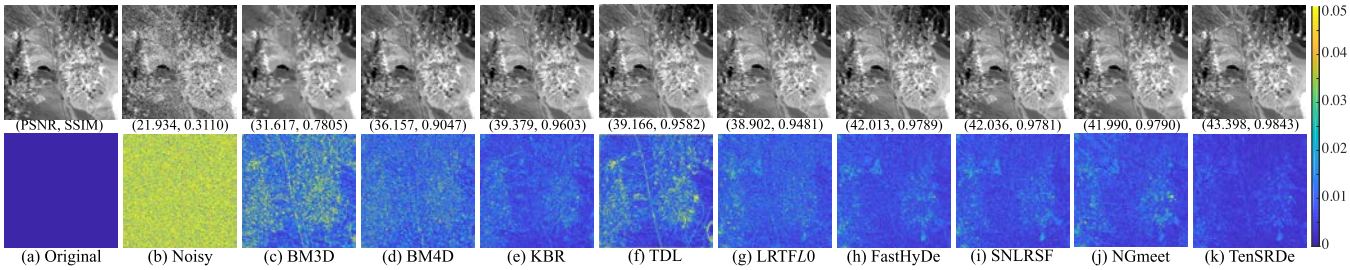


Fig. 14. (a) Original image and residual image of the Cuprite data set in band 126. (b) Noisy image and residual image under the Gaussian noise level $\sigma = 0.08$. Denoised images and residual images (the difference between each denoised band with the original band) of (c) BM3D, (d) BM4D, (e) KBR, (f) TDL, (g) LRTFL0, (h) FastHyDe, (i) SNLRSF, (j) NGmeet, and (k) TenSRDe.

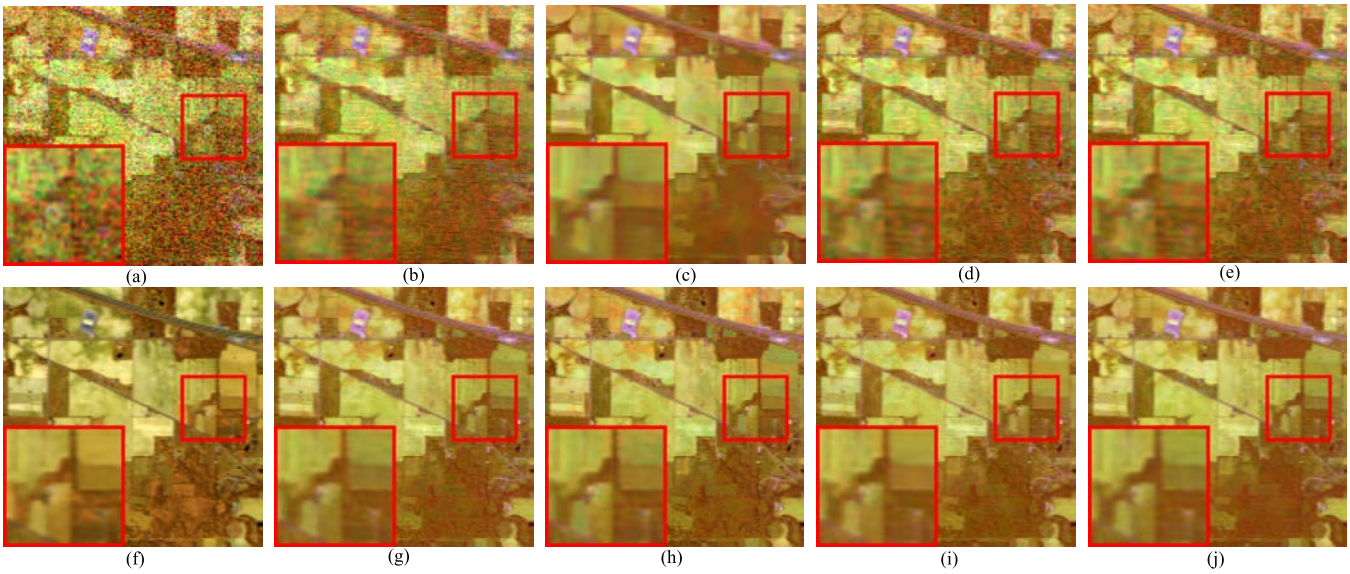


Fig. 15. (a) Pseudocolor image (R: 200, G: 144, B: 3) of the original Indian Pines data set. Pseudocolor image of the denoising results by (b) BM3D, (c) BM4D, (d) KBR, (e) TDL, (f) LRTFL0, (g) FastHyDe, (h) SNLRSF, (i) NGmeet, and (j) TenSRDe.

denoising results. From the enlarged areas of the band 205, we can see that BM3D and BM4D oversmooth the image, and TDL and FastHyDe remove some noise. Overall, KBR, SNLRSF, NGmeet, and TenSRDe have the best performance in noise removal and detail preservation. This also illustrates that the proposed TenSRDe can not only remove the Gaussian noise but also stripes in real experiments. To further demonstrate the denoising effect, Fig. 17 shows the horizontal mean profiles of the band 209 before and after denoising. From Fig. 17(a), due to the mixed noise corruption, there are rapid

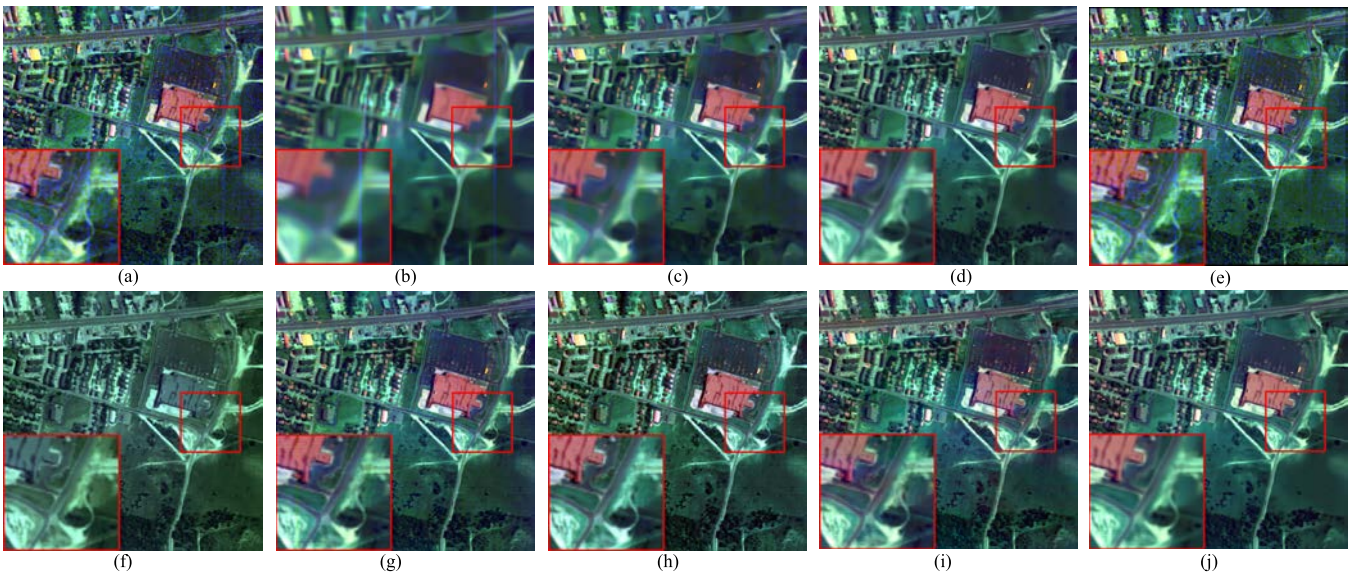


Fig. 16. (a) Pseudocolor image (R: 30, G: 110, B: 207) of the original Urban data set. Pseudocolor image of the denoising results by (b) BM3D, (c) BM4D, (d) KBR, (e) TDL, (f) LRTFL0, (g) FastHyDe, (h) SNLRSF, (i) NGmeet, and (j) TenSRDe.

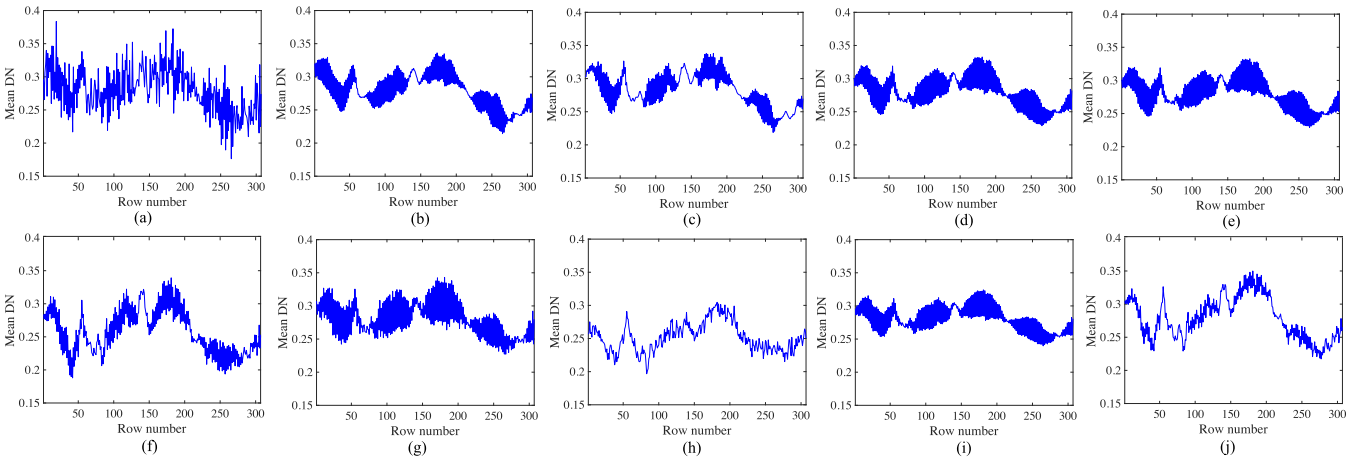


Fig. 17. Horizontal mean profiles of the band 209 on the Urban data set. (a) Original data. Denoising results of (b) BM3D, (c) BM4D, (d) KBR, (e) TDL, (f) LRTFL0, (g) FastHyDe, (h) SNLRSF, (i) NGmeet, and (j) TenSRDe.

fluctuations in the curve. After denoising, the fluctuations are suppressed by all the methods. It is easy to observe that SNLRSF and TenSRDe achieve smoother curves. The result is in accordance with the visual effect.

In addition, the no-reference measure index Q-metric [61] is employed to quantitatively evaluate the results of the real experiments, and a larger index value corresponds to a better denoising result. We compute the Q-metric value of each band and then set the mean value of the Q-metric as the final evaluation result. Table II lists the index values of all comparison methods for the Indian Pines and Urban data sets, and the highest values of the index are highlighted in bold. The proposed method obtains the best results.

TABLE II
NO-REFERENCE MEASURE INDEX Q-METRIC COMPARISON ON TWO DATA SETS

Dataset	Noisy	BM3D	BM4D	KBR	TDL
Indian	0.0914	0.1024	0.1013	0.1017	0.0974
Urban	0.0894	0.1088	0.1085	0.1092	0.0920
Dataset	LRTF$L0$	FastHyDe	SNLRSF	NGmeet	TenSRDe
Indian	0.1003	0.0999	0.0981	0.1022	**0.1026**
Urban	0.0939	0.0986	0.0955	0.1087	**0.1093**

C. Discussion

1) Influence of Different Modes: We analyze the properties of low-tubal rankness and NSS with spectral dimensions in different modes. We first define five permuted tensors. Given an HSI tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, there are the following five permuted tensors:

$$\vec{\mathcal{X}}_1 = \text{Permute}(\mathcal{X}, [2, 1, 3]) \in \mathbb{R}^{n_2 \times n_1 \times n_3}$$
$$\vec{\mathcal{X}}_2 = \text{Permute}(\mathcal{X}, [3, 1, 2]) \in \mathbb{R}^{n_3 \times n_1 \times n_2}$$
$$\vec{\mathcal{X}}_3 = \text{Permute}(\mathcal{X}, [3, 2, 1]) \in \mathbb{R}^{n_3 \times n_2 \times n_1}$$
$$\vec{\mathcal{X}}_4 = \text{Permute}(\mathcal{X}, [1, 3, 2]) \in \mathbb{R}^{n_1 \times n_3 \times n_2}$$
$$\vec{\mathcal{X}}_5 = \text{Permute}(\mathcal{X}, [2, 3, 1]) \in \mathbb{R}^{n_2 \times n_3 \times n_1}$$

where "Permute" is the MATLAB command.

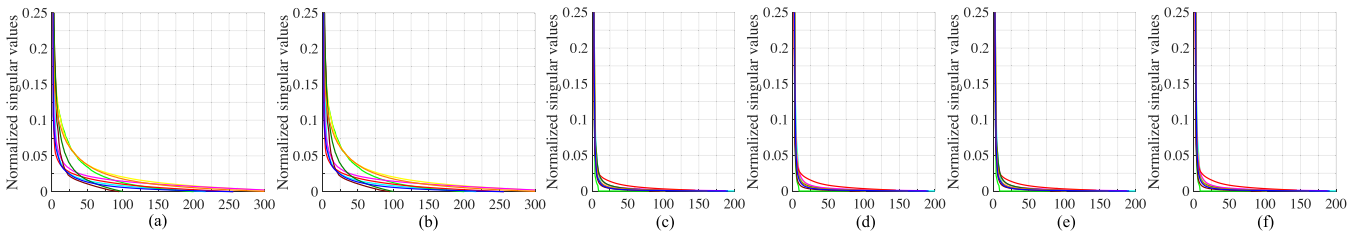


Fig. 18. Statistical analysis of low-tubal rankness on ten clean HSI data sets (the same color represents the same data set). (a) Normalized tensor singular value curve of original HSI tensor $\mathcal{X}$. (b)–(f) Normalized singular value curves of permuted HSI tensors $\vec{\mathcal{X}}_1$, $\vec{\mathcal{X}}_2$, $\vec{\mathcal{X}}_3$, $\vec{\mathcal{X}}_4$, and $\vec{\mathcal{X}}_5$, respectively.

Low-Tubal Rankness: To explore the low-tubal rankness of tensors when the spectral dimension is in different modes, we select ten clean HSI data sets and draw the tensor singular value[2] curves of original HSI tensor $\mathcal{X}$ and permuted HSI tensors $\vec{\mathcal{X}}_1$, $\vec{\mathcal{X}}_2$, $\vec{\mathcal{X}}_3$, $\vec{\mathcal{X}}_4$, and $\vec{\mathcal{X}}_5$, as shown in Fig. 18. From Fig. 18(c)–(f), we observe that permuted tensors $\vec{\mathcal{X}}_2$, $\vec{\mathcal{X}}_3$, $\vec{\mathcal{X}}_4$, and $\vec{\mathcal{X}}_5$ are more low-tubal-rank than original tensor $\mathcal{X}$ and permuted tensor $\vec{\mathcal{X}}_1$. This implies that the tensor whose mode-1 or mode-2 is the spectral dimension is more low-tubal-rank than the tensor whose mode-3 is the spectral dimension.

NSS Property: Since permuted tensors $\vec{\mathcal{X}}_2$, $\vec{\mathcal{X}}_3$, $\vec{\mathcal{X}}_4$, and $\vec{\mathcal{X}}_5$ are more low-tubal-rank, we learn the subspaces from them, and the corresponding coefficient tensors are $\mathcal{Z}_2$, $\mathcal{Z}_3$, $\mathcal{Z}_4$, and $\mathcal{Z}_5$, respectively. The WDC data set is selected to test the NSS of the coefficient tensors generated from four permuted tensors. We set $r = 6$ and then obtain the six horizontal slices of each coefficient tensor, as shown in Fig. 19. It is difficult to visually compare the NSS properties of the four coefficient tensors. Thus, we measure them in a quantitative way. Define the accumulation energy ratio (AER) of the top k singular values as

$$\mathrm{AER}_k = \frac{\sum_i^k \sigma_i}{\sum_i^n \sigma_i}$$

where σ_i is the ith singular value of a matrix and n is the number of singular values. We define that the matrix whose AER_{10} is in $[0, 0.7)$, $[0.7, 0.9)$, and $[0.9, 1]$ is nonlow-rank (Non-LR), weak low-rank (Weak LR), and strong low-rank (Strong LR), respectively, and then compare the NSS properties of four coefficient tensors as follows: 1) set 3-D block size $p = 5$ and 3-D block number $q = 150$ to find the similar $5 \times 5 \times 6$ 3-D blocks of $\mathcal{Z}_2$, $\mathcal{Z}_3$, $\mathcal{Z}_4$, and $\mathcal{Z}_5$; 2) unfold each 3-D block group to a reshaped matrix and compute the AER_{10} values of all matrices; and 3) for each coefficient tensor, divide all reshaped matrices into three classes according to AER_{10} and count the percentage of each class. All details are presented in Table III. The more the low-rank matrices of a coefficient tensor, the stronger the NSS property it has. From Table III, $\mathcal{Z}_4$ and $\mathcal{Z}_5$ possess stronger NSS properties than $\mathcal{Z}_2$ and $\mathcal{Z}_3$. This indicates that the coefficient tensor generated from a tensor whose mode-2 is the spectral dimension has stronger NSS properties than the coefficient tensor generated from a tensor whose mode-1 is the spectral dimension.

In summary, the permuted tensor whose mode-2 is the spectral dimension not only possesses the stronger low-tubal rankness and makes the generated tensor subspace have physical meaning (see Figs. 2 and 3), but also its coefficient tensor has NSS property. Therefore, we rearrange the spectral dimension to mode-2 in our method.

[2]We follow [43] in calling the entries on the diagonal of $\mathcal{S}(:,:,1)$ as the tensor singular values of $\mathcal{A}$, where $\mathcal{S}$ is from the t-SVD of $\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$.

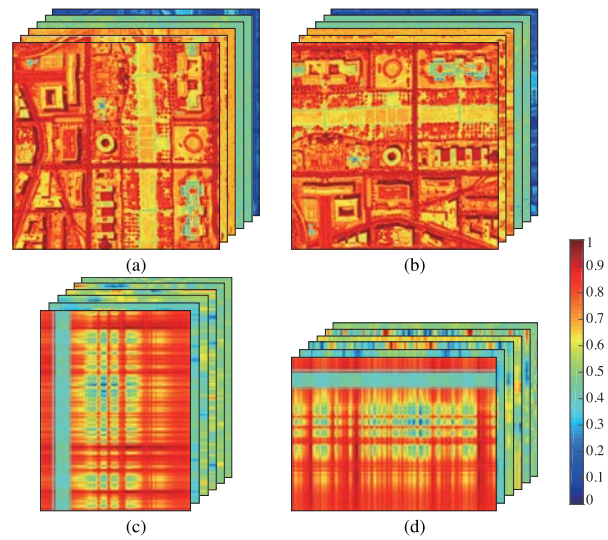


Fig. 19. Comparison of the NSS property on the WDC data set. (a)–(d) Horizontal slices of coefficient tensors $\mathcal{Z}_2$, $\mathcal{Z}_3$, $\mathcal{Z}_4$, and $\mathcal{Z}_5$, respectively.

TABLE III

INFORMATION ON THE RESHAPED MATRIX AND THE PERCENTAGE OF THE THREE KINDS OF ALL MATRICES

Coefficient tensor	Reshaped matrix		Percentage of different matrices		
	Size	Number	Non-LR	Weak LR	Strong LR
$\mathcal{Z}_2$	$150 \times (5^2 \times 6)$	4356	82.05	17.95	0
$\mathcal{Z}_3$	$150 \times (5^2 \times 6)$	4356	82.05	17.95	0
$\mathcal{Z}_4$	$150 \times (5^2 \times 6)$	3234	0.15	59.80	40.05
$\mathcal{Z}_5$	$150 \times (5^2 \times 6)$	3234	0.15	59.80	40.05

2) Matrix Subspace Versus Tensor Subspace: We analyze the representation capability of the matrix subspace and tensor subspace. Define the reconstruction error (RecErr) as

$$\mathrm{RecErr} = \frac{\|\mathcal{R}_i - \mathcal{O}\|_F}{\|\mathcal{O}\|_F}$$

where $\mathcal{O}$ is the original image and $\mathcal{R}_i$ is the reconstructed image with subspace dimensions i. Fig. 20(a) and (b) shows the reconstruction MPSNR and RecErr with respect to the different subspace dimensions, respectively. From Fig. 20, the higher reconstruction MPSNR and lower RecErr illustrate the strong representation capability of the tensor subspace.

Furthermore, we use FastHyDe to initialize the matrix subspace in NGmeet and give the MPSNR comparison of

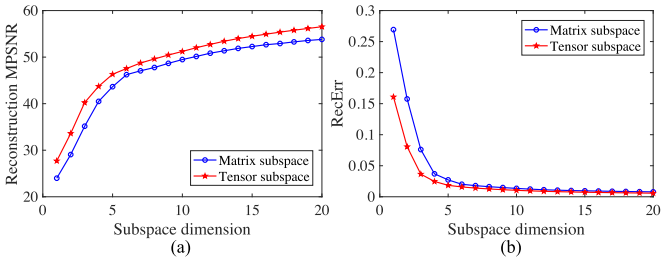


Fig. 20. Presentation capability comparison of the matrix and tensor subspace. (a) Reconstruction MPSNR with respect to the subspace dimension. (b) RecErr with respect to the subspace dimension.

TABLE IV

MPSNR OF INTI-NGMEET AND TENSRDE RESULTS ON ALL DATA SETS WITH SIMULATED NOISE

Dataset	Method	σ=0.02	σ=0.04	σ=0.06	σ=0.08	σ=0.10
PaC	INIT-NGmeet	48.29	43.96	41.31	39.57	38.25
	TenSRDe	49.24	44.64	41.94	40.02	38.65
WDC	INIT-NGmeet	49.98	45.00	42.43	40.58	39.15
	TenSRDe	51.62	46.40	43.75	41.74	40.26
Cuprite	INIT-NGmeet	46.24	43.98	42.56	41.11	40.08
	TenSRDe	47.75	45.28	43.47	41.95	40.83

TABLE V

MPSNR OF OUR METHOD UNDER 2-D-PATCH AND 3-D-BLOCK METHODS ON THE PAC DATA SET WITH SIMULATED NOISE

Method	σ=0.02	σ=0.04	σ=0.06	σ=0.08	σ=0.10
2D-patch	47.83	43.38	40.26	38.09	36.57
3D-block	49.25	44.64	41.94	40.02	38.65

the results of the "Initialization+NGmeet" (INIT-NGmeet) and TenSRDe methods, as presented in Table IV. Compared with the result of NGmeet in Table I, INIT-NGmeet has a slight improvement in MPSNR, which means that the representation capability of the matrix subspace reaches the limit. From Table IV, the relatively better performance of the proposed TenSRDe verifies the superiority of the tensor subspace in representation capability.

3) 2-D Patch Versus 3-D Block: In the process of coefficient tensor denoising, there are two ways to find similar patches/blocks. The first way is to search the similar 2-D patches in each band and denoise each similar 2-D patch group in a band-by-band manner. The second way is to search the similar 3-D blocks throughout all bands and then denoise each similar 3-D block group. To test the performance difference of the two ways in TenSRDe, we use them respectively in the coefficient tensor denoising. Table V lists the denoising results on the PaC data set. The superior results of the 3-D-block show that, in our method, 3-D blocks mine more similar information in the coefficient tensor, which can greatly boost the denoising effect.

4) Initialization Method: We study the influence of different initialization methods on the denoising results of our method. Advanced and efficient Gaussian noise removal methods BM3D [10], BM4D [54], and FastHyDe [34] are selected to initialize the basis tensor. Table VI presents the MPSNR and average time (the average value of the running time for all noise levels) of our method combined with different initialization methods under five noise levels on the PaC data set. It can be seen that a better initialization can lead to better

TABLE VI

MPSNR AND AVERAGE TIME (IN MINUTES) OF OUR METHOD COMBINED WITH DIFFERENT INITIALIZATION METHODS ON THE PAC DATA SET WITH SIMULATED NOISE

Initialization method	MPSNR					Average time
	σ=0.02	σ=0.04	σ=0.06	σ=0.08	σ=0.10	
BM3D	46.28	40.64	37.29	35.34	34.04	0.232
BM4D	48.81	44.15	40.72	39.28	37.67	0.408
FastHyDe	49.25	44.64	41.94	40.02	38.65	0.318

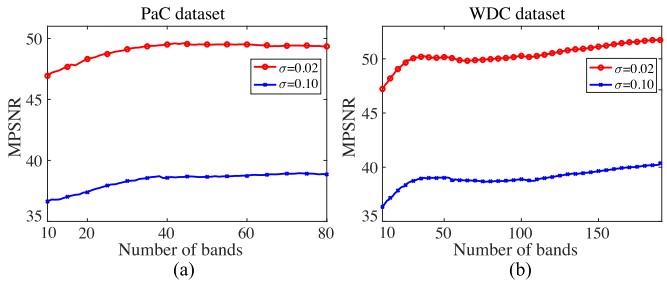


Fig. 21. Influence of the number of spectral bands on the denoising effect. (a) PaC data set. (b) WDC data set.

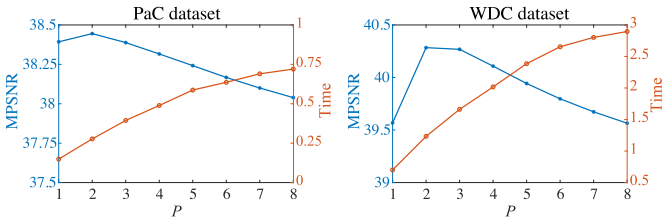


Fig. 22. Influence of the value of P on the denoising effect and efficiency.

denoising performance. Considering the effect and efficiency, we use FastHyDe to initialize the basis tensor in our method.

5) Effects of Number of Bands: We discuss the influence of the number of spectral bands on the denoising effect. Fig. 21 shows the changes in the MPSNR values with different numbers of bands (from the tenth to the last) on the PaC and WDC data sets under two noise levels. From Fig. 21(a) and (b), it can be seen that the denoising results become gradually better as the number of bands increases. This implies that more bands can provide more information, which can help improve the denoising effect of the proposed method.

6) Effects of the Value of P: We discuss the influence of the value of P in Algorithm 1 on the denoising effect and efficiency of PAM. Fig. 22 shows the changes in the MPSNR values and time (in minutes) with different values of P on the PaC and WDC data sets under noise level $\sigma = 0.10$. It can be observed that as the value of P increases, the value of MPSNR varies slightly, and the time cost increases rapidly. Considering the denoising effect and efficiency, we set $P = 2$ in all experiments.

7) Parameter Analysis: There are eight parameters in our method, including regularization parameter λ, penalty parameter γ, proximal parameter ρ, iterative regularization parameter θ, 3-D block size p, 3-D block number q, initial subspace dimension r, and increment δ. Problem (9) is a weighted nuclear norm proximal problem. According to [12], parameter $\lambda/[\gamma(1+\rho)]$ in (9) corresponds to noise variance σ^2. Thus, we set $\lambda = \gamma(1+\rho)\sigma^2$ in all experiments. We set

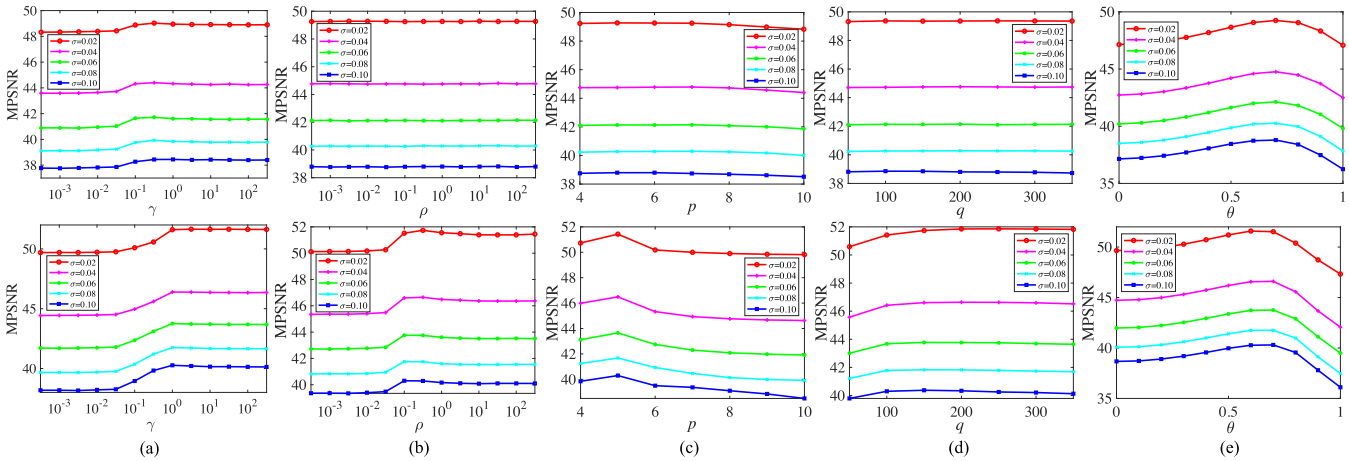


Fig. 23. Sensitivity analysis of the parameters on the PaC and WDC data sets under different noise levels. (a) MPSNR versus proximal parameter γ. (b) MPSNR versus proximal parameter ρ. (c) MPSNR versus 3-D block size p. (d) MPSNR versus 3-D block number q. (e) MPSNR versus iterative regularization parameter θ. (Top) PaC data set. (Bottom) WDC data set.

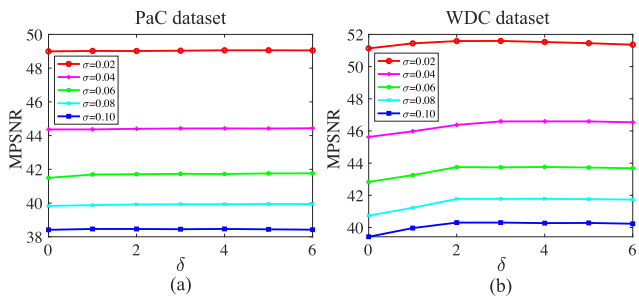


Fig. 24. Sensitivity analysis of parameters δ under different noise levels. (a) MPSNR versus parameter δ on the PaC data set. (b) MPSNR versus parameter δ on the WDC data set.

$r = \lfloor \sqrt[3]{n_1 n_2 n_3}/(120\sqrt{\sigma}) \rfloor$ in all experiments, where σ is the noise level, and n_1, n_2, and n_3 are the HSI sizes. We then analyze the sensitivity of γ, ρ, p, and q and the effectiveness of θ and δ on different data sets.

Sensitivity Analysis of Parameters γ, ρ, p, and q: Penalty parameter γ is used in the ADMM algorithm, ρ is used in the PAM algorithm to adjust the proximal term that can guarantee the algorithm stability, and p and q, respectively, measure the size and number of similar 3-D blocks in coefficient tensor denoising. Fig. 23(a)–(d) presents the MPSNR curves with respect to the different values of γ, ρ, p, and q under five noise levels. It is clear that all MPSNR curves corresponding to the different parameters are slightly fluctuating. To obtain the best performance on the different data sets, we empirically set $\gamma = 5$, $\rho = 0.5$, $p = 5$, and $q = 200$ in all experiments.

Effectiveness of Parameter θ: In our method, iterative regularization is adopted to refine the subspace in each iteration, and θ balances the denoised image and noisy image. A proper value of θ can more effectively improve the denoising effect. Fig. 23(e) shows the MPSNR curves when θ is changed from 0 to 1 with an increment 0.1 under five noise levels. We observe that the MPSNR achieves the highest value when θ is 0.7, and then, the iterative regularization parameter is set to $\theta = 0.7$ in all experiments.

Effectiveness of Parameter δ: The increase in the subspace dimension means that more original information can be extracted with the iteration, and then, the performance of denoising can be improved. Fig. 24 shows the MPSNR with respect to increased δ on the PaC and WDC data sets under five noise levels. We observe that, compared with the fixed subspace dimension ($\delta = 0$), increasing subspace dimension can improve the denoising effect. However, a larger δ leads to a higher time cost. To balance the denoising effect and efficiency, we set $\delta = 3$ in all experiments.

TABLE VII

PARAMETER SETTINGS OF TENSRDE IN ALL EXPERIMENTS

Parameter	r			λ		
Setting	$\lfloor \sqrt[3]{n_1 n_2 n_3}/(120\sqrt{\sigma}) \rfloor$			$\lambda = \gamma(1+\rho)\sigma^2$		
Parameter	γ	ρ	p	q	θ	δ
Setting	5	0.5	5	200	0.7	3

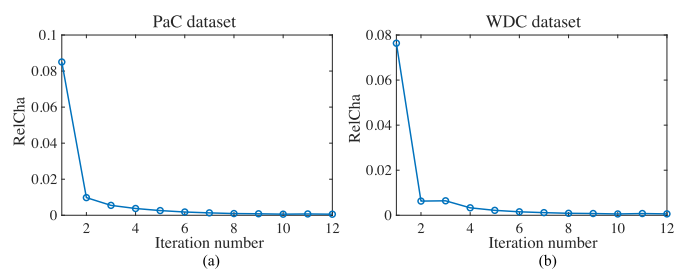


Fig. 25. RelCha in each iteration on two data sets. (a) PaC data set. (b) WDC data set.

8) Convergence Behavior: We empirically analyze the convergence of the PAM algorithm. The relative change (RelCha) in the kth iteration is defined as

$$\text{RelCha} = \frac{\|\mathcal{A}^k * \mathcal{Z}^k - \mathcal{A}^{k-1} * \mathcal{Z}^{k-1}\|_F}{\|\mathcal{A}^{k-1} * \mathcal{Z}^{k-1}\|_F}.$$

Fig. 25(a) and (b) shows the RelCha curve in our algorithm on the PaC and WDC data sets with noise level $\sigma = 0.02$. It is clear that each RelCha rapidly tends to zero. This confirms the global convergence of our algorithm.

V. Conclusion

In this work, we proposed a TenSR-based HSI denoising method for Gaussian noise removal. The proposed method advances the HSI denoising field in two aspects: subspace representation capability and computational complexity. The proposed basis tensor, which is learned from the tensor, avoids the loss of intrinsic information. The proposed method projects the original HSI onto a low-dimensional tensor subspace and denoises the projected coefficient tensor, which greatly reduces the computational complexity. Although the nonconvex model seems to be difficult to solve, we develop an efficient PAM algorithm and theoretically prove its global convergence. Compared with the current state-of-the-art methods, extensive experiments indicate that the proposed method significantly improves the denoising performance in terms of qualitative and quantitative evaluation. In the future, we will extend our model to further improve its noise removal ability, e.g., the removal of impulse noise, stripes, and deadlines.

Acknowledgment

The authors would like to thank the editors and the anonymous reviewers for their constructive comments that greatly helped to improve the quality of this article.

References

[1] H. Zhang, L. Liu, W. He, and L. Zhang, "Hyperspectral image denoising with total variation regularization and nonlocal low-rank tensor decomposition," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3071–3084, May 2020.

[2] Y. Chang, L. Yan, H. Fang, S. Zhong, and W. Liao, "HSI-DeNet: Hyperspectral image restoration via convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 667–682, Dec. 2019.

[3] Q. Yuan, Q. Zhang, J. Li, H. Shen, and L. Zhang, "Hyperspectral image denoising employing a spatial–spectral deep residual convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1205–1218, Sep. 2019.

[4] X.-L. Zhao, W.-H. Xu, T.-X. Jiang, Y. Wang, and M. K. Ng, "Deep plug-and-play prior for low-rank tensor completion," *Neurocomputing*, vol. 400, pp. 137–149, Dec. 2020.

[5] L. Sun *et al.*, "Low rank component induced spatial–spectral kernel method for hyperspectral image classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3829–3842, Oct. 2020.

[6] H. Zhang, J. Li, Y. Huang, and L. Zhang, "A nonlocal weighted joint sparse representation classification method for hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2056–2065, Jun. 2014.

[7] L. Zhuang, C. Lin, M. A. T. Figueiredo, and J. M. Bioucas-Dias, "Regularization parameter selection in minimum vol. hyperspectral, unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 9858–9877, Dec. 2019.

[8] Y. Zhang, L. Zhang, B. Du, and S. Wang, "A nonlinear sparse representation-based binary hypothesis model for hyperspectral target detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2513–2522, Jun. 2015.

[9] J. Li, Q. Yuan, H. Shen, and L. Zhang, "Noise removal from hyperspectral image with joint Spectral–Spatial distributed sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5425–5439, Sep. 2016.

[10] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[11] M. Lebrun, A. Buades, and J. M. Morel, "A nonlocal Bayesian image denoising algorithm," *SIAM J. Imag. Sci.*, vol. 6, no. 3, pp. 1665–1688, Jan. 2013.

[12] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, "Weighted nuclear norm minimization and its applications to low level vision," *Int. J. Comput. Vis.*, vol. 121, no. 2, pp. 183–208, Jan. 2017.

[13] P. Parul, "Hyperspectral image denoising with a spatial–spectral view fusion strategy," *Int. J. Eng. Comput. Sci.*, vol. 15, pp. 2314–2325, Aug. 2015.

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

[15] W. He, H. Zhang, L. Zhang, and H. Shen, "Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 178–188, Jan. 2016.

[16] Y. Xie, Y. Qu, D. Tao, W. Wu, Q. Yuan, and W. Zhang, "Hyperspectral image restoration via iteratively regularized weighted schatten p-norm minimization," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4642–4659, Aug. 2016.

[17] Y. Chen, Y. Guo, Y. Wang, D. Wang, C. Peng, and G. He, "Denoising of hyperspectral images using nonconvex low rank matrix approximation," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5366–5380, Sep. 2017.

[18] R. Dian and S. Li, "Hyperspectral image super-resolution via subspace-based low tensor multi-rank regularization," *IEEE Trans. Image Process.*, vol. 28, no. 10, pp. 5135–5146, Oct. 2019.

[19] J.-H. Yang, X.-L. Zhao, J.-J. Mei, S. Wang, T.-H. Ma, and T.-Z. Huang, "Total variation and high-order total variation adaptive model for restoring blurred images with cauchy noise," *Comput. Math. with Appl.*, vol. 77, no. 5, pp. 1255–1272, Mar. 2019.

[20] L.-B. Cui, X.-Q. Zhang, and S.-L. Wu, "A new preconditioner of the tensor splitting iterative method for solving multi-linear systems with $\mathcal{M}$-tensors," *Comput. Appl. Math.*, vol. 39, no. 3, p. 173, 2020.

[21] 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.

[22] T.-X. Jiang, M. K. Ng, X.-L. Zhao, and T.-Z. Huang, "Framelet representation of tensor nuclear norm for third-order tensor completion," *IEEE Trans. Image Process.*, vol. 29, pp. 7233–7244, 2020, doi: 10.1109/TIP.2020.3000349.

[23] W. He, N. Yokoya, L. Yuan, and Q. Zhao, "Remote sensing image reconstruction using tensor ring completion and total variation," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8998–9009, Nov. 2019.

[24] J.-H. Yang, X.-L. Zhao, T.-Y. Ji, T.-H. Ma, and T.-Z. Huang, "Low-rank tensor train for tensor robust principal component analysis," *Appl. Math. Comput.*, vol. 367, p. 124783, 2020.

[25] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[26] N. Renard, S. Bourennane, and J. Blanc-Talon, "Denoising and dimensionality reduction using multilinear tools for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 5, no. 2, pp. 138–142, Apr. 2008.

[27] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, Aug. 2009.

[28] X. Liu, S. Bourennane, and C. Fossati, "Denoising of hyperspectral images using the PARAFAC model and statistical performance analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 3717–3724, Oct. 2012.

[29] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of Eckart-Young decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, Sep. 1970.

[30] H. Fan, Y. Chen, Y. Guo, H. Zhang, and G. Kuang, "Hyperspectral image restoration using low-rank tensor recovery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 10, pp. 4589–4604, Oct. 2017.

[31] M. E. Kilmer and C. D. Martin, "Factorization strategies for third-order tensors," *Linear Algebra Appl.*, vol. 435, no. 3, pp. 641–658, Aug. 2011.

[32] Q. Zhang, Q. Yuan, J. Li, X. Liu, H. Shen, and L. Zhang, "Hybrid noise removal in hyperspectral imagery with a spatial–spectral gradient network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7317–7329, Dec. 2019.

[33] L. Sun and B. Jeon, "A novel subspace spatial-spectral low rank learning method for hyperspectral denoising," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2017, pp. 1–4.

[34] L. Zhuang and J. M. Bioucas-Dias, "Fast hyperspectral image denoising and inpainting based on low-rank and sparse representations," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 730–742, Mar. 2018.

[35] W. He, Q. Yao, C. Li, N. Yokoya, and Q. Zhao, "Non-local meets global: An integrated paradigm for hyperspectral denoising," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6861–6870.

[36] C. Cao, J. Yu, C. Zhou, K. Hu, F. Xiao, and X. Gao, "Hyperspectral image denoising via subspace-based nonlocal low-rank and sparse factorization," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 3, pp. 973–988, Mar. 2019.

[37] Q. Zhao *et al.*, "A novel sparsity measure for tensor recovery," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 271–279.

[38] Y.-B. Zheng, T.-Z. Huang, X.-L. Zhao, T.-X. Jiang, T.-H. Ma, and T.-Y. Ji, "Mixed noise removal in hyperspectral image via low-fibered-rank regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 734–749, Jan. 2020.

[39] Q. Xie, Q. Zhao, D. Meng, and Z. Xu, "Kronecker-basis-representation based tensor sparsity and its applications to tensor recovery," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1888–1902, Aug. 2018.

[40] Y.-B. Zheng, T.-Z. Huang, X.-L. Zhao, T.-X. Jiang, T.-Y. Ji, and T.-H. Ma, "Tensor N-tubal rank and its convex relaxation for low-rank tensor recovery," *Inf. Sci.*, vol. 532, pp. 170–189, Sep. 2020.

[41] K. Braman, "Third-order tensors as linear operators on a space of matrices," *Linear Algebra Appl.*, vol. 433, no. 7, pp. 1241–1253, Dec. 2010.

[42] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, and L.-J. Deng, "Multi-dimensional imaging data recovery via minimizing the partial sum of tubal nuclear norm," *J. Comput. Appl. Math.*, vol. 372, Jul. 2020, Art. no. 112680.

[43] 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.

[44] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Math. Program.*, vol. 146, nos. 1–2, pp. 459–494, 2014.

[45] K. Eriksson, D. Estep, and C. Johnson, *Lipschitz Continuity*. Berlin, Germany: Springer, 2004.

[46] Y. Wang, W. Yin, and J. Zeng, "Global convergence of ADMM in nonconvex nonsmooth optimization," *J. Sci. Comput.*, vol. 78, no. 1, pp. 29–63, Jan. 2019.

[47] Y. Chang, L. Yan, and S. Zhong, "Hyper-laplacian regularized unidirectional low-rank tensor recovery for multispectral image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4260–4268.

[48] K. Fan and A. J. Hoffman, "Some metric inequalities in the space of matrices," in *Proc. Amer. Math. Soc.*, vol. 6, no. 1, pp. 111–116, 1955.

[49] E. J. Candes, X. Li, and M. Soltanolkotabi, "Phase retrieval via wirtinger flow: Theory and algorithms," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1985–2007, Apr. 2015.

[50] Y. Chen and E. J. Candès, "Solving random quadratic systems of equations is nearly as easy as solving linear systems," *Commun. Pure Appl. Math.*, vol. 70, no. 5, pp. 822–883, May 2017.

[51] W. Miao. (2013). *Matrix Completion Models With Fixed Basis Coefficients and Rank Regularized Problems With Hard Constraints*. [Online]. Available: https://scholarbank.nus.edu.sg/handle/10635/37889

[52] H. Attouch, J. Bolte, and B. F. Svaiter, "Convergence of descent methods for semi-algebraic and tame problems: Proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods," *Math. Program.*, vol. 137, nos. 1–2, pp. 91–129, Feb. 2013.

[53] Y. Yang, Y. Feng, and J. A. K. Suykens, "Robust low-rank tensor recovery with regularized redescending M-estimator," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 9, pp. 1933–1946, Sep. 2016.

[54] M. Maggioni, V. Katkovnik, K. Egiazarian, and A. Foi, "Nonlocal transform-domain filter for volumetric data denoising and reconstruction," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 119–133, Jan. 2013.

[55] Y. Peng, D. Meng, Z. Xu, C. Gao, Y. Yang, and B. Zhang, "Decomposable nonlocal tensor dictionary learning for multispectral image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2949–2956.

[56] F. Xiong, J. Zhou, and Y. Qian, "Hyperspectral restoration via L_0 gradient regularized low-rank tensor factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 10410–10425, Dec. 2019.

[57] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[58] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.

[59] L. Wald, *Data Fusion: Definitions Architectures: Fusion Images Different Spatial Resolutions*. Paris, France: École des Mines de Paris, 2002.

[60] J. M. Bioucas-Dias and J. M. P. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sensing*, vol. 46, no. 8, pp. 2435–2445, Aug. 2008.

[61] X. Zhu and P. Milanfar, "Automatic parameter selection for denoising algorithms using a no-reference measure of image content," *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3116–3132, Dec. 2010.

Jie Lin (Graduate Student Member, IEEE) received the B.S. degree in information and computing science from the Anhui University of Finance and Economics, Bengbu, China, in 2018. He is pursuing the Ph.D. degree with the School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, China.

His research interests include low-rank and sparse modeling, tensor decomposition, and high-dimensional image processing.

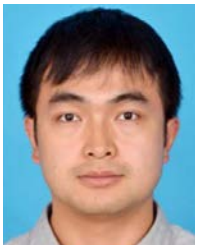
Ting-Zhu Huang received the B.S., M.S., and Ph.D. degrees in computational mathematics from the Department of Mathematics, Xi'an Jiaotong University, Xi'an, China.

He is a Professor with the School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, China. His research interests include scientific computation and applications, numerical algorithms for image processing, numerical linear algebra, preconditioning technologies, and matrix analysis with applications.

Dr. Huang is an Editor of the *Scientific World Journal*, *Advances in Numerical Analysis*, the *Journal of Applied Mathematics*, the *Journal of Pure and Applied Mathematics: Advances in Applied Mathematics*, and the *Journal of Electronic Science and Technology*, China.

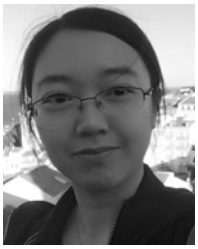
Xi-Le Zhao received the M.S. and Ph.D. degrees from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2009 and 2012, respectively.

He is a Professor with the School of Mathematical Sciences, UESTC. His research interests include image processing, computer vision, and machine learning.



Tai-Xiang Jiang (Member, IEEE) received the B.S. and Ph.D. degrees in mathematics and applied mathematics from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2013.

He is an Associate Professor with the School of Economic Information Engineering, Southwestern University of Finance and Economics, Chengdu. His research interests include sparse and low-rank modeling, tensor decomposition and multidimensional image processing. More information can be found on his homepage https://sites.google.com/view/taixiangjiang/.



Lina Zhuang (Member, IEEE) received the bachelor's degrees in geographic information system and in economics from South China Normal University, Guangzhou, China, in 2012, the M.S. degree in cartography and geography information system from the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing, China, in 2015, and the Ph.D. degree in electrical and computer engineering from the Instituto Superior Tecnico, Universidade de Lisboa, Lisbon, Portugal, in 2018.

Since 2015, she has been with the Instituto de Telecomunicações, as a Marie Curie Early Stage Researcher of Sparse Representations and Compressed Sensing Training Network (SpaRTaN number 607290). Currently, she is a Research Assistant Professor with Hong Kong Baptist University, Hong Kong.