



HYPERSPECTRAL IMAGE DENOISING VIA WEIGHTED DOUBLE SPARSITY TOTAL VARIATION AND LOW-RANK REPRESENTATION

JIE HUANG^{✉1}, KE-HAN CHEN^{✉1}, JIN-JU WANG^{✉2} AND WEN YAN^{✉1}

¹School of Mathematical Sciences, University of Electronic Science and Technology of China,
Chengdu, Sichuan, 611731, China

²School of Mathematics, Southwest Minzu University, Chengdu, Sichuan, 610225, China

(Communicated by Weihong Guo)

ABSTRACT. Hyperspectral images (HSIs) are always contaminated by various mixed noise, which degrades the quality of acquired images and seriously affects the subsequent extensive applications. Total variation (TV) is popular for its capability of preserving details and promoting smoothness in HSI denoising. However, TV may cause over-smoothness and details loss. To tackle the above problems, we propose a double sparsity TV and low-rank representation denoising model (LRDSTV) for the mixed noise removal. Specifically, the double sparsity TV means fiber sparsity with sparse fibers in the gradient domain, promoting piecewise smooth structures and properly using the spatial information of the HSI. Moreover, we utilize the weighted nuclear norm to explore the low-rank property of mode-3 unfolding of the HSI, taking advantage of the spectral correlation and helping maintain more details to avoid over-smoothing. Then, the alternating direction method of multipliers (ADMM) is applied for the optimization of the LRDSTV model. Finally, a series of denoising experiments on simulated and real data sets demonstrate the effectiveness and superiority of the proposed algorithm compared with some state-of-the-art algorithms.

1. Introduction. Hyperspectral images (HSIs) are obtained by imaging spectrometers that utilizing hundreds of continuous bands image the target area simultaneously. There are many researches on this special image data [14, 24, 36, 38, 39]. The plentiful spectral information of HSIs leads to various applications, e.g., agriculture [9], exploration of oil and gas [41], cultural relics protection [35] and food quality control [13]. However, it will inevitably be affected by various factors during its imaging process. As a result, HSIs are easily polluted by mixed noise, including Gaussian noise, impulse noise, stripes, and deadlines [56]. HSIs contaminated by mixed noise will lost a lot of useful information and cause a very serious impact on subsequent applications. Therefore, denoising processing of HSIs is a prerequisite for later steps and has attracted attentions of many researchers.

2020 *Mathematics Subject Classification.* Primary: 94A08, 68U10; Secondary: 49N45.

Key words and phrases. Hyperspectral images, mixed noise, double sparsity, total variation.

This research is supported in part by NSFC under Grant 12371517 and 12171072 and in part by National Key Research and Development Program of China under Grant 2020YFA0714001.

*Corresponding author: Jie Huang.

Each band of HSI is treated as a grayscale image. Some straightforward denoising approaches are proposed by denoising each band separately [8, 11, 31]. These methods ignore the strong correlation of spectral bands, and so hard to achieve a satisfactory result. So as to put this significant spectral correlation into consideration, various methods have been proposed. A method based on hybrid spatial-spectral derivative-domain wavelet shrinkage has been proposed in [34]. This method applies the dissimilarity of the signal regularity in the spatial and the spectral dimensions of HSIs. In addition, a spatial-spectral adaptive sparse representation method in [29] investigates the highly correlated spectral information and similar spatial information via sparse representation. In [55], a spectral-spatial adaptive total variation (TV) model has been presented to consider the spectral noise differences and spatial information differences.

However, above methods all focus on eliminating almost only single zero-mean Gaussian noise and neglect the complexity of noise in the real world. In addition to Gaussian noise, many noise such as impulse noise, stripes, deadlines, and other kinds of noise also exist in the real scenes. In order to deal with these complex noise, the robust principal component analysis (RPCA) framework [3] has been proposed and used in mixed noise elimination. Affected by RPCA, Zhang *et al.* proposes a low-rank matrix recovery (LRMR) model [56] to remove mixed noise. LRMR successfully explores the low-rank property of clean HSIs and obtains impressive denoising performance. Then, the matrix-factorization-based methods are gradually developed, for example, in [10], a bandwise noise model combined with low-rank matrix factorization has been proposed, and a framelet-regularized low-rank nonnegative matrix factorization model has been presented in [7]. Moreover, many other researchers apply matrix factorization into their models [12, 19, 50, 54].

The TV regularization has the ability of preserving edge information and promoting piecewise smoothness. Noisy images with excessive and possibly spurious detail have higher total variation, thus we can reduce the TV of original images to remove the redundant details. At the same time, the important details like the edge details are reserved. According to these features, TV is widely used in the area of HSI denoising. Over these years, a popular structure of HSI denoising is the combination of the TV regularization and a low-rank constraint. He *et al.* in [20] absorb above two important aspects well, and the classic band-by-band TV regularization has proved its denoising performance. Later in [55] and [1], the spatial-spectral TV regularization that developing from the original one catches the spectral smooth structure. And in [43] and [18], different low rank terms matching with spatial-spectral TV regularization both achieve nice performance.

Being an efficient tool for depicting global correlation of HSIs, tensor decomposition is widely used these years. There are so many methods, to name a few, [2, 5, 16, 25, 28, 32, 52, 53], combining different forms of tensor decomposition with other regularization, for example, nonlocal similarity with tucker decomposition, group sparse with low-rank tensor decomposition and other various terms. In addition, we see the anisotropic spatial-spectral TV regularization combining with tensor decomposition in [44]. Moreover, Chen *et al.* in [4] propose a weighted group sparsity-regularized low-rank tensor decomposition model (LRTDGS), which tries to employ the group sparsity information of difference images. Furthermore, many deep learning (DL)-based algorithms have been proposed to improve the restoration performance [30, 33, 40]. And all of them have the state-of-the-art denoising results.

However, few problems still exist. The TV regularization is a double-edged sword in dealing with HSI denoising. On the one hand, it preserves edge information and promotes piecewise smoothness. On the other hand, excessive use of TV results in loss of details and over-smoothing [57]. So, an emergent point is to balance the smooth structure and the details in denoising results. An immediate way is to express the sparsity in the gradient domain more cautiously, and utilizing the low-rank regularization to maintain more details is the other remedy.

In this paper, we propose a double sparsity TV and low-rank representation (LRDSTV) model for HSI mixed noise removal. The concept of double sparsity TV is first introduced to employ fiber sparsity with sparse fibers simultaneously, emphasizing both zero fibers in the difference images of the HSI and zero values in the remaining non-zero fibers. Then, we utilize the weighted ℓ_1 -norm to enhance the double sparsity with a self-adaptive reweighting strategy. Developing from the original TV regularization, the weighted double sparsity TV promotes smooth structures and properly utilizes the spatial and spectral information of the HSI. Additionally, we use the weighted nuclear norm to depict the low-rank property of the mode-3 unfolding of the HSI, thus exploiting the spectral correlation of the HSI and helping maintain more details to avoid over-smoothing. We propose two alternating direction method of multipliers (ADMM) based algorithms to solve the proposed LRDSTV model. The denoising performance in the simulated and real experiments all show the ability of the proposed algorithms in preserving details and removing mixed noise.

The contribution of this paper is summarized in the following two aspects.

- 1) A double sparsity TV regularization is proposed for better describing the sparse attributes of spatial difference images, which is presented as the weighted ℓ_1 -norm. The double sparsity structure not only uses the fiber sparsity in the entire differential image of the HSI, but also takes the sparsity of individual fibers into consideration.
- 2) A novel LRDSTV model is proposed for the HSI mixed noise removal problem, combining the weighted double sparsity TV with low-rank representation. Two ADMM based optimization algorithms, named LRDSTV₁ and LRDSTV₂, are adopted for solving the proposed model. The denoising performance in the simulated and real experiments all show the ability of the proposed algorithms in preserving details and removing mixed noise.

The rest of this paper is arranged as follows. Section 2 introduces the notations and preliminaries at first, and the related HSI denoising knowledge are presented then. Section 3 contains the motivation of the proposed LRDSTV model and two ADMM based algorithms. Section 4 shows the denoising results of LRDSTV in both simulated and real experiments. At last, Section 5 summarizes this article.

2. Notations and related work.

2.1. Problem formulation. We will introduce some notations and preliminaries in this part. We use a lowercase letter to represent a vector, for example, w . We use a capitalized letter to present a matrix, for instance, X . As known, tensor is a kind of multidimensional data, which is seen as the promotion of vectors and matrices in multidimensional spaces, and will be denoted by a capitalized calligraphic letter. For a tensor with N dimensions, which is denoted as $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$, and every

element of \mathcal{X} is x_{i_1, i_2, \dots, i_N} . Moreover, the mode- n unfolding of a tensor \mathcal{X} reshapes \mathcal{X} into a matrix, which is presented by $X_{(n)} \in \mathbb{R}^{I_n \times (I_1 \cdots I_{n-1} I_{n+1} \cdots I_N)}$. A fiber refers to the vector extracted from the tensor. Fixing one of the dimensions in the matrix can get rows or columns. Similar to the matrix operation, fixing other dimensions, and only keeping one dimension change in a tensor \mathcal{X} can obtain a fiber. For example, a 3-D tensor \mathcal{X} has three kinds of fibers, and they are $\mathcal{X}(:, j, k)$, $\mathcal{X}(i, :, k)$, and $\mathcal{X}(i, j, :)$. Detailed information about tensors can be found in [23]. In addition, the Frobenius norm of a tensor \mathcal{X} is defined as $\|\mathcal{X}\|_F = (\sum_{i_1, \dots, i_N} |x_{i_1, \dots, i_N}|^2)^{1/2}$. The ℓ_1 -norm of \mathcal{X} is calculated as $\|\mathcal{X}\|_1 = \sum_{i_1, \dots, i_N} |x_{i_1, \dots, i_N}|$. The nuclear norm of a matrix X is defined as the sum of its singular values, i.e., $\|X\|_* = \sum_i \sigma_i$, where σ_i denotes the i th singular value of X . Furthermore, the weighted nuclear norm of matrix X is $\|X\|_{w,*} = \sum_i w_i \sigma_i$. And we use $\langle \cdot, \cdot \rangle$ to present the inner product.

In the real world, HSIs are easily destroyed by various mixed noise, i.e., Gaussian noise, impulse noise, stripes, deadlines, and other noise [42, 56]. All of these noise are generalized into two categories, Gaussian noise and sparse noise. Combining these information, the HSI noise degradation model is generated as:

$$\mathcal{Y} = \mathcal{X} + \mathcal{N} + \mathcal{S}, \quad (1)$$

where all variables have the same size of $m \times n \times p$, where $m \times n$ is the spatial size of each band, and p denotes the number of bands. \mathcal{Y} is the corrupted noisy HSI, \mathcal{X} presents the latent clean HSI, \mathcal{N} denotes the Gaussian noise, and \mathcal{S} is sparse noise containing impulse noise, stripes, and deadlines. The goal of HSI denoising is to obtain the clean image \mathcal{X} from the noisy image \mathcal{Y} .

2.2. TV-based HSI denoising. Recall that Rudin *et al.* introduce the TV in [37], and its denoising capability is found effective. TV digs useful spatial and spectral information from HSIs, which is very valid for denoising. Combining with a low-rank constraint, the TV low-rank model properly utilizes the prior information of HSIs in spatial and spectral domain and achieves superb denoising performance. The general TV denoising model for HSIs is as follows

$$\begin{aligned} & \min_{\mathcal{X}, \mathcal{S}} \mathcal{F}(\mathcal{X}) + \lambda_1 \|\mathcal{S}\|_1 + \lambda_2 \|\mathcal{X}\|_{TV} \\ \text{s.t. } & \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2 \leq \varepsilon_0, \end{aligned} \quad (2)$$

where λ_1 and λ_2 are positive regularized parameters, and ε_0 is the Gaussian noise density variance, $\mathcal{F}(\mathcal{X})$ is the low-rank term of the HSI, which has various types but all focusing on the important low-rank property of the clean HSI, and $\|\mathcal{X}\|_{TV}$ denotes the TV regularization of the HSI.

Developing from the original band to band TV, there are some variants, such as spatial-spectral TV, which exploits spectral correlation in HSIs [1, 55]. A widely used one is the combination of the anisotropic TV regularization and the low-rank constraint. Existing TV-based methods remove mixed noise efficiently, however, the various uses of sparse information will make denoising results far different. Besides, dealing with the balance between TV and low-rank term is still a problem. Edges and important details reservation are suitable indexes to measure the effectiveness of denoising results. We will introduce the proposed method in dealing with above points later in Section 3.

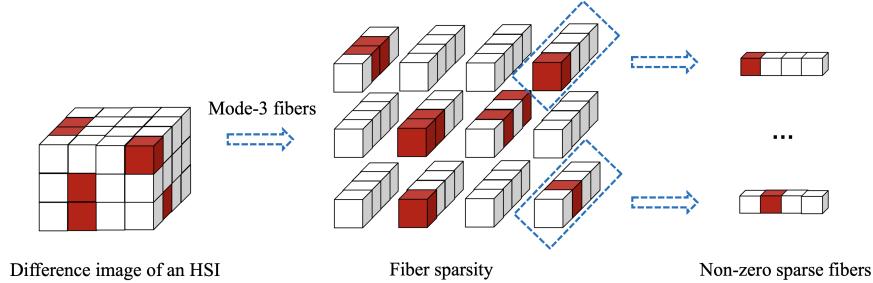


FIGURE 1. The illustration of double sparsity in HSIs in the gradient domain. The values of red parts are non-zero and white parts are zero. It is clear to see the fiber sparsity in this figure, and we choose two typical non-zero fibers to show their sparsity. Fiber sparsity means that only a few fibers are non-zero in the difference image.

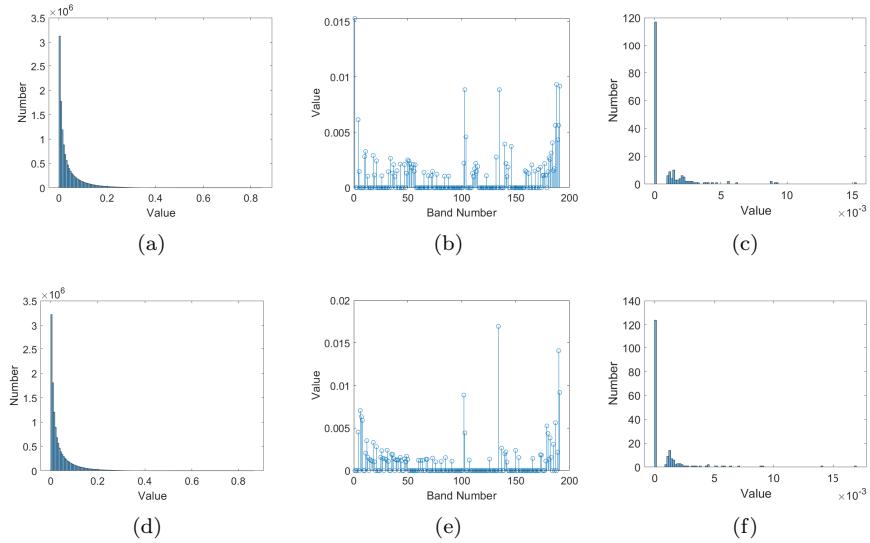


FIGURE 2. Double sparsity analysis of the Washington DC Mall data set. The first column denotes the histogram of horizontal difference image and vertical difference image, respectively. The second column presents the stem distribution of the 95th fiber in horizontal difference image and the 103rd fiber in vertical difference image, respectively. The last column shows the histogram of the 95th fiber in horizontal difference image and the 103rd fiber in vertical difference image, respectively.

3. Double sparsity-regularized TV and low-rank representation model.

3.1. Proposed double sparsity TV model. Owing to all the discussion about TV-based denoising algorithms, we now introduce the motivation of the proposed

model. The first to be concerned is the TV regularization, which is usually employed to explore the sparsity of difference images. However, TV makes final denoising results too smooth if the sparse prior is over characterized. LRTDGS [4] applies the weighted $\ell_{2,1}$ -norm to characterize group sparsity and successfully enhances the denoising results compared to other TV-based algorithms. Moreover, we further explore the sparsity of difference images and note another important characteristic on spectral domain, that is, most non-zero fibers are sparse in the gradient domain.

Based on the above discussion, we know that an HSI has double sparsity in the gradient domain as shown in Fig. 1, i.e., fiber sparsity with sparse fibers. First there exist a lot of zero fibers in the difference images of the HSI, what's more, the remaining non-zero fibers are sparse. The details to demonstrate double sparsity will be shown in Fig. 2. As shown in Fig. 2(a) and Fig. 2(d), difference images of the Washington DC Mall data set have a global sparse feature. The other notable point is that there exist a lot of non-zero fibers in difference images, and these fibers are sparse because a large amount of values in them are zero or tend to be zero. In addition, most of the non-zero values are extremely small in these fibers. After processing these small values (we set the value less than 10^{-3} to 0), the sparsity of fibers is more clear. In the middle and the right columns in Fig. 2, we show the sparsity of two typical fibers in horizontal and vertical difference images, respectively. There are amounts of non-zero fibers having the same sparsity like these two fibers. Above phenomenon proves that the difference images have a strong fiber sparsity property. Considering the above characteristics, we try to integrate fiber sparsity with sparse fibers and get the double sparsity model. We propose a weighted double sparsity regularization term as

$$\|W \odot \mathcal{D}\mathcal{X}\|_{1,1} = \sum_{i=1}^m \sum_{j=1}^n W_h(i,j) \|\mathcal{D}_h\mathcal{X}(i,j,:)\|_1 + \sum_{i=1}^m \sum_{j=1}^n W_v(i,j) \|\mathcal{D}_v\mathcal{X}(i,j,:)\|_1, \quad (3)$$

where $W = \begin{pmatrix} W_h \\ W_v \end{pmatrix}$ is a nonnegative weighting matrix to promote double sparsity that will be automatically update and introduced later in the optimization process, and \mathcal{D} is the differential operator, which has horizontal and vertical dimensions, i.e., \mathcal{D}_h and \mathcal{D}_v . They are presented as

$$\begin{cases} \mathcal{D}_h\mathcal{X}(i,j,k) = x_{i,j+1,k} - x_{i,j,k} \\ \mathcal{D}_v\mathcal{X}(i,j,k) = x_{i+1,j,k} - x_{i,j,k}. \end{cases}$$

Next, we consider about the low-rank part of our model. In [15], the weighted nuclear norm minimization (WNNM) problem has been studied and introduced into HSI denoising. Many researches and applications have demonstrated its performance in describing the low-rank prior of HSIs, e.g., [17, 21, 22, 43, 47–49, 51], and further improvement can be found, e.g., in [6, 27]. As shown in Fig. 3, the mode-3 unfoldings of the HSIs show the most prominent low-rank attributes. According to above analysis, the mode-3 unfolding of an HSI combined with the weighted nuclear norm can properly characterize the low-rank property of the clean HSI and utilize spectral information. That's the reason why we choose the following low-rank regularization

$$\|X_{(3)}\|_{w,*}.$$

As aforementioned, we notice that the mode-3 unfolding of the original HSI has a stronger low-rank property and try to describe it by the weighted nuclear

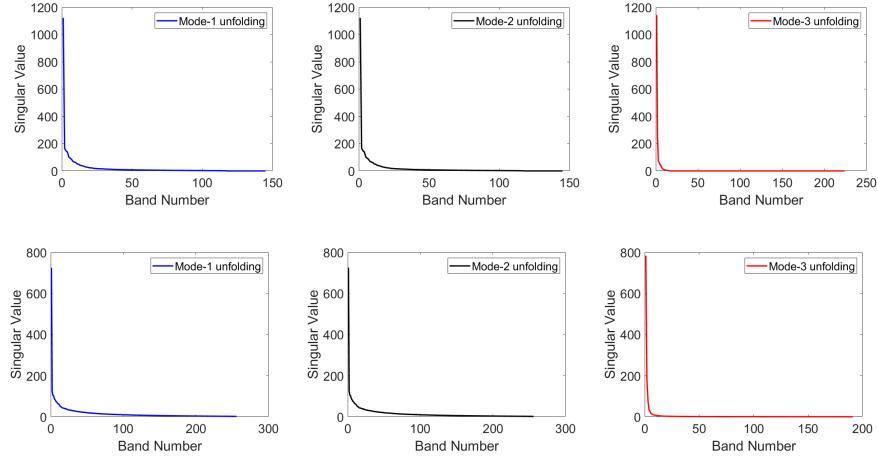


FIGURE 3. Low-rank property analysis of two simulated data sets. The top row denotes the unfoldings of the simulated Indian Pines data. The bottom row is the unfoldings of the Washington DC Mall data.

norm. What's more, another significant term — TV regularization is formed as equation (3). This TV term efficiently utilizes the sparsity of spectral mode of the HSI, i.e., the double sparsity that fuses the fiber sparsity and sparse fibers. Regular TV-based algorithms usually smooth some details due to the overuse of sparsity on difference images. Thus, combining with the weighted nuclear norm, we propose a weighted double sparsity-regularized low-rank representation (LRDSTV) model for HSI denoising as follows

$$\min_{\mathcal{X}, \mathcal{S}} \frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2 + \lambda_1 \|\mathcal{S}\|_1 + \lambda_2 \|W \odot \mathcal{D}\mathcal{X}\|_{1,1} + \lambda_3 \|X_{(3)}\|_{w,*}. \quad (4)$$

In the above proposed model, Gaussian noise, sparse noise, the low-rank property of HSIs, and the piecewise smooth structure are all concerned. The Frobenius term is efficacious in elimination of the Gaussian noise. The next ℓ_1 -norm captures the sparse noise, i.e., the impulse noise, stripes and deadlines. Furthermore, the weighted ℓ_1 -norm exploits double sparsity of difference images in spectral domain and helps to promote piecewise smooth structure. The final low-rank term takes advantage of original HSI information and tries to obtain more details, which can avoid denoising results being over smoothed. The specific optimization of the proposed model will be introduced in the following section.

3.2. Optimization process. In the next content, we will introduce two algorithms of the proposed model in (4). They have a small difference in the number of auxiliary variables and calculation order of these variables.

3.2.1. LRDSTV₁ optimization. The above model (4) is transformed to an equivalent problem by introducing four auxiliary variables \mathcal{U} , \mathcal{V} , \mathcal{P} , \mathcal{S}_1 as follows

$$\begin{aligned} & \min_{\mathcal{X}, \mathcal{U}, \mathcal{V}, \mathcal{P}, \mathcal{S}, \mathcal{S}_1} \frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2 + \lambda_1 \|\mathcal{S}_1\|_1 + \lambda_2 \|W \odot \mathcal{V}\|_{1,1} + \lambda_3 \|P_{(3)}\|_{w,*} \\ & \text{s.t. } \mathcal{X} = \mathcal{U}, \mathcal{D}\mathcal{U} = \mathcal{V}, \mathcal{X} = \mathcal{P}, \mathcal{S} = \mathcal{S}_1. \end{aligned} \quad (5)$$

Then the nonconvex ADMM in [45] is applied to optimize equation (5). Define

$$\mathcal{A} = \begin{pmatrix} \mathcal{I} & -\mathcal{I} & 0 \\ 0 & \mathcal{D} & 0 \\ \mathcal{I} & 0 & 0 \\ 0 & 0 & \mathcal{I} \end{pmatrix}, \mathcal{B} = \begin{pmatrix} 0 & 0 & 0 \\ -\mathcal{I} & 0 & 0 \\ 0 & -\mathcal{I} & 0 \\ 0 & 0 & -\mathcal{I} \end{pmatrix}, \hat{\mathcal{X}} = \begin{pmatrix} \mathcal{X} \\ \mathcal{U} \\ \mathcal{S} \end{pmatrix}, \mathcal{Z} = \begin{pmatrix} \mathcal{V} \\ \mathcal{P} \\ \mathcal{S}_1 \end{pmatrix},$$

and

$$f(\hat{\mathcal{X}}) = \frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2, \\ g(\mathcal{Z}) = \lambda_1 \|\mathcal{S}_1\|_1 + \lambda_2 \|W \odot \mathcal{V}\|_{1,1} + \lambda_3 \|P_{(3)}\|_{w,*},$$

Then, we reformulate the model (5) into the following form

$$\begin{aligned} & \min_{\hat{\mathcal{X}}, \mathcal{Z}} f(\hat{\mathcal{X}}) + g(\mathcal{Z}) \\ & \text{s.t. } \mathcal{A}\hat{\mathcal{X}} + \mathcal{B}\mathcal{Z} = 0. \end{aligned} \quad (6)$$

And the augmented Lagrangian function \mathcal{L}_β is shown

$$\mathcal{L}_\beta(\hat{\mathcal{X}}, \mathcal{Z}, \mathcal{T}) = f(\hat{\mathcal{X}}) + g(\mathcal{Z}) + \langle \mathcal{T}, \mathcal{A}\hat{\mathcal{X}} + \mathcal{B}\mathcal{Z} \rangle + \frac{\beta}{2} \|\mathcal{A}\hat{\mathcal{X}} + \mathcal{B}\mathcal{Z}\|_F^2, \quad (7)$$

where $\beta > 0$ and \mathcal{T} denotes the multipliers, i.e., $\mathcal{T} = \begin{pmatrix} \mathcal{T}_1 \\ \mathcal{T}_2 \\ \mathcal{T}_3 \\ \mathcal{T}_4 \end{pmatrix}$. Then, combining ADMM with a reweighting strategy and we get the following framework

$$\begin{cases} \hat{\mathcal{X}}^{k+1} = \arg \min_{\hat{\mathcal{X}}} \mathcal{L}_\beta(\hat{\mathcal{X}}, \mathcal{Z}^k, \mathcal{T}^k) \\ \mathcal{Z}^{k+1} = \arg \min_{\mathcal{Z}} \mathcal{L}_\beta(\hat{\mathcal{X}}^{k+1}, \mathcal{Z}, \mathcal{T}^k) \\ \mathcal{T}^{k+1} = \mathcal{T}^k + \beta(\mathcal{A}\hat{\mathcal{X}}^{k+1} + \mathcal{B}\mathcal{Z}^{k+1}). \end{cases} \quad (8)$$

where the update details of W and w will be shown later. Additionally, the augmented Lagrangian function of (24) is shown as below

$$\begin{aligned} & \mathcal{L}_\beta(\mathcal{X}, \mathcal{U}, \mathcal{S}, \mathcal{V}, \mathcal{P}, \mathcal{S}_1, \mathcal{T}) \\ &= \frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2 + \lambda_1 \|\mathcal{S}_1\|_1 + \lambda_2 \|W \odot \mathcal{V}\|_{1,1} + \lambda_3 \|P_{(3)}\|_{w,*} + \frac{\beta}{2} \|\mathcal{X} - \mathcal{U}\|_F^2 \\ &+ \langle \mathcal{T}_1, \mathcal{X} - \mathcal{U} \rangle + \frac{\beta}{2} \|\mathcal{D}\mathcal{U} - \mathcal{V}\|_F^2 + \langle \mathcal{T}_2, \mathcal{D}\mathcal{U} - \mathcal{V} \rangle + \frac{\beta}{2} \|\mathcal{X} - \mathcal{P}\|_F^2 \\ &+ \langle \mathcal{T}_3, \mathcal{X} - \mathcal{P} \rangle + \frac{\beta}{2} \|\mathcal{S} - \mathcal{S}_1\|_F^2 + \langle \mathcal{T}_4, \mathcal{S} - \mathcal{S}_1 \rangle. \end{aligned} \quad (9)$$

The equation (9) will be transformed into several subproblems. To solve the $\hat{\mathcal{X}}$ subproblem, we update each of the primal variables \mathcal{X} , \mathcal{U} , and \mathcal{S} with the others fixed, and then the \mathcal{Z} subproblem has been decoupled solved, following multipliers \mathcal{T} updates. All showed up superscript $k+1$ means the $(k+1)$ th iteration.

- $\hat{\mathcal{X}}$ subproblem

1) By collecting all parts related to \mathcal{X} in equation (9), we get the following optimization

$$\mathcal{X}^{k+1} = \arg \min_{\mathcal{X}} \frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}^k\|_F^2 + \frac{\beta}{2} \|\mathcal{X} - \mathcal{U}^k + \frac{1}{\beta} \mathcal{T}_1^k\|_F^2 + \frac{\beta}{2} \|\mathcal{X} - \mathcal{P}^k + \frac{1}{\beta} \mathcal{T}_3^k\|_F^2. \quad (10)$$

And its closed-form solution is

$$\mathcal{X}^{k+1} = \frac{1}{2\beta + 1}(\mathcal{Y} - \mathcal{S}^k + \beta(\mathcal{U}^k + \mathcal{P}^k) - (\mathcal{T}_1^k + \mathcal{T}_3^k)). \quad (11)$$

2) After processing, the \mathcal{U} -subproblem is optimized in the following

$$\mathcal{U}^{k+1} = \arg \min_{\mathcal{U}} \frac{1}{2} \|\mathcal{X}^{k+1} - \mathcal{U} + \frac{1}{\beta} \mathcal{T}_1^k\|_F^2 + \frac{1}{2} \|\mathcal{D}\mathcal{U} - \mathcal{V}^k + \frac{1}{\beta} \mathcal{T}_2^k\|_F^2, \quad (12)$$

which is a least-squares problem, solving that and we obtain

$$\mathcal{U}^{k+1} = (\mathcal{D}^T \mathcal{D} + \mathcal{I})^{-1}(\mathcal{X}^{k+1} + \frac{1}{\beta} \mathcal{T}_1^k + \mathcal{D}^T(\mathcal{V}^k - \frac{1}{\beta} \mathcal{T}_2^k)). \quad (13)$$

3) Gathering all \mathcal{S} terms, we get

$$\mathcal{S}^{k+1} = \arg \min_{\mathcal{S}} \frac{1}{2} \|\mathcal{Y} - \mathcal{X}^{k+1} - \mathcal{S}\|_F^2 + \frac{\beta}{2} \|\mathcal{S} - \mathcal{S}_1^k + \frac{1}{\beta} \mathcal{T}_4^k\|_F^2.$$

And \mathcal{S} is updated by

$$\mathcal{S}^{k+1} = \frac{1}{1 + \beta}(\mathcal{Y} - \mathcal{X}^{k+1} + \beta \mathcal{S}_1^k - \mathcal{T}_4^k). \quad (14)$$

- \mathcal{Z} subproblem

4) According to above analysis, \mathcal{V} -subproblem is

$$\mathcal{V}^{k+1} = \arg \min_{\mathcal{V}} \lambda_2 \|W^{k+1} \odot \mathcal{V}\|_{1,1} + \frac{\beta}{2} \|\mathcal{D}\mathcal{U}^{k+1} - \mathcal{V} + \frac{1}{\beta} \mathcal{T}_2^k\|_F^2.$$

This problem is solved by the soft-thresholding operator, which means that \mathcal{V} is updated by

$$\mathcal{V}^{k+1} = \text{soft}(\mathcal{D}\mathcal{U}^{k+1} + \frac{1}{\beta} \mathcal{T}_2^k, \frac{\lambda_2}{\beta} W^{k+1}), \quad (15)$$

where

$$\text{soft}(x, \tau) = \text{sgn}(x) \max\{|x| - \tau, 0\}, \quad (16)$$

for $x \in \mathbb{R}$ and $\tau > 0$. Similarly as in [21], the weighting matrix W^{k+1} is updated as follows to promote the sparsity of \mathcal{V}

$$W^{k+1}(i, j) = \frac{1}{\|(\mathcal{D}\mathcal{U}^{k+1} + \frac{1}{\beta} \mathcal{T}_2^k)(i, j, :) \|_1 + \epsilon}, \quad (17)$$

where ϵ is a very small number to avoid the singularity. Then, we calculate each fiber separately

$$\mathcal{V}^{k+1}(i, j, :) = \text{soft}((\mathcal{D}\mathcal{U}^{k+1} + \frac{1}{\beta} \mathcal{T}_2^k)(i, j, :), \frac{\lambda_2}{\beta} W^{k+1}(i, j)).$$

5) Extracting \mathcal{P} terms in the augmented Lagrangian function \mathcal{L}_β , we get \mathcal{P} -subproblem as

$$\mathcal{P}^{k+1} = \arg \min_{\mathcal{P}} \lambda_3 \|P_{(3)}\|_{w^{k+1}, *} + \frac{\beta}{2} \|X_{(3)}^{k+1} - P_{(3)} + \frac{1}{\beta} T_{3(3)}^k\|_F^2.$$

To solve this problem, we first give some definitions. Let $X = U\Sigma V^T$ be the singular value decomposition (SVD) of X , and Σ is a diagonal matrix with σ_i being the i th singular value of X . Define the weighted singular value thresholding operator $SVT_{w,\beta}$ on X as

$$SVT_{w,\beta}(X) = US_{w,\beta}(\Sigma)V^T,$$

where $S_{w,\beta}$ is the soft-thresholding operator [26] with a weighting vector w , which is denoted

$$(S_{w,\beta}(\Sigma))(i,i) = \max(\sigma_i - \beta w_i, 0).$$

Thus, \mathcal{P} is updated as

$$P_{(3)}^{k+1} = SVT_{w^{k+1}, \frac{\lambda_3}{\beta}}(X_{(3)}^{k+1} + \frac{1}{\beta}T_{(3)}^k), \quad (18)$$

where each element of w^{k+1} is defined as $w_i^{k+1} = \frac{1}{\sigma_i^{k+1} + \epsilon}$, and σ_i^{k+1} is the i th singular value of $X_{(3)}^{k+1} + \frac{1}{\beta}T_{(3)}^k$.

6) Gathering all \mathcal{S}_1 terms we have

$$\mathcal{S}_1^{k+1} = \arg \min_{\mathcal{S}_1} \lambda_1 \|\mathcal{S}_1\|_1 + \frac{\beta}{2} \|\mathcal{S}^{k+1} - \mathcal{S}_1 + \frac{1}{\beta} \mathcal{T}_4^k\|_F^2.$$

Using the soft-thresholding in (16) and \mathcal{S}_1 is updated by

$$\mathcal{S}_1^{k+1} = \text{soft}(\mathcal{S}^{k+1} + \frac{1}{\beta} \mathcal{T}_4^k, \frac{\lambda_1}{\beta}). \quad (19)$$

• Update \mathcal{T}

7) The update rules of multipliers \mathcal{T}_1 , \mathcal{T}_2 , \mathcal{T}_3 , and \mathcal{T}_4 are

$$\begin{cases} \mathcal{T}_1^{k+1} = \mathcal{T}_1^k + \beta(\mathcal{X}^{k+1} - \mathcal{U}^{k+1}) \\ \mathcal{T}_2^{k+1} = \mathcal{T}_2^k + \beta(\mathcal{D}\mathcal{U}^{k+1} - \mathcal{V}^{k+1}) \\ \mathcal{T}_3^{k+1} = \mathcal{T}_3^k + \beta(\mathcal{X}^{k+1} - \mathcal{P}^{k+1}) \\ \mathcal{T}_4^{k+1} = \mathcal{T}_4^k + \beta(\mathcal{S}^{k+1} - \mathcal{S}_1^{k+1}). \end{cases} \quad (20)$$

Algorithm 1: LRDSTV₁

Input: Noisy image \mathcal{Y} , stopping criterion ε , parameters λ_1 , λ_2 , λ_3 , and β

Output: The restored HSI \mathcal{X}

Initialize: $\mathcal{U} = \mathcal{S} = \mathcal{V} = \mathcal{P} = \mathcal{S}_1 = 0$, $\mathcal{T} = 0$, $\beta = 4$

While not converge **do**

- 1: Update \mathcal{X} via (11).
 - 2: Update \mathcal{U} via (13).
 - 3: Update \mathcal{S} via (14).
 - 4: Update \mathcal{V} via (15).
 - 5: Update \mathcal{P} via (18).
 - 6: Update \mathcal{S}_1 via (19).
 - 7: Update all Lagrange multipliers via (20).
- Check the convergence condition: $\frac{\|\mathcal{X}^{k+1} - \mathcal{X}^k\|_F^2}{\|\mathcal{X}^k\|_F^2} \leq \varepsilon$.

End While.

The above procedure solves the proposed optimization problem and the resulting algorithm is named as LRDSTV₁. Algorithm 1 presents every step of the whole process. Inputs of LRDSTV₁ are: the original noisy HSI $\mathcal{Y} \in \mathbb{R}^{m \times n \times p}$, regularization parameters λ_1 , λ_2 , λ_3 , $\beta = 4$ and the stop criterion ε , which was set to 10^{-4} in the whole algorithm. More details about parameter choosing will be discussed in Section 4. Before processing the noisy image, we initialize variables $\mathcal{U} = \mathcal{S} = \mathcal{V} = \mathcal{P} = \mathcal{S}_1 = 0$, and Lagrange multipliers are also started at 0.

3.2.2. *LRDSTV₂ optimization.* The model of the second algorithm has fewer variables to be calculated. Introducing three auxiliary variables \mathcal{U} , \mathcal{V} , \mathcal{P} and we get the second term of our model as follows

$$\begin{aligned} & \min_{\mathcal{X}, \mathcal{Y}, \mathcal{U}, \mathcal{P}, \mathcal{S}} \frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2 + \lambda_1 \|\mathcal{S}\|_1 + \lambda_2 \|W \odot \mathcal{V}\|_{1,1} + \lambda_3 \|P_{(3)}\|_{w,*} \\ & \text{s.t. } \mathcal{X} = \mathcal{U}, \mathcal{D}\mathcal{U} = \mathcal{V}, \mathcal{X} = \mathcal{P}. \end{aligned} \quad (21)$$

Define

$$\tilde{\mathcal{A}} = \begin{pmatrix} \mathcal{I} & 0 & 0 \\ 0 & -\mathcal{I} & 0 \\ \mathcal{I} & 0 & 0 \end{pmatrix}, \tilde{\mathcal{B}} = \begin{pmatrix} -\mathcal{I} & 0 \\ \mathcal{D} & 0 \\ 0 & -\mathcal{I} \end{pmatrix}, \tilde{\mathcal{X}} = \begin{pmatrix} \mathcal{X} \\ \mathcal{V} \\ \mathcal{S} \end{pmatrix}, \tilde{\mathcal{Z}} = \begin{pmatrix} \mathcal{U} \\ \mathcal{P} \end{pmatrix},$$

and

$$\begin{aligned} \tilde{f}(\tilde{\mathcal{X}}) &= \frac{1}{2} \|\mathcal{Y} - \mathcal{X} - \mathcal{S}\|_F^2 + \lambda_1 \|\mathcal{S}\|_1 + \lambda_2 \|W \odot \mathcal{V}\|_{1,1}, \\ \tilde{g}(\tilde{\mathcal{Z}}) &= \lambda_3 \|P_{(3)}\|_{w,*}. \end{aligned}$$

Then, we reformulate the model (21) into the following form

$$\begin{aligned} & \min_{\tilde{\mathcal{X}}, \tilde{\mathcal{Z}}} \tilde{f}(\tilde{\mathcal{X}}) + \tilde{g}(\tilde{\mathcal{Z}}) \\ & \text{s.t. } \tilde{\mathcal{A}}\tilde{\mathcal{X}} + \tilde{\mathcal{B}}\tilde{\mathcal{Z}} = 0. \end{aligned} \quad (22)$$

And the augmented Lagrangian function $\tilde{\mathcal{L}}_\beta$ is shown

$$\tilde{\mathcal{L}}_\beta(\tilde{\mathcal{X}}, \tilde{\mathcal{Z}}, \tilde{\mathcal{T}}) = \tilde{f}(\tilde{\mathcal{X}}) + \tilde{g}(\tilde{\mathcal{Z}}) + \langle \tilde{\mathcal{T}}, \tilde{\mathcal{A}}\tilde{\mathcal{X}} + \tilde{\mathcal{B}}\tilde{\mathcal{Z}} \rangle + \frac{\beta}{2} \|\tilde{\mathcal{A}}\tilde{\mathcal{X}} + \tilde{\mathcal{B}}\tilde{\mathcal{Z}}\|_F^2, \quad (23)$$

where $\beta > 0$ and $\tilde{\mathcal{T}}$ denotes the multipliers, i.e., $\tilde{\mathcal{T}} = \begin{pmatrix} \tilde{\mathcal{T}}_1 \\ \tilde{\mathcal{T}}_2 \\ \tilde{\mathcal{T}}_3 \end{pmatrix}$ and the weighting

coefficients W and w will be updated by a reweighting strategy. Then according to the nonconvex ADMM, we get the following framework

$$\begin{cases} \tilde{\mathcal{X}}^{k+1} = \arg \min_{\tilde{\mathcal{X}}} \tilde{\mathcal{L}}_\beta(\tilde{\mathcal{X}}, \tilde{\mathcal{Z}}^k, \tilde{\mathcal{T}}^k) \\ \tilde{\mathcal{Z}}^{k+1} = \arg \min_{\tilde{\mathcal{Z}}} \tilde{\mathcal{L}}_\beta(\tilde{\mathcal{X}}^{k+1}, \tilde{\mathcal{Z}}, \tilde{\mathcal{T}}^k) \\ \tilde{\mathcal{T}}^{k+1} = \tilde{\mathcal{T}}^k + \beta(\tilde{\mathcal{A}}\tilde{\mathcal{X}}^{k+1} + \tilde{\mathcal{B}}\tilde{\mathcal{Z}}^{k+1}). \end{cases} \quad (24)$$

Let $\mathcal{W} = -\tilde{\mathcal{T}}/\beta$. We solve each subproblem in the following. Notice that the process of solving subproblems is similar to LRDSTV₁, we therefore omit the details.

• $\tilde{\mathcal{X}}$ subproblem

1) \mathcal{X} -subproblem

$$\mathcal{X}^{k+1} = \frac{1}{2\beta + 1} (\mathcal{Y} - \mathcal{S}^k + \beta(\mathcal{U}^k + \mathcal{W}_1^k + \mathcal{P}^k + \mathcal{W}_3^k)). \quad (25)$$

2) \mathcal{V} -subproblem

$$\mathcal{V}^{k+1} = \text{soft}((\mathcal{D}\mathcal{U}^k - \mathcal{W}_2^k), \frac{\lambda_2}{\beta} \mathcal{W}^{k+1}). \quad (26)$$

Similarly, we calculate each fiber separately:

$$\mathcal{W}^{k+1}(i, j, :) = \text{soft}((\mathcal{D}\mathcal{U}^k - \mathcal{W}_2^k)(i, j, :), \frac{\lambda_2}{\beta} \mathcal{W}^{k+1}(i, j)),$$

where the weighting matrix W^{k+1} is updated as

$$W^{k+1}(i, j) = \frac{1}{\|(\mathcal{D}\mathcal{U}^k - \mathcal{W}_2^k)(i, j, :) \|_1 + \epsilon}.$$

3) \mathcal{S} -subproblem

$$\mathcal{S}^{k+1} = \text{soft}(\mathcal{Y} - \mathcal{X}^{k+1}, \lambda_1). \quad (27)$$

- $\tilde{\mathcal{Z}}$ subproblem

4) \mathcal{U} -subproblem

$$\mathcal{U}^{k+1} = (\mathcal{D}^T \mathcal{D} + \mathcal{I})^{-1}(\mathcal{X}^{k+1} - \mathcal{W}_1^k + \mathcal{D}^T(\mathcal{V}^{k+1} + \mathcal{W}_2^k)). \quad (28)$$

5) \mathcal{P} -subproblem

$$P_{(3)}^{k+1} = \text{SVT}_{w^{k+1}, \frac{\lambda_3}{\beta}}(X_{(3)}^{k+1} - W_{3(3)}^k), \quad (29)$$

where $w_i^{k+1} = \frac{1}{\sigma_i^{k+1} + \epsilon}$, and σ_i^{k+1} is the i th singular value of $X_{(3)}^{k+1} - W_{3(3)}^k$.

- Update \mathcal{W}

6) The update rules of multipliers \mathcal{W}_1 , \mathcal{W}_2 , and \mathcal{W}_3 are

$$\begin{cases} \mathcal{W}_1^{k+1} = \mathcal{W}_1^k - (\mathcal{X}^{k+1} - \mathcal{U}^{k+1}) \\ \mathcal{W}_2^{k+1} = \mathcal{W}_2^k - (\mathcal{D}\mathcal{U}^{k+1} - \mathcal{V}^{k+1}) \\ \mathcal{W}_3^{k+1} = \mathcal{W}_3^k - (\mathcal{X}^{k+1} - \mathcal{P}^{k+1}). \end{cases} \quad (30)$$

We have the same strategy of setting initial values as in Algorithm 1. And the above process, summarized in Algorithm 2 below, is our second algorithm, which is named as LRDSTV₂. It is clear that LRDSTV₂ has one less subproblem and one less submultiplier than LRDSTV₁.

Algorithm 2: LRDSTV₂

Input: Noisy image \mathcal{Y} , stopping criterion ε , parameters λ_1 , λ_2 , λ_3 , and β

Output: The restored HSI \mathcal{X}

Initialize: $\mathcal{V} = \mathcal{U} = \mathcal{P} = \mathcal{S} = 0$, $\beta = 4$, $\mathcal{W}_1 = \mathcal{W}_2 = \mathcal{W}_3 = 0$

While not converge **do**

 1: Update \mathcal{X} via (25).

 2: Update \mathcal{V} via (26).

 3: Update \mathcal{S} via (27).

 4: Update \mathcal{U} via (28).

 5: Update \mathcal{P} via (29).

 6: Update all Lagrange multipliers via (30).

 Check the convergence condition: $\frac{\|\mathcal{X}^{k+1} - \mathcal{X}^k\|_F^2}{\|\mathcal{X}^k\|_F^2} \leq \varepsilon$.

End While.

4. Experimental results and discussion. In this section, we present results of experiments on both simulated and real data sets of the proposed algorithms. To further prove the extraordinary effect of the double sparsity structure combined with the low-rank term, five more popular algorithms are compared in all

experiments. Most of them obtain impressive results, i.e., LRMR [56], the total-variation-regularized matrix factorization (LRTV) [20], the spatial spectral total-variation (SSTV)¹ [1], the total-variation combined with low-rank tensor decomposition (LRTDTV)² [44], and LRTDGS³ [4].

All showed up parameters are manually adjusted according to the instructions in these papers to get the optimum. Otherwise, pixel values in all bands are normalized to [0, 1]. The following experiments are all calculated in MATLAB R2020a in a laptop with 16G RAM.

4.1. Simulated experiments. We choose two clean HSI data sets in simulated experiments to demonstrate the effectiveness of the proposed two algorithms. One is the simulated Indian Pines data set⁴ with the size of $145 \times 145 \times 224$. The other is the Hyperspectral Digital Imagery Collection Experiment (HYDICE) Washington DC Mall data with 256×256 in spatial domain and has 191 bands. We use various mixed noise in three different cases to verify the performance of above competitive algorithms. The specific details of each case are as follows:

Case 1. The zero-mean Gaussian noise are added to every band, and we separate this case into four subcases with noise variances of 0.025, 0.05, 0.075, and 0.1.

Case 2. The noise variance of zero-mean Gaussian noise in each band is different, which is randomly selected in the range of [0, 0.2].

Case 3. Gaussian noise, impulse noise, deadline noise, and stripe noise are all added to the clean data. Gaussian noise is just added as in case 2. Besides, we choose to add different percentages of impulse noise in each band and the percentage is randomly selected in the range of [0, 0.2]. Furthermore, deadlines and stripe noise with number ranging from 3 to 10 are added randomly to 20% of all bands in both data sets. We will compare the denoising results of the above algorithms from the following aspects.

4.1.1. Visual comparison. For the simulated Indian Pines data set, case 3 is chosen to demonstrate the difference among all above algorithms, and a typical band is involved in to show the restoration effect. As shown in Fig. 4, band 12 is extracted for the denoising results comparison. For better observation, we also enlarge some areas in the results. It is clear to see that LRMR and SSTV are hard to eliminate all noise, there still exists some noisy parts. LRTV, LRTDTV, LRTDGS, and LRDSTV remove all the noise in image. However, we find difference in details which are mainly shown as the edge difference. LRTV, LRTDTV, and LRTDGS are difficult to preserve all the edges and details as well as the proposed algorithm. In the denoising results of LRTV and LRTDTV, some similar parts of the simulated Indian Pines data set are more easily to be seen as a whole part rather than separate clearly. Comparing with the algorithms mentioned above, denoising results of LRTDGS reflect that most of the edge information is well preserved, and only a very small amount of edges are still blurred. However, paint-like traces appear to be attached to the surface of some parts as shown in the denoising image. And both in Figs. 4(h) and 4(i), we notice that edges and details are preserved well as that in original image, moreover, each part stay at where they suppose to be and do not mix together.

¹<https://sites.google.com/view/hkaggarwal/publications>

²<http://gr.xjtu.edu.cn/web/dymeng/>

³<https://sites.google.com/site/rshewei/home>

⁴<https://engineering.purdue.edu/biehl/MultiSpec/hyperspectral.html>

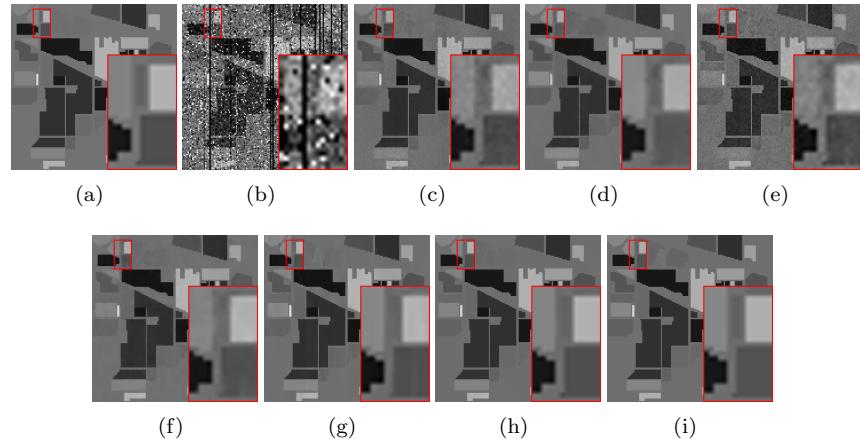


FIGURE 4. Denoising results of above algorithms under case 3 of the simulated Indian Pines data set, band 12 is extracted for comparison. (a) Original band. (b) Noisy band. (c) LRMR. (d) LRTV. (e) SSTV. (f) LRTDTV. (g) LRTDGS. (h) LRDSTV₁. (i) LRDSTV₂.

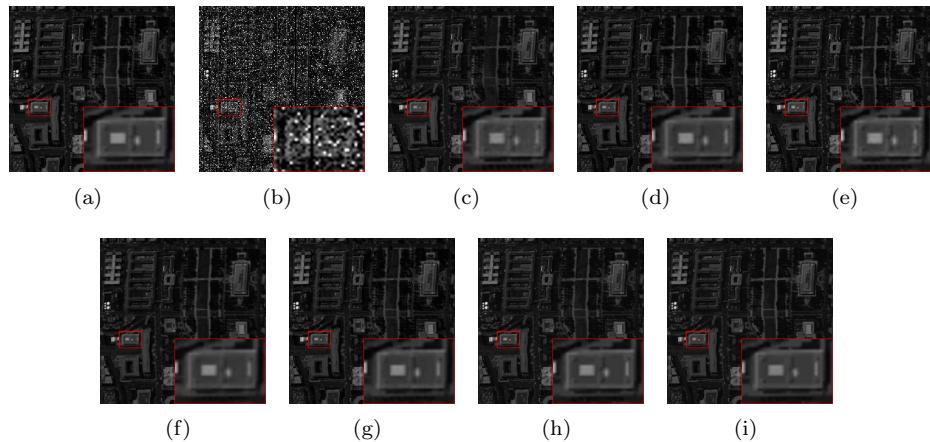


FIGURE 5. Denoising results of above algorithms under case 3 of the Washington DC Mall data set, band 27 is extracted for comparison. (a) Original band. (b) Noisy band. (c) LRMR. (d) LRTV. (e) SSTV. (f) LRTDTV. (g) LRTDGS. (h) LRDSTV₁. (i) LRDSTV₂.

Fig. 5 shows the denoising results of all above algorithms for the Washington DC Mall data set in case 3. LRMR and SSTV are failed in clearing all noise, so there still remains some noisy parts in denoising results as Figs. 5(c) and (e) show. The rest algorithms LRTV, LRTDTV, LRTDGS, and LRDSTV remove mixed noise and achieve better visual performance. Also from the enlarged boxes in Fig. 5 we figure out that comparing with other algorithms, our algorithms do better in preserving

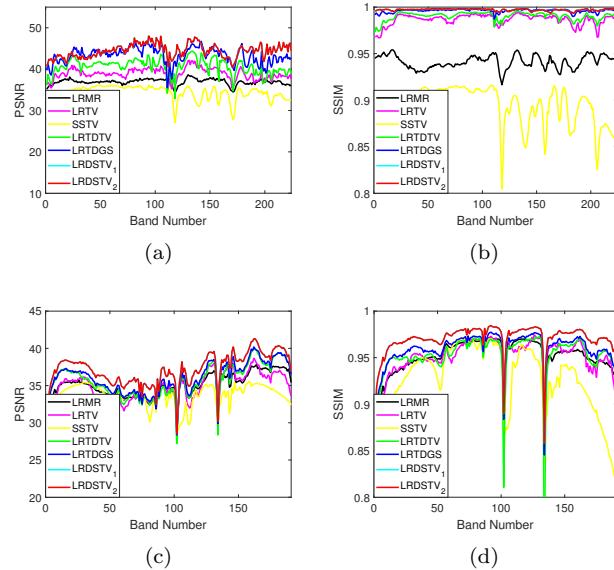


FIGURE 6. PSNR and SSIM values of each band for the fourth subcase of case 1 in two datasets: (top row) the simulated Indian Pines data set, and (bottom row) the Washington DC Mall data set.

We further compare the spectral signatures of all algorithms. Figs. 7 and 8 show the spectral signatures of pixel (5, 20) in the simulated Indian Pines data set and spectral signatures of pixel (140, 30) in the Washington DC Mall data set in case 3, respectively. From these two figures we get that LRMR, LRTV, and SSTV can not remove the noise efficiently, LRTDTV erases some details while denoising so its spectrum curve is smoother than others. The spectrum curves of LRTDGS and LRDSTV are both similar to original one, and there still exists some noise owing to the high noise intensity of these two cases, moreover, extra noise appears in few points of the LRTDGS and LRDSTV₁ curves. And the spectral signatures seem to be protected better in LRDSTV₂. In short, two algorithms of our model both maintain spectral information well in these two cases.

In addition, we select a 10×10 patch centering on the two chosen pixels and try to compare the spectral signatures difference of original simulated data sets and all seven denoising results on this whole patch, which will better prove the effectiveness of our algorithms. Specifically, the patch we selected is first reshaped to a matrix, for example, for the Washington DC Mall data set, the patch of size $10 \times 10 \times 191$ is reshaped to a 100×191 matrix, then we calculate the difference of the clean simulated patch and the denoising one to get the difference matrix. The sparsity of the difference matrix can be utilized to measure the similarity between denoising image and the original one. If the difference matrix is sparser, the image restoration results would be better. In the simulated Indian Pines data set, it is clear that Fig. 9(g) is sparser which means the denoising result of LRDSTV₂ preserving the best spectral signatures. What's more, even the results are not clear as that in the simulated Indian Pines data set, details tell the difference of above algorithms in the Washington DC Mall data set. The proposed two algorithms restore the

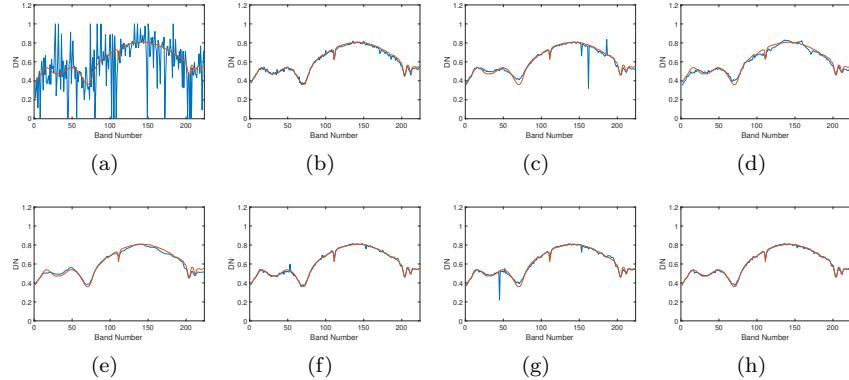


FIGURE 7. Spectrum comparison of pixel (5, 20) among different denoising algorithms under case 3 of the simulated Indian Pines data set. The orange and blue lines denote the original spectral signature and its estimation, respectively. (a) Noisy. (b) LRMR. (c) LRTV. (d) SSTV. (e) LRTDTV. (f) LRTDGS. (g) LRDSTV₁. (h) LRDSTV₂.

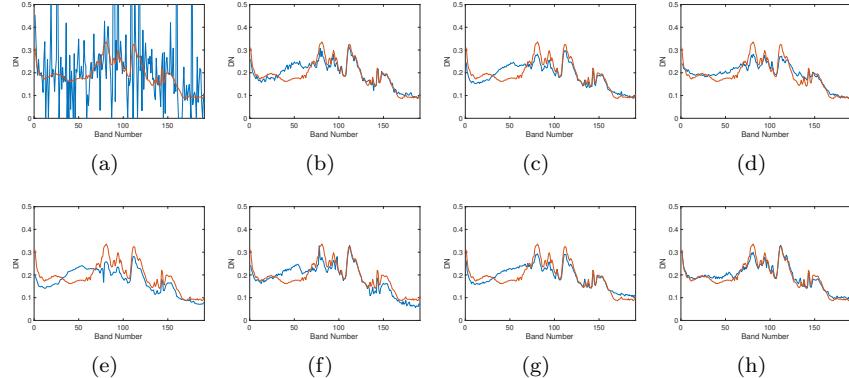


FIGURE 8. Spectrum comparison of pixel (140, 30) among different denoising algorithms under case 3 of the Washington DC Mall data set. The orange and blue lines denote the original spectral signature and its estimation, respectively. (a) Noisy. (b) LRMR. (c) LRTV. (d) SSTV. (e) LRTDTV. (f) LRTDGS. (g) LRDSTV₁. (h) LRDSTV₂.

patch spectral signature better than the rest algorithms as expected. In this part, we compare seven different denoising results in three aspects, which convincingly demonstrates the excellence of the proposed model.

4.2. Real experiments. The HYDICE urban data set⁵ and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Indian Pines data set are chosen in

⁵<http://www.tec.army.mil/hypercube>

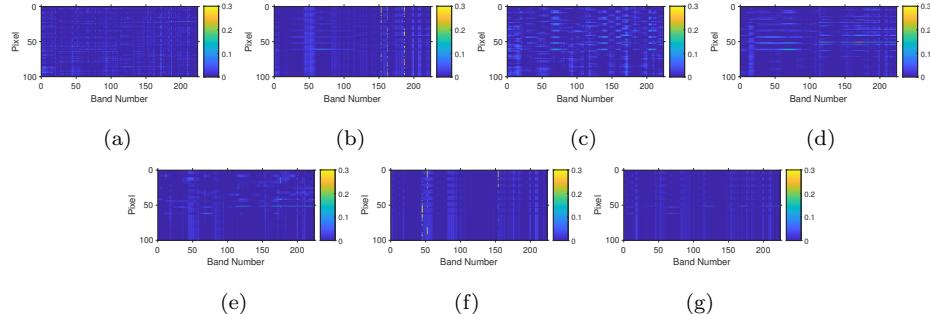


FIGURE 9. Spectral signatures difference of patch centered on pixel (5, 20) between the original image and seven denoised ones under case 3 of the simulated Indian Pines data set. (a) LRMR. (b) LRTV. (c) SSTV. (d) LRTDTV. (e) LRTDGS. (f) LRDSTV₁. (g) LRDSTV₂.

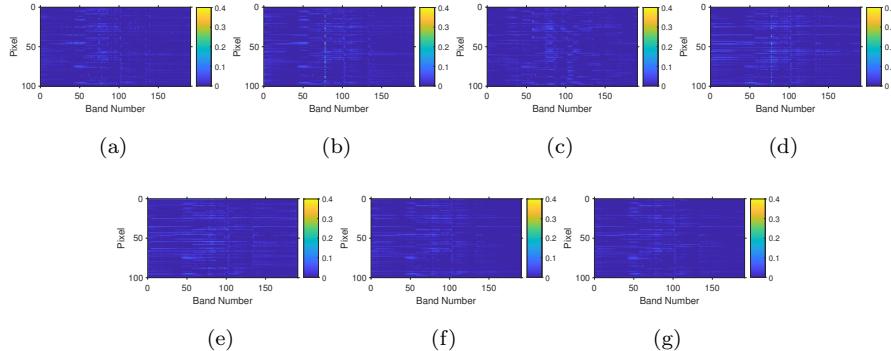


FIGURE 10. Spectral signatures difference of patch centered on pixel (140, 30) between the original image and seven denoised ones under case 3 of the Washington DC Mall data set. (a) LRMR. (b) LRTV. (c) SSTV. (d) LRTDTV. (e) LRTDGS. (f) LRDSTV₁. (g) LRDSTV₂.

our real experiments part. All pixel values have been normalized to [0, 1] before image processing.

4.2.1. HYDICE Urban data set. The HYDICE Urban data set has 307×307 pixels and 210 bands. This data set is corrupted by mixed noise, i.e., Gaussian noise, stripe noise, deadlines and other unknown noise. Figs. 11 and 12 present the denoising results of all compared algorithms for band 139 and band 207 in the Urban data set. These two bands are typical populated bands with deadlines, stripes and Gaussian noise in them which is observed in the two figures. Some noise still remain in LRMR, LRTV, SSTV, and LRDSTV₁ denoising results, so the mixed noise are not be effectively removed. LRTDTV, LRTDGS, and LRDSTV₂ eliminate all mixed noise thus leading the better visual results. However, we notice that LRTDTV and LRDSTV₂ have better image contrast than LRTDGS. The left and right sides

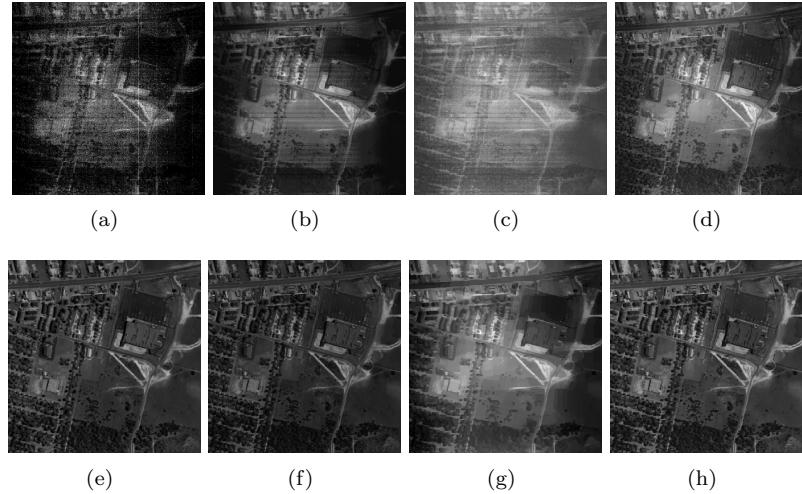


FIGURE 11. Comparison of all denoised images in band 139 of the real Urban data set. (a) Original. (b) LRMR. (c) LRTV. (d) SSTV. (e) LRTDTV. (f) LRTDGS. (g) LRDSTV₁. (h) LRDSTV₂.

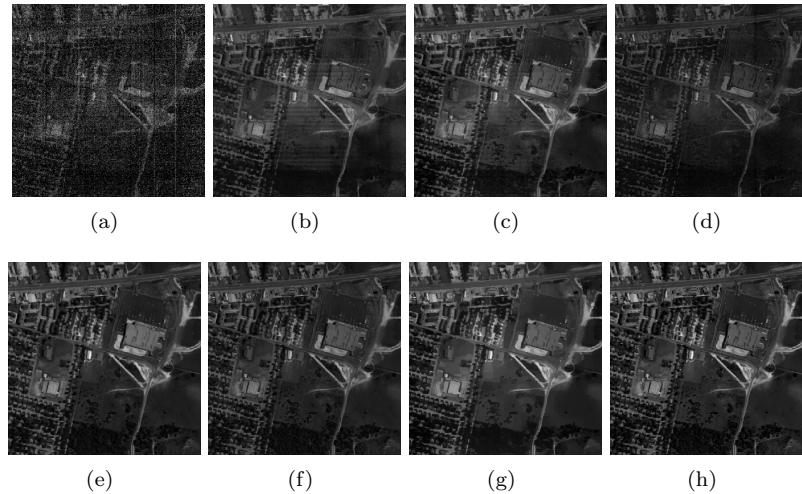


FIGURE 12. Comparison of all denoised images in band 207 of the real Urban data set. (a) Original. (b) LRMR. (c) LRTV. (d) SSTV. (e) LRTDTV. (f) LRTDGS. (g) LRDSTV₁. (h) LRDSTV₂.

of original noisy images are hard to see clearly and become darkness. Results of LRTDTV and LRDSTV₂ bright the sides and make each parts in denoising results more easily to distinguish, what's more, LRDSTV₂ does that better in detail performance. Some parts of the LRTDTV denoising results seem to be smoothed which is caused by the missing details.

Horizontal mean profiles is the next being concerned in the real Urban data set, that of band 207 in the real Urban data set is presented in Fig. 13, which can further

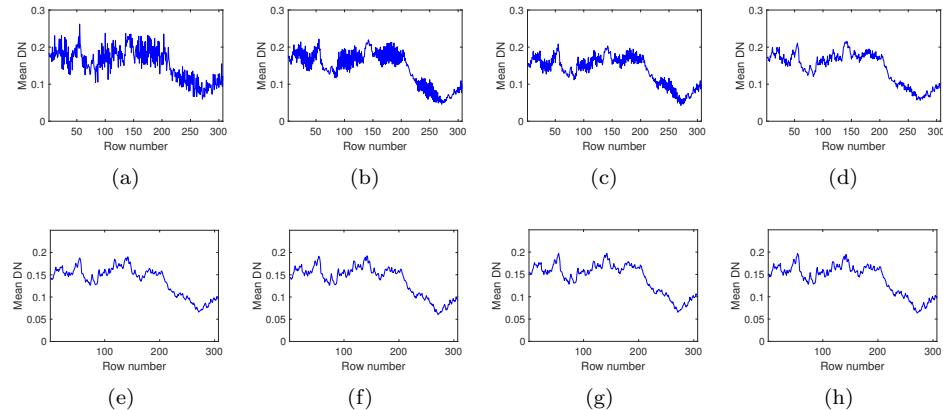


FIGURE 13. Horizontal mean profiles of band 207 in the Urban data set real experiment. (a) Original. (b) LRMR. (c) LRTV. (d) SSTV. (e) LRTDTV. (f) LRTDGS. (g) LRDSTV₁. (h) LRDSTV₂.

explain the denoising results. The horizontal axis represents the row number where the vertical axis is the mean digital number value of each row. Fig. 13(a) implies that sparse noise occur in original image, i.e., stripes and other noise. Many sudden fluctuations appear in the original curve, which is also seen in the LRMR, LRTV, and SSTV. Noise reduction seems not that effectively in these three algorithms, and we are able to notice some stripes in their denoising results of band 207 in the Urban data set. LRTDTV, LRTDGS, and two LRDSTV algorithms have more smooth curves and achieve good enough denoising results. All types of noise are eliminated effectively using these three algorithms. While combining Fig. 13(e)-(h) with Fig. 12(e)-(h), one will see that LRDSTV gets a more reasonable denoising result in the real data experiment.

4.2.2. AVIRIS Indian Pines data set. The popular AVIRIS Indian Pines data set is also applied to compare different denoising results in the real experiment. The size of this tensor data is $145 \times 145 \times 220$. Deadlines, impulse noise, Gaussian noise, water absorption and other noise have destroyed the initial HSI. As we did in the real Urban data set, two typical bands — band 109 and band 220 were chosen and denoising results of them could be seen in Figs. 14 and 15. Figs. 14(a) and 15(a) indicate that noise intensity are at a high level, and it is hard to get useful information about the clean images. In the above two figures, small parts of noise exist in the results of LRMR, LRTV and SSTV and the images are still polluted. LRTDTV, LRTDGS and two LRDSTV algorithms remove all noise in the proper way, but difference appears in the visual results. LRTDTV eliminates mixed noise, however, some parts are not separated clearly. LRTDGS gets nice denoising results while some small details are missing, what's more, the whole image seems a little blurry. Comparing with the other two algorithms, LRDSTV₁ and LRDSTV₂ keep most of the details, preserves the edges and achieves more proper visual results.

Similar to the Urban data set, horizontal mean profiles of band 109 in the Indian Pines data set is shown in Fig. 16. From the presented curves, we get the information that LRMR, LRTV, and SSTV hardly remove all mixed noise. They all fail to clear the heavy noise. The rest algorithms, i.e., LRTDTV, LRTDGS, and LRDSTV

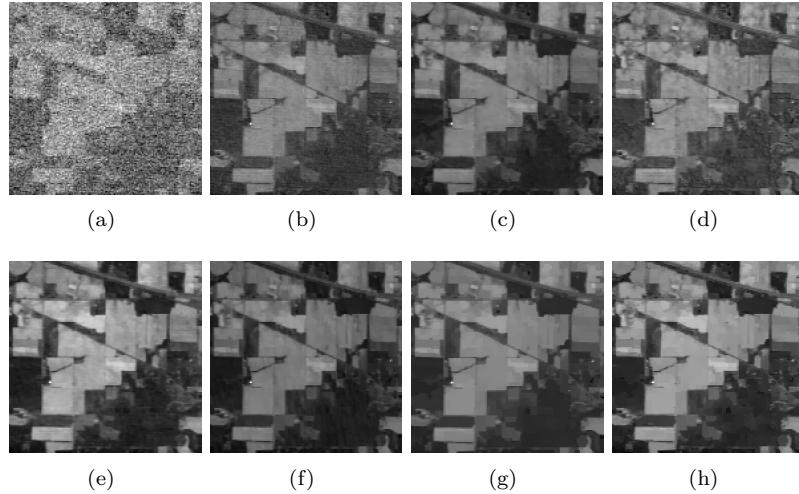


FIGURE 14. Comparison of all denoised images in band 109 of the real Indian Pines data set. (a) Original. (b) LRMRR. (c) LRTV. (d) SSTV. (e) LRTDTV. (f) LRTDGS. (g) LRDSTV₁. (h) LRDSTV₂.

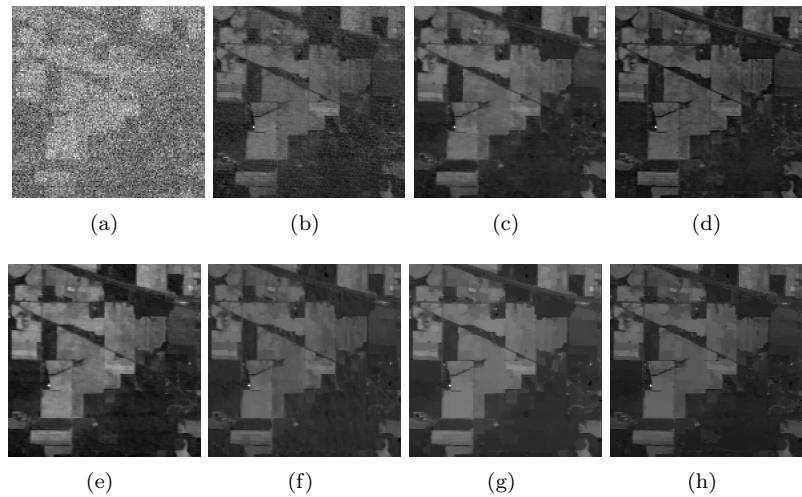


FIGURE 15. Comparison of all denoised images in band 220 of the real Indian Pines data set. (a) Original. (b) LRMRR. (c) LRTV. (d) SSTV. (e) LRTDTV. (f) LRTDGS. (g) LRDSTV₁. (h) LRDSTV₂.

get better curves with fewer fluctuations on them. Observing clearly from the curves, LRDSTV preserves better details while LRTDTV and LRTDGS ignore some small details. Besides, LRDSTV achieves better visual performance, which reflects the superiority of the proposed model. Furthermore, it is shown that for the real data experiments, LRDSTV₂ offers comparable or better denoising results than LRDSTV₁, which is in line with the observation from the simulated experiments.

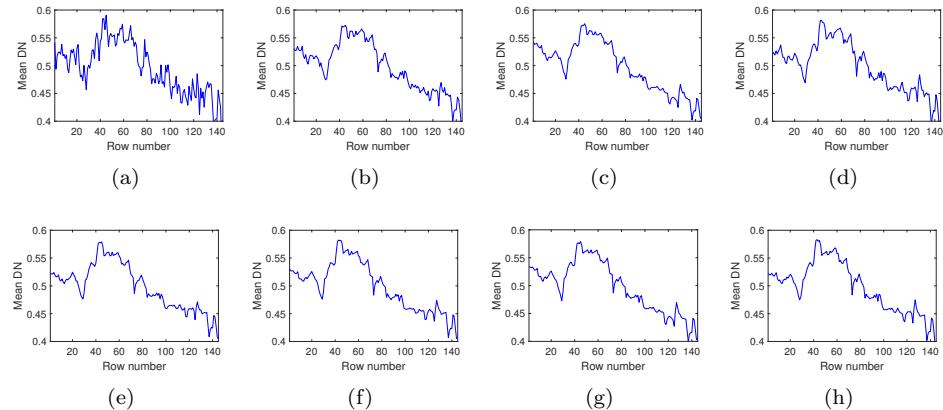


FIGURE 16. Horizontal mean profiles of band 109 in the Indian Pines data set real experiment. (a) Original. (b) LRMR. (c) LRTV. (d) SSTV. (e) LRTDTV. (f) LRTDGS. (g) LRDSTV₁. (h) LRDSTV₂.

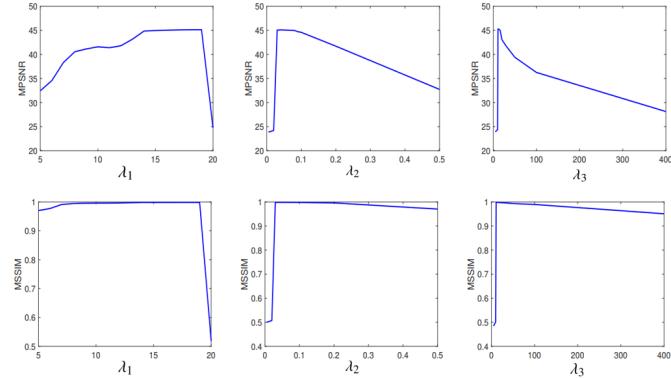


FIGURE 17. Sensitivity analysis of parameters (from left to right) λ_1 , λ_2 , and λ_3 in LRDSTV₂ under case 2 of the simulated Indian Pines data set.

4.3. Discussion. In the proposed LRDSTV model, the denoising result is affected by some parameters, they are λ_1 , λ_2 , λ_3 , and the penalty parameter β . In all simulated and real experiments, we fix the value of β to 4, and this will not change. Regularization parameter λ_1 denotes the sparsity of sparse noise, where we set it as $\lambda_1 = c/\sqrt{mn}$. When the sparse noise level becomes different, the value of c will be set synchronously to guarantee the removal of it. In different cases of different data sets, we find the denoising results are better when c varies from 15 to 150, and that's a relatively stable range. Parameter λ_2 controls the sparsity of difference image, and it also helps to eliminate some noise. As a result, the value of λ_2 is different and it is adjusted according to various cases. For relatively better PSNR and SSIM values, we suggest the value of λ_2 be chosen in the set {0.001, 0.005, 0.01, 0.04, 0.07, 0.1, 0.3, 0.5}. Parameter λ_3 collects low rank information in noisy image,

TABLE 3. Parameters for simulated experiments.

Case	LRDSTV ₁ (c, λ_2, λ_3)			LRDSTV ₂ (c, λ_2, λ_3)	
	Indian Pine	Washington DC	Indian Pine	Washington WDC	
Case 1	$\sigma = 0.025$	150, 0.01, 1	50, 0.001, 20	150, 0.01, 1	50, 0.001, 20
	$\sigma = 0.050$	150, 0.07, 1	50, 0.005, 60	150, 0.07, 1	50, 0.005, 60
	$\sigma = 0.075$	150, 0.30, 1	50, 0.01, 100	150, 0.30, 1	50, 0.01, 100
	$\sigma = 0.100$	150, 0.50, 5	50, 0.01, 200	150, 0.50, 5	50, 0.01, 200
Case 2		17, 0.04, 15	15, 0.004, 100	17, 0.04, 15	15, 0.005, 60
Case 3		15, 0.05, 30	15, 0.008, 100	17, 0.04, 16	15, 0.005, 60

TABLE 4. Parameters for real experiments.

Data	LRDSTV ₁ (c, λ_2, λ_3)	LRDSTV ₂ (c, λ_2, λ_3)
Urban	15, 0.05, 400	10, 0.005, 400
Indian Pines	15, 0.05, 30	15, 0.04, 16

which means that the higher noise intensity needs larger value of λ_3 to balance the information lost causing by too much noise. In the Indian Pines data set, which has fewer pixels than the other simulated data set in each band, λ_3 is set at a low level. That is to say, λ_3 is larger in the Washington DC Mall data set, but its value is still in a small range.

For the sensitivity analysis of parameters, we choose case 2 of the simulated Indian Pines data set in LRDSTV₂ and that will be shown in Fig. 17. In the first two subfigures of Fig. 17, the values of PSNR and SSIM rise at the beginning and then come to a peak when c becomes about 17. Similarly, the resting four subfigures have the same trend of curves which means the values of λ_2 and λ_3 are about 0.04 and 15, respectively. In the following text, Table 3 lists all the suggesting parameters concerned in the simulated experiments. We see that the parameters in LRDSTV₁ and LRDSTV₂ are just the same in case 1. And their parameters are slightly different in cases 2-3. Data in Table 3 tells the robustness of these three parameters. Finally in Table 4, the recommended parameters for two real data sets are displayed too.

Next, we will prove the importance of weights for the double sparsity TV in the update of variable \mathcal{V} in the proposed algorithms. Because of the similarity denoising results of two LRDSTV algorithms, we take LRDSTV₂ for an example. The simulated Indian Pines data set is chosen to measure that. Two cases, as Fig. 18 shows, are involved in to test the effectiveness of weights for spectral sparsity part in our model. Using weights on double sparsity part can better promote smooth structure and greatly improve denoising results according to indices in Fig. 18.

Moreover, we apply both the relative change, i.e., $\|\mathcal{X}^{k+1} - \mathcal{X}^k\|_F^2 / \|\mathcal{X}^k\|_F^2$, and the residual error, i.e., $\|\mathcal{A}\hat{\mathcal{X}}^k + \mathcal{B}\mathcal{Z}^k\|_F$ in LRDSTV₁ and $\|\tilde{\mathcal{A}}\hat{\mathcal{X}}^k + \tilde{\mathcal{B}}\tilde{\mathcal{Z}}^k\|_F$ in LRDSTV₂, to demonstrate the convergence of our algorithms. Figs. 19 and 20 respectively show the curves of the relative change and the residual error with iterations in three different cases in the simulated Indian Pines data set. What we get from these figures is that both the relative change and the residual error begin

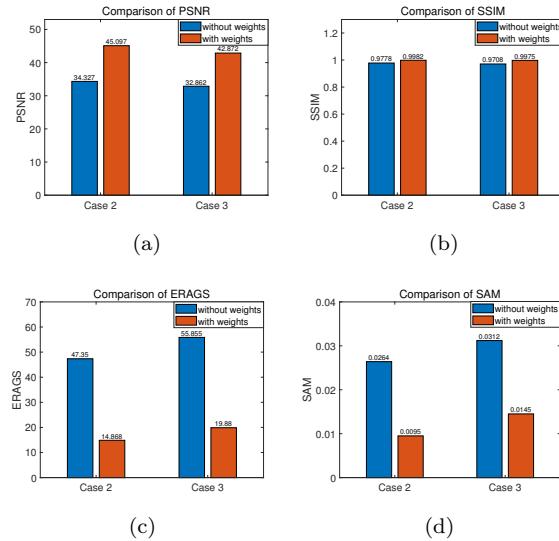


FIGURE 18. Effectiveness of weights for the double sparsity TV in LRDSTV₂ under cases 2 and 3 of the simulated Indian Pines data set. (a) Comparison of PSNR value. (b) Comparison of SSIM value. (c) Comparison of ERAGS value. (d) Comparison of SAM value.

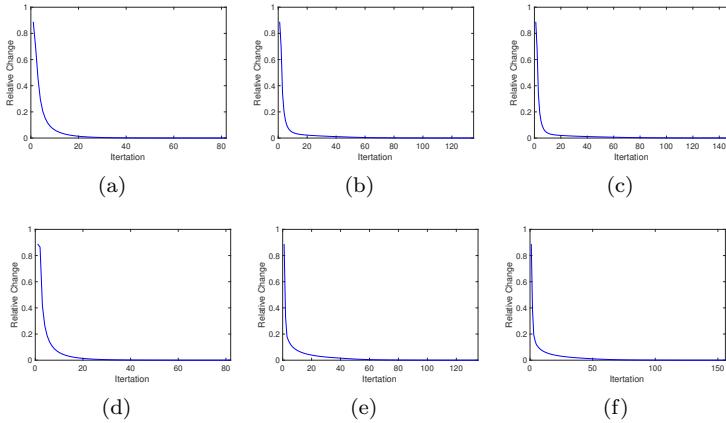


FIGURE 19. Relative change $\|\mathcal{X}^{k+1} - \mathcal{X}^k\|_F^2 / \|\mathcal{X}^k\|_F^2$ against iteration of the LRDSTV solver in the simulated Indian Pines data set. (a)-(c) Cases 1-3 of LRDSTV₁, (d)-(f) cases 1-3 of LRDSTV₂. And the case 1 denotes the fourth subcase of case 1 here.

with a sudden decrease and converges to zero as iteration increases. The above phenomenon numerically shows the convergence of the proposed algorithms.

At last, to further demonstrate the effectiveness of the proposed algorithms, we compare them with the state-of-the-art unsupervised DL algorithm: Stein's unbiased risk estimate convolutional neural network (SURE-CNN) [33]. Recall that

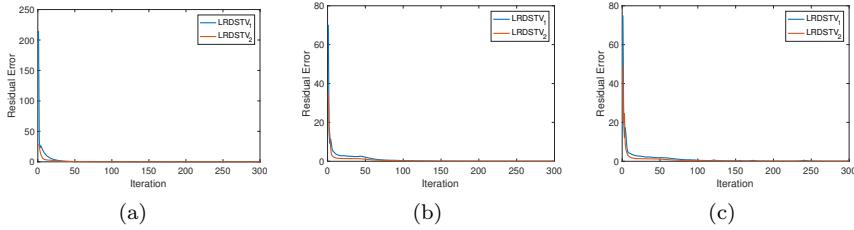


FIGURE 20. Residual error against iteration of LRDSTV in the simulated Indian Pines data set. (a) The fourth subcase of case 1. (b) Case 2. (c) Case 3.

TABLE 5. Quantitative comparison of LRDSTV with SURE-CNN in the Washington DC Mall data set.

Algorithm	Case 1 with $\sigma = 0.05$				Case 2			
	PSNR	SSIM	ERGAS	SAM	PSNR	SSIM	ERGAS	SAM
SURE-CNN	40.638	0.9867	38.782	0.0571	37.642	0.9763	53.820	0.0766
LRDSTV ₁	40.847	0.9865	38.518	0.0627	37.165	0.9738	58.102	0.0876
LRDSTV ₂	40.850	0.9865	38.513	0.0628	37.709	0.9736	54.610	0.0854

SURE-CNN theoretically assumes the Gaussian distribution, so we adopt cases 1 (with noise level $\sigma = 0.05$) and 2 in the simulated experiments for a fair comparison. In addition, we adopt the Washington DC Mall dataset, which has been used in [33]. SURE-CNN uses default parameter setting, except that the learning rate and the number of iterations are set to 0.0001 and 15000, respectively, for better denoising performance. Table 5 lists the quantitative comparison of LRDSTV and SURE-CNN. It can be seen that SURE-CNN and LRDSTV provide comparable SSIM and ERGAS values. LRDSTV offers slightly better PSNR values and SURE-CNN offers better SAM values. In summary, the proposed algorithms still have their own advantages when compared to the powerful DL-based algorithm.

5. Conclusion. In this paper, we have proposed a weighted double sparsity TV and low-rank representation model for HSI mixed noise removal. The weighted double sparsity regularization promotes smooth structure well and properly uses the spatial information of HSIs. What's more, the weighted nuclear norm of the HSI mode-3 unfolding takes advantage of the spectral correlation and helps remain important details to avoid results being over-smoothing. According to the above model, we propose two algorithms named as LRDSTV₁ and LRDSTV₂. More importantly, the combination of these two terms shows the superb denoising performance both in the simulated and real data experiments. Specifically, we note that compared with LRDSTV₁, LRDSTV₂ provides better or comparable denoising results for the considered data sets. In the future, we will consider more appropriate regularization terms to describe the prior information of HSIs and try to utilize the spatial and spectral correlation in a more proper way for better denoising performance.

Acknowledgments. The authors would like to thank the editors and the anonymous reviewers for their constructive, detailed, and helpful advice regarding this paper.

REFERENCES

- [1] H. K. Aggarwal and A. Majumdar, **Hyperspectral image denoising using spatio-spectral total variation**, *IEEE Geosci. Remote Sens. Lett.*, **13** (2016), 442-446.
- [2] X. Bai, F. Xu, L. Zhou, Y. Xing, L. Bai and J. Zhou, **Nonlocal similarity based nonnegative Tucker decomposition for hyperspectral image denoising**, *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, **11** (2018), 701-712.
- [3] E. J. Candès, X. Li, Y. Ma and J. Wright, **Robust principal component analysis?**, *J. ACM*, **58** (2011), 1-37.
- [4] Y. Chen, W. He, N. Yokoya and T.-Z. Huang, **Hyperspectral image restoration using weighted group sparsity-regularized low-rank tensor decomposition**, *IEEE Trans. Cybern.*, **50** (2020), 3556-3570.
- [5] Y. Chen, W. He, N. Yokoya, T.-Z. Huang and X.-L. Zhao, **Nonlocal tensor-ring decomposition for hyperspectral image denoising**, *IEEE Trans. Geosci. Remote Sens.*, **58** (2020), 1348-1362.
- [6] Y. Chen, T.-Z. Huang, W. He, X.-L. Zhao, H. Zhang and J. Zeng, **Hyperspectral image denoising using factor group sparsity-regularized nonconvex low-rank approximation**, *IEEE Trans. Geosci. Remote Sens.*, **60** (2022), 1-16.
- [7] Y. Chen, T.-Z. Huang, X.-L. Zhao and L.-J. Deng, **Hyperspectral image restoration using framelet-regularized low-rank nonnegative matrix factorization**, *Appl. Math. Model.*, **63** (2018), 128-147.
- [8] K. Dabov, A. Foi, V. Katkovnik and K. Egiazarian, **Image denoising by sparse 3-D transform-domain collaborative filtering**, *IEEE Trans. Image Process.*, **16** (2007), 2080-2095.
- [9] L. M. Dale, A. Thewis, C. Boudry, I. Rotar, P. Dardenne, V. Baeten and J. A. F. Pierna, **Hyperspectral imaging applications in agriculture and agro-food product quality and safety control: A review**, *Appl. Spectrosc. Rev.*, **48** (2013), 142-159.
- [10] B. Du, Z. Huang and N. Wang, **A bandwise noise model combined with low-rank matrix factorization for hyperspectral image denoising**, *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, **11** (2018), 1070-1081.
- [11] M. Elad and M. Aharon, **Image denoising via sparse and redundant representations over learned dictionaries**, *IEEE Trans. Image Process.*, **15** (2006), 3736-3745.
- [12] H. Fan, J. Li, Q. Yuan, X. Liu and M. Ng, **Hyperspectral image denoising with bilinear low rank matrix factorization**, *Signal Process.*, **163** (2019), 132-152.
- [13] A. A. Gowen, C. P. O'Donnell, P. J. Cullen, G. Downey and J. M. Frias, **Hyperspectral imaging—An emerging process analytical tool for food quality and safety control**, *Trends Food Sci. Technol.*, **18** (2007), 590-598.
- [14] H. Grahn and P. Geladi, **Techniques and Applications of Hyperspectral Image Analysis**, John Wiley & Sons, 2007.
- [15] S. Gu, L. Zhang, W. Zuo and X. Feng, **Weighted nuclear norm minimization with application to image denoising**, In *Proc. IEEE Comput. Vis. Pattern Recognit.*, (2014), 2862-2869.
- [16] X. Guo, X. Huang, L. Zhang and L. Zhang, **Hyperspectral image noise reduction based on rank-1 tensor decomposition**, *ISPRS J. Photogramm. Remote Sens.*, **83** (2013), 50-63.
- [17] W. He, Q. Yao, C. Li, N. Yokoya and Q. Zhao, **Non-local meets global: An integrated paradigm for hyperspectral denoising**, In *Proc. IEEE Comput. Vis. Pattern Recognit.*, (2019), 6868-6877.
- [18] W. He, H. Zhang, H. Shen and L. Zhang, **Hyperspectral image denoising using local low-rank matrix recovery and global spatial-spectral total variation**, *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, **11** (2018), 713-729.
- [19] W. He, H. Zhang, L. Zhang and H. Shen, **Hyperspectral image denoising via noise-adjusted iterative low-rank matrix approximation**, *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, **8** (2015), 3050-3061.
- [20] 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.*, **54** (2016), 178-188.
- [21] J. Huang, T.-Z. Huang, L.-J. Deng and X.-L. Zhao, **Joint-sparse-blocks and low-rank representation for hyperspectral unmixing**, *IEEE Trans. Geosci. Remote Sens.*, **57** (2019), 2419-2438.

- [22] J. Huang, T.-Z. Huang, X.-L. Zhao and L.-J. Deng, **Nonlocal tensor-based sparse hyperspectral unmixing**, *IEEE Trans. Geosci. Remote Sens.*, **59** (2021), 6854-6868.
- [23] T. G. Kolda and B. W. Bader, **Tensor decompositions and applications**, *SIAM Rev.*, **51** (2009), 455-500.
- [24] D. Landgrebe, **Hyperspectral image data analysis**, *IEEE Signal Process. Mag.*, **19** (2002), 17-28.
- [25] T. Lin and S. Bourennane, **Survey of hyperspectral image denoising methods based on tensor decompositions**, *EURASIP J. Adv. Signal Process.*, **2013** (2013), 186.
- [26] G. Liu, Z. Lin, S. Yan, et al., **Robust recovery of subspace structures by low-rank representation**, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2013), 171-184.
- [27] T. Liu, D. Hu, Z. Wang, J. Gou and Wu Chen, **Hyperspectral image denoising using nonconvex fraction function**, *IEEE Geosci. Remote Sens. Lett.*, **20** (2023), 1-5.
- [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.*, **50** (2012), 3717-3724.
- [29] T. Lu, S. Li, L. Fang, Y. Ma and J. A. Benediktsson, **Spectral-spatial adaptive sparse representation for hyperspectral image denoising**, *IEEE Trans. Geosci. Remote Sens.*, **54** (2016), 373-385.
- [30] A. Maffei, J. M. Haut, M. E. Paoletti, J. Plaza, L. Bruzzone and A. Plaza, **A single model CNN for hyperspectral image denoising**, *IEEE Trans. Geosci. Remote Sens.*, **58** (2020), 2516-2529.
- [31] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, **Non-local sparse models for image restoration**, In *Proc. IEEE Comput. Vis. Pattern Recognit.*, (2009), 2272-2279.
- [32] S. Meng, L.-T. Huang and W.-Q. Wang, **Tensor decomposition and PCA jointed algorithm for hyperspectral image denoising**, *IEEE Geosci. Remote Sens. Lett.*, **13** (2016), 897-901.
- [33] H. V. Nguyen, M. O. Ulfarsson and J. R. Sveinsson, **Hyperspectral image denoising using SURE-based unsupervised convolutional neural networks**, *IEEE Trans. Geosci. Remote Sens.*, **59** (2021), 3369-3382.
- [34] H. Othman and S.-E. Qian, **Noise reduction of hyperspectral imagery using hybrid spatial-spectral derivative-domain wavelet shrinkage**, *IEEE Trans. Geosci. Remote Sens.*, **44** (2006), 397-408.
- [35] N. Pan, M. Hou, S. Lv, Y. Hu, X. Zhao, Q. Ma, S. Li and A. Shaker, **Extracting faded mural patterns based on the combination of spatial-spectral feature of hyperspectral image**, *J. Cult. Herit.*, **27** (2017), 80-87.
- [36] A. Plaza, J. A. Benediktsson, et al., **Recent advances in techniques for hyperspectral image processing**, *Remote Sens. Environ.*, **113** (2009), S110-S122.
- [37] L. I. Rudin, S. Osher and E. Fatemi, **Nonlinear total variation based noise removal algorithms**, *Physica D*, **60** (1992), 259-268.
- [38] G. Shaw and D. Manolakis, **Signal processing for hyperspectral image exploitation**, *IEEE Signal Process. Mag.*, **19** (2002), 12-16.
- [39] P. Shippert, **Introduction to hyperspectral image analysis**, *J. Space Comm.*, **2** (2003), 1-13.
- [40] O. Sidorov and J. Y. Hardeberg, **Deep hyperspectral prior: Single-image denoising, inpainting, super-resolution**, In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, (2019), 3844-3851.
- [41] Q. Tian, **Study on oil-gas reservoir detecting methods using hyperspectral remote sensing**, *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, **39** (2012), 157-162.
- [42] M. Wang, J. Yu, J.-H. Xue and W. Sun, **Denoising of hyperspectral images using group low-rank representation**, *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, **9** (2016), 4420-4427.
- [43] Q. Wang, Z. Wu, J. Jin, T. Wang and Y. Shen, **Low rank constraint and spatial spectral total variation for hyperspectral image mixed denoising**, *Signal Process.*, **142** (2018), 11-26.
- [44] Y. Wang, J. Peng, Q. Zhao, Y. Leung, X.-L. Zhao and D. Meng, **Hyperspectral image restoration via total variation regularized low-rank tensor decomposition**, *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, **11** (2018), 1227-1243.
- [45] Y. Wang, W. Yin and J. Zeng, **Global convergence of ADMM in nonconvex nonsmooth optimization**, *J. Sci. Comput.*, **78** (2019), 29-63.
- [46] 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.*, **13** (2004), 600-612.
- [47] L. Wu, J. Huang and M.-S. Guo, **Multidimensional low-rank representation for sparse hyperspectral unmixing**, *IEEE Geosci. Remote Sens. Lett.*, **20** (2023), 5502805.

- [48] Z. Wu, Q. Wang, J. Jin and Y. Shen, **Structure tensor total variation-regularized weighted nuclear norm minimization for hyperspectral image mixed denoising**, *Signal Process.*, **131** (2017), 202-219.
- [49] Z. Wu, Q. Wang, Z. Wu and Y. Shen, **Total variation-regularized weighted nuclear norm minimization for hyperspectral image mixed denoising**, *J. Electron. Imaging*, **25** (2016), 13-37.
- [50] F. Xu, Y. Chen, C. Peng, Y. Wang, X. Liu and G. He, **Denoising of hyperspectral image using low-rank matrix factorization**, *IEEE Geosci. Remote Sens. Lett.*, **14** (2017), 1141-1145.
- [51] J. Xu, L. Zhang, D. Zhang and X. Feng, **Multi-channel weighted nuclear norm minimization for real color image denoising**, In *Proc. IEEE Int. Conf. Comput. Vis.*, (2017), 1096-1104.
- [52] J. Xue and Y. Zhao, **Rank-1 tensor decomposition for hyperspectral image denoising with nonlocal low-rank regularization**, In *Proc. Int. Conf. Mach. Vis. Inf. Technol. (CMVIT)*, (2017), 40-45.
- [53] J. Xue, Y. Zhao, W. Liao and J. C.-W. Chan, **Nonlocal low-rank regularized tensor decomposition for hyperspectral image denoising**, *IEEE Trans. Geosci. Remote Sens.*, **57** (2019), 5174-5189.
- [54] M. Ye, Y. Qian and J. Zhou, **Multitask sparse nonnegative matrix factorization for joint spectral-spatial hyperspectral imagery denoising**, *IEEE Trans. Geosci. Remote Sens.*, **53** (2015), 2621-2639.
- [55] Q. Yuan, L. Zhang and H. Shen, **Hyperspectral image denoising employing a spectral-spatial adaptive total variation model**, *IEEE Trans. Geosci. Remote Sens.*, **50** (2012), 3660-3677.
- [56] H. Zhang, W. He, L. Zhang, H. Shen and Q. Yuan, **Hyperspectral image restoration using low-rank matrix recovery**, *IEEE Trans. Geosci. Remote Sens.*, **52** (2014), 4729-4743.
- [57] Y. Zhang, W.-H. Zhang, H. Chen, M.-L. Yang, T.-Y. Li and J.-L. Zhou, **Few-view image reconstruction combining total variation and a high-order norm**, *Int. J. Imag. Syst. Technol.*, **23** (2013), 249-255.

Received June 2023; revised December 2023; early access March 2024.